# Documentation 7.4

ZABBIX

12.06.2026

## Contents


**Руководство пользователя** **5**

**Добро пожаловать в документацию Zabbix** **5**

Начало работы с Zabbix . . . 5
Zabbix Cloud . . . 5
Центр разработчиков . . . 5
Сообщество и другие ресурсы . . . 5
1. Введение . . . 5
1. Структура руководства . . . 6
2. Что такое Zabbix . . . 6
3. Возможности Zabbix . . . 7
4. Обзор Zabbix . . . 8
5 Что нового в Zabbix 7.4.0 . . . 9
6 Что нового в Zabbix 7.4.x . . . 18
2. Определения . . . 21
3. Процессы Zabbix . . . 23
1. Сервер . . . 23
2. Агент . . . 32
3. Агент 2 . . . 36
4. Прокси . . . 40
5. Java gateway . . . 44
6. Sender . . . 48
7. Get . . . 49
8. JS . . . 50
9. Веб-сервис . . . 51
4. Установка и первые шаги . . . 51
1. Получение Zabbix . . . 52
2. Требования . . . 53
3. Установка из исходных кодов . . . 63
4. Установка из пакетов . . . 79
5. Установка из контейнеров . . . 92
6. Установка веб-интерфейса . . . 100
7. Процедура обновления . . . 107
8. Известные проблемы . . . 120
9 Изменения шаблонов . . . 131
10 Примечания к обновлению до 7.4.0 . . . 132
11 Примечания к обновлению до 7.4.x . . . 135
5 Быстрый старт . . . 136
1 Login and configuring user . . . 137
2. Новый узел сети . . . 141
3. Новый элемент данных . . . 143
4. Новый триггер . . . 145
5 Receiving problem notification . . . 146
6. Новый шаблон . . . 150
6. Готовое решение Zabbix . . . 152
7. Настройка . . . 155
1. Узлы сети и группы узлов сети . . . 165
2. Элементы данных . . . 184
3. Триггеры . . . 408
4. События . . . 428
5. Корреляция событий . . . 431

6. Тегирование . . . 438
7. Визуализация . . . 441
8. Шаблоны и группы шаблонов . . . 467
9. Готовые шаблоны . . . 467
10. Оповещения на события . . . 481
11. Макросы . . . 533
12. Пользователи и группы пользователей . . . 548
13. Хранение секретов . . . 557
14. Регулярные отчёты . . . 565
15. Экспорт данных . . . 569
8. Мониторинг услуг . . . 576
1 Дерево услуг . . . 577
2. SLA . . . 581
3. Пример настройки . . . 583
9. Веб-мониторинг . . . 587
1. Элементы данных веб-мониторинга . . . 596
2. Сценарий из реальной жизни . . . 598
10. Мониторинг виртуальных машин . . . 605
1. Ключи элементов данных для мониторинга VMware . . . 607
2. Поля ключей обнаружения виртуальных машин . . . 626
3. Примеры JSON для элементов данных VMware . . . 630
4. Пример настройки мониторинга VMware . . . 635
11. Обслуживание . . . 640
12. Регулярные выражения . . . 646
13. Подтверждение проблем . . . 652
1. Подавление проблем . . . 654
14. Экспорт/импорт настроек . . . 655
1. Группы шаблонов . . . 657
2. Группы узлов сети . . . 657
3. Шаблоны . . . 657
4. Узлы сети . . . 675
5. Карты сети . . . 692
6. Способы оповещений . . . 701
15. Обнаружение . . . 708
1. Сетевое обнаружение . . . 708
2. Авторегистрация активных агентов . . . 718
3. Низкоуровневое обнаружение (LLD) . . . 722
16. Распределённый мониторинг . . . 779
1. Прокси . . . 780
17. Шифрование . . . 789
1. Использование сертификатов . . . 796
2. Использование общих ключей (pre-shared keys) . . . 803
3. Решение проблем . . . 806
18. Веб-интерфейс . . . 809
1. Меню . . . 809
2. Разделы веб-интерфейса . . . 818
3 Настройки пользователя . . . 1059
4 Глобальный поиск . . . 1064
5 Режим обслуживания веб-интерфейса . . . 1066
6 Параметры страницы . . . 1067
7 Определения . . . 1070
8 Создание своей темы . . . 1071
9 Режим отладки . . . 1071
10 Файлы cookie используемые Zabbix . . . 1072
11 Часовые пояса . . . 1073
12 Сброс пароля . . . 1074
13. Селектор периода времени . . . 1074
19. Наилучшие практики . . . 1076
1. Наилучшие практики по безопасности . . . 1076
2 Лучшие практики настройки . . . 1085
20. API . . . 1086
Справочник методов . . . 1091
Приложение 1. Справочные комментарии . . . 1922

Приложение 2. Изменения с 7.2 до 7.4 . . . 1929
Приложение 3. Изменения в 7.4 . . . 1930
21. Расширения . . . 1930
1. Подгружаемые модули . . . 1933
2. Плагины . . . 1942
3. Модули веб-интерфейса . . . 1950
22. Приложения . . . 1951
1. Установка и настройка . . . 1951
2. Конфигурация демонов . . . 2010
3. Протоколы . . . 2097
4. Элементы данных . . . 2123
5 Поддерживаемые функции . . . 2160
6. Макросы . . . 2193
7. Символы единиц измерения . . . 2230
8. Настройка периодов времени . . . 2232
9. Выполнение команд . . . 2232
10. Совместимость версий . . . 2233
11 Динамическая библиотека Zabbix sender для Windows . . . 2235
12. Обновление мониторинга услуг . . . 2235
13. Другие проблемы . . . 2236
14. Отличия между Zabbix агентом и Zabbix агентом 2 . . . 2237
15. Примеры экранирования . . . 2238
23 Краткие справочные руководства . . . 2239
Обзор . . . 2240
1 Мониторинг Linux с помощью агента Zabbix . . . 2240
2 Мониторинг Windows с помощью агента Zabbix . . . 2244
3 Мониторинг Apache через HTTP . . . 2247
4 Мониторинг MySQL с помощью агента Zabbix 2 . . . 2252
5 Мониторинг VMware с помощью Zabbix . . . 2259
6 Мониторинг сетевого трафика с помощью Zabbix . . . 2263
7 Мониторинг сетевого трафика с использованием активных проверок . . . 2266
8 Мониторинг веб-сайтов с помощью элементов данных Browser . . . 2269
9 Мониторинг сертификатов веб-сайтов с помощью Zabbix agent 2 (пассивный) . . . 2274
10 Мониторинг сетевого коммутатора или маршрутизатора с помощью Zabbix . . . 2278
11 Мониторинг журнала событий Windows с использованием активных проверок . . . 2285

**Zabbix Cloud** **2289**

**Zabbix Cloud** **2289**
Настройка и управление Zabbix Cloud . . . 2290
Узнайте о Zabbix Cloud . . . 2290
Развертывание Zabbix в облаке . . . 2290
Конфигурация узла сети . . . 2294
Обзор . . . 2294
Вкладка Overview . . . 2295
Вкладка фильтров доступа . . . 2296
Вкладка Encryption . . . 2297
Вкладка Backups . . . 2298
Вкладка History . . . 2298
Вкладка Maintenance . . . 2299
Вкладка Upgrade . . . 2300
Добавление пользователей . . . 2302
Введение . . . 2302
Добавление пользователей в организации . . . 2302
Добавление пользователей к нодам . . . 2303
Основные отличия Zabbix Cloud . . . 2304
Введение . . . 2304
Журнал аудита . . . 2306

**Центр разработчика** **2308**

**Центр разработчика** **2308**
Сборка на Zabbix . . . 2308
Модули . . . 2308

Файловая структура модуля . . . 2309
Виджеты . . . 2316
Руководства . . . 2325
Примеры . . . 2351
Библиотека Python для Zabbix . . . 2352
Установка . . . 2352
Руководство по быстрому началу работы . . . 2353
Использование Zabbix API . . . 2356
Сбор данных с агента Zabbix . . . 2359
Отправка данных на сервер Zabbix или прокси . . . 2361
Отладочное логирование . . . 2364
Плагины . . . 2365
Примеры . . . 2367
Интерфейсы плагинов . . . 2367
Создание плагина (руководство) . . . 2367
Изменения в разработке расширений . . . 2372

**Страницы помощи Zabbix** **2372**
zabbix_agent2 . . . 2372
zabbix_agentd . . . 2374
zabbix_get . . . 2376
zabbix_js . . . 2378
zabbix_proxy . . . 2379
zabbix_sender . . . 2381
zabbix_server . . . 2386
zabbix_web_service . . . 2388

**Уведомление об авторских правах** **2389**

# Руководство пользователя

## Добро пожаловать в документацию Zabbix

Ваш основной источник информации по работе с мониторингом Zabbix — от базовой настройки до расширенных конфигураций. Это руководство охватывает все необходимое для установки, настройки и эксплуатации Zabbix.

Что нового в Zabbix 7.4

### Начало работы с Zabbix

**Установка** Пошаговая инструкция по установке Zabbix на выбранную вами платформу, охватывающая различные операционные системы и конфигурации баз данных.

**Требования** Список поддерживаемых платформ и необходимых программных компонентов, чтобы помочь вам подготовить среду для успешного развёртывания Zabbix.

**Краткие руководства** Краткие практические инструкции, которые проведут вас через основы — от первых шагов настройки до получения первого уведомления о проблеме.

### Zabbix Cloud

**Начните использовать Zabbix Cloud** Пошаговые инструкции по быстрой настройке и запуску вашего экземпляра Zabbix Cloud, включая первоначальную настройку и подключение отслеживаемых устройств.

**Zabbix: облачная версия или локальная** Сравните различия в управлении, масштабируемости и обслуживании между двумя вариантами развёртывания и выясните, какой из них лучше всего подходит для поддержки вашего рабочего процесса.

**Изучите Zabbix Cloud** Ознакомьтесь с обзором Zabbix Cloud, включая его ключевые функции, преимущества и условия подписки. Часто задаваемые вопросы по настройке и использованию смотрите в разделе FAQ.

### Центр разработчиков

**Модули веб-интерфейса** Руководства и справочные материалы по созданию пользовательских модулей, расширяющих или модифицирующих веб-интерфейс Zabbix для решения конкретных задач.

**Виджеты** Подробный анализ структуры и логики виджетов, а также инструкции по созданию пользовательских элементов панели, адаптированных под ваши потребности.

**Плагины** Обзор того, как разрабатывать и управлять плагинами Zabbix для расширения функциональности или интеграции с внешними системами.

### Сообщество и другие ресурсы

**Форумы Zabbix** Пространство для обмена идеями, поиска решений и обмена опытом между специалистами Zabbix всех уровней.

**Блог Zabbix** Новости, обучающие материалы и примеры из практики, подготовленные командой Zabbix и участниками сообщества.

**Видеоинструкции** Видеотека с демонстрациями, обсуждениями и советами, которые помогут вам максимально эффективно использовать Zabbix.

## 1. Введение

Zabbix — это платформа мониторинга корпоративного уровня с открытым исходным кодом. Она отслеживает параметры в сетях и IT-инфраструктуре (от серверов и приложений до баз данных, виртуальных машин и сервисов), а затем отправляет оповещения при возникновении проблем, сохраняя при этом исторические данные для анализа и визуализации тенденций.

Вы можете создавать панели, автоматизировать реагирование на проблемы и визуализировать динамику изменений. Zabbix может масштабироваться от небольших сетей до инфраструктур с тысячами узлов сети.

В данном руководстве описаны установка, настройка и продвинутые темы. Для поиска информации по конкретной теме воспользуйтесь боковым меню навигации.

## 1. Структура руководства

Структура

Содержимое данного руководства разделено на разделы и подразделы для обеспечения удобного доступа к представляющим интерес темам.

При переходе к соответствующим разделам убедитесь, что вы раскрываете папки раздела, чтобы была возможность увидеть содержимое, находящееся в подразделах и на отдельных страницах.

Перекрёстные ссылки между страницами соответствующего содержания используются как можно чаще, чтобы убедиться, что актуальная информация не будет упущена пользователями.

Разделы

Введение содержит общую информацию о текущем программном обеспечении Zabbix. Изучение этого раздела поможет вам понять, почему стоит выбрать Zabbix.

Определения Zabbix объясняют терминологию, используемую в Zabbix, а процессы Zabbix содержат подробности о компонентах Zabbix.

Разделы Установка и Быстрый старт помогут вам начать работу с Zabbix. Zabbix appliance — это альтернатива для быстрого знакомства с тем, каково это — использовать Zabbix.

Конфигурация — один из самых больших и важных разделов этого руководства. Он содержит множество важных рекомендаций о том, как настроить Zabbix для мониторинга вашей среды: от настройки узлов сети до сбора и просмотра важных данных, а также настройки уведомлений и удаленных команд, выполняемых в случае проблем.

Мониторинг сервисов подробно описывает, как использовать Zabbix для получения общего представления о вашей среде мониторинга.

Веб-мониторинг поможет вам узнать, как контролировать доступность веб-сайтов.

Мониторинг виртуальных машин представляет собой руководство по настройке мониторинга среды VMware.

Обслуживание, Регулярные выражения, Подтверждение проблем и Экспорт/импорт конфигурации — это дополнительные разделы, которые показывают, как использовать различные аспекты программного обеспечения Zabbix.

Обнаружение содержит инструкции по настройке автоматического обнаружения сетевых устройств, активных агентов, файловых систем, сетевых интерфейсов и т. д.

Распределенный мониторинг рассматривает возможности использования Zabbix в более крупных и сложных средах.

Шифрование помогает понять возможности шифрования обмена данными между компонентами Zabbix.

Веб-интерфейс содержит информацию, специфичную для использования веб-интерфейса Zabbix.

Лучшие практики содержат полезные советы для оптимального и безопасного использования.

Раздел API представляет подробности работы с API Zabbix.

Расширения — это набор способов расширения Zabbix.

Подробные списки технической информации приведены в Приложениях.

Краткие руководства содержат пошаговые инструкции по выполнению задач в Zabbix.

## 2. Что такое Zabbix

Обзор

Zabbix был создан Алексеем Владышевым, и в настоящее время активно разрабатывается и поддерживается компанией Zabbix SIA.

Zabbix — это решение распределённого мониторинга корпоративного класса с открытыми исходными кодами.

Zabbix — это программное обеспечение для мониторинга многочисленных параметров сети, жизнеспособности и целостности серверов, виртуальных машин, приложений, сервисов, баз данных, веб-сайтов, облачных сред и многого другого. Zabbix использует гибкий механизм оповещений, что позволяет пользователям настраивать основанные на e-mail уведомления практически на любое событие. Такой подход позволяет быстро реагировать на проблемы с серверами. Zabbix предлагает отличные функции отчётности и визуализации данных, основанные на данных истории. Это делает Zabbix идеальным при планировании мощностей.

Zabbix поддерживает как поллеры, так и трапперы. Все отчёты и статистика Zabbix, так же как и параметры настройки, доступны через Веб-интерфейс. Веб-интерфейс обеспечивает доступ к информации о состоянии вашей сети и жизнеспособности ваших серверов из любого места. Корректно настроенный, Zabbix может играть важную роль в мониторинге ИТ инфраструктуры. Это верно и для маленьких организаций с несколькими серверами, и для больших организаций со множеством серверов.

Zabbix бесплатен. Zabbix написан и распространяется под лицензией AGPL-3.0. Это означает, что его исходный код свободно распространяется и доступен для неограниченного круга лиц.

Коммерческая поддержка [en] доступна и осуществляется как самой компанией Zabbix, так и партнёрами компании по всему миру.

Узнайте больше о возможностях Zabbix.

Пользователи Zabbix

Множество организаций различных размеров по всему миру полагаются на Zabbix как на основную платформу мониторинга.

## 3. Возможности Zabbix

Обзор

Zabbix — высокоинтегрированное решение для мониторинга сети, предлагающее множество функций в одном пакете.

### Сбор данных

- проверки доступности и производительности
- поддержка мониторинга с использованием SNMP (и трапперы, и поллеры), IPMI, JMX, VMware
- пользовательские проверки
- сбор желаемых данных с использованием пользовательских интервалов
- выполняются сервером/прокси и агентами

### Гибкие определения порогов

- вы можете задавать очень гибкие пороги проблем, называемые триггерами, ссылаясь на значения из базы данных

### Множество настроек оповещений

- отправка оповещений может быть настроена с использованием расписания эскалаций, получателей, способов оповещений
- оповещения можно сделать информативными и полезными с использованием переменных макросов
- автоматические действия, включающие в себя удалённые команды

### Построение графиков в режиме реального времени

- при помощи встроенного функционала построения графиков сразу же доступны графики по наблюдаемым элементам данных

### Возможности веб-мониторинга

- Zabbix может имитировать нажатия мышкой на веб-сайте, проверять работоспособность и время ответа

### Широкие возможности визуализации

- возможность создавать пользовательские графики, что позволяет комбинировать множество элементов данных в одном месте
- карты сети
- слайд-шоу в обзоре в стиле панели
- отчёты
- высокоуровневое (бизнес) представление наблюдаемых ресурсов

### Хранение данных истории

- запись данных в базу данных
- настраиваемая история
- встроенная процедура очистки истории

**Простая настройка**

- добавление наблюдаемых устройств в виде узлов сети
- как только узлы сети появляются в базе данных, они сразу готовы к мониторингу
- применение шаблонов к наблюдаемым устройствам

**Использование шаблонов**

- группировка проверок в шаблоны
- шаблоны могут наследоваться от других шаблонов

**Сетевое обнаружение**

- автоматическое обнаружение сетевых устройств
- автоматическая регистрация агентов
- обнаружение файловых систем, сетевых устройств и SNMP OID’ов

**Быстрый веб-итерфейс**

- веб-интерфейс, основанный на языке PHP
- доступен из любого места
- удобная навигация
- журнал аудита

**Zabbix API**

- Zabbix API предоставляет программируемый интерфейс к Zabbix для массовых манипуляций, интеграции стороннего программного обеспечения и других целей.

**Система прав доступа**

- безопасная аутентификация пользователей
- возможность ограничения доступа отдельным пользователям к конкретным страницам

**Полнофункциональный и легко расширяемый агент**

- разворачивается на наблюдаемых машинах
- можно развернуть как на Linux, так и на Windows

**Бинарные демоны**

- написаны на языке Си, имеют высокую производительность и используют небольшой объём памяти
- легко переносимы

**Готовность к сложным средам**

- простой удалённый мониторинг с использованием Zabbix прокси

### 4. Обзор Zabbix

Архитектура

Zabbix состоит из нескольких основных программных компонентов. Их обязанности описаны ниже.

Сервер

Zabbix server — это центральный компонент, которому агенты передают информацию о доступности и целостности, а также статистику. Сервер является центральным хранилищем, в котором сохраняются все конфигурационные, статистические и операционные данные.

Хранилище базы данных

Вся информация о конфигурации, а также данные, собираемые Zabbix, хранятся в базе данных.

Веб-интерфейс

Интерфейс на основе Веб поставляется для лёгкого доступа к Zabbix из любого места и с любой платформы. Интерфейс является частью Zabbix сервера и обычно (но не обязательно) работает на той же самой физической машине, что и сервер.

Прокси

Zabbix прокси может собирать данные о производительности и доступности от имени Zabbix сервера. Прокси является опциональной частью Zabbix; однако он может быть полезен, чтобы распределить нагрузку одного Zabbix сервера.

Агент

Zabbix агенты разворачиваются на наблюдаемых системах для активного мониторинга за локальными ресурсами и приложениями, и для отправки собранных данных Zabbix серверу или прокси. Начиная с Zabbix 4.4, имеются два типа агентов: Zabbix агент (легковесный, поддерживается на большом количестве платформ, написан на языке C) и Zabbix агент 2 (особо гибкий, легко расширяемый при помощи плагинов, написан на языке Go).

Поток данных

Кроме того, важно сделать шаг назад и взглянуть на весь поток данных в Zabbix. Для того чтобы создать элемент данных, который будет собирать данные, вы должны сначала создать узел сети. Перемещаясь в другой конец спектра Zabbix, у вас должен быть элемент данных, чтобы создать триггер. У вас должен быть триггер, чтобы создать действие. Таким образом, если вы хотите получать оповещения о слишком высокой загрузке CPU на *Сервере X*, вы сначала должны создать запись об узле сети для *Сервера X*, затем элемент данных для наблюдения за CPU, затем триггер, который сработает, если загрузка CPU будет слишком высокой, а затем действие, которое отправит вам e-mail. Хотя может показаться, что требуется слишком много шагов, при использовании шаблонов на самом деле это не так. Тем не менее, такое построение системы позволяет создавать очень гибкие инсталляции.

## 5 Что нового в Zabbix 7.4.0

См. breaking changes для этой версии.

Вложенное обнаружение на низком уровне

Теперь можно создавать многоуровневое обнаружение объектов с помощью прототипов обнаружения внутри правила обнаружения на низком уровне. Например, вы можете захотеть обнаружить все экземпляры базы данных на сервере баз данных, затем обнаружить табличные пространства для каждого экземпляра, а затем обнаружить таблицы для каждого табличного пространства.

Прототипы обнаружения — это вложенные правила обнаружения внутри ”родительского” правила обнаружения. У прототипов обнаружения есть собственные прототипы элемента данных, триггера, графика, узла сети и обнаружения.

Вложенный прототип обнаружения может использовать то же значение JSON, что и родительское правило, но затем использовать другой ”срез” данных из значения JSON.

Уровни вложенности для прототипов обнаружения не ограничены.

Прототипы узлов сети на обнаруженных узлах сети

Прототипы узлов сети теперь поддерживаются на обнаруженных узлах сети, что позволяет Zabbix автоматически обнаруживать и отслеживать сущности внутри других обнаруженных сущностей (например, гипервизоры, их виртуальные машины и контейнеры внутри этих виртуальных машин).

Вы можете создавать прототипы узлов сети на обнаруженных узлах сети, создавая правила низкоуровневого обнаружения с прототипами узлов сети или связывая шаблон с прототипами узлов сети. Кроме того, вы можете связать шаблон с прототипом узла сети, используемым для обнаружения, что приведет к тому, что обнаруженные узлы сети унаследуют прототипы узлов сети из шаблона.

Если ваша текущая конфигурация включает прототипы узлов сети, которые используют шаблоны, содержащие другие прототипы узлов сети, см. Примечания к обновлению.

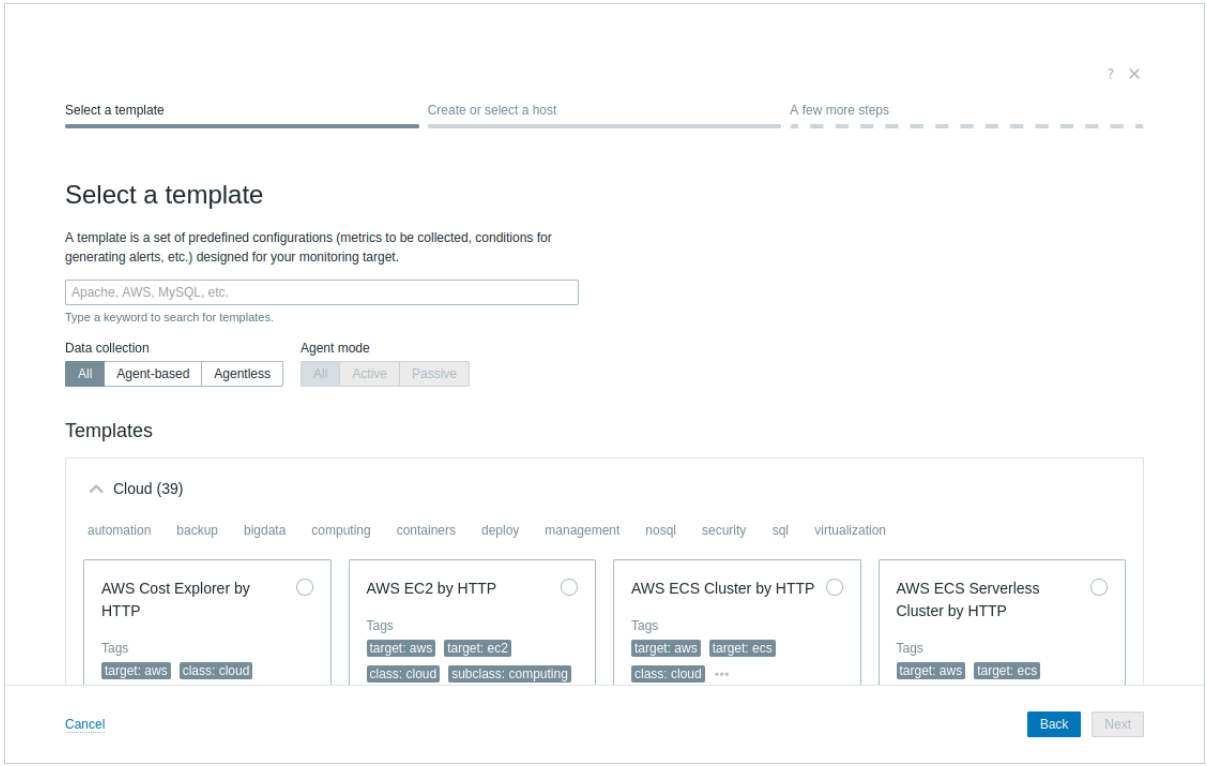
Аутентификация OAuth 2.0

Аутентификация OAuth 2.0 теперь поддерживается для протокола SMTP. Чтобы настроить аутентификацию OAuth, выберите ”OAuth” в параметре Authentication при настройке типа медиа для электронной почты, а затем укажите параметры для получения токенов OAuth.

Получение токенов OAuth имеет автоматизированные функции для почтовых провайдеров **Gmail**, **Gmail relay** и **Office365**. Требуется указать только значения параметров *Redirection endpoint*, *Client ID* и *Client secret*. Zabbix автоматически заполнит остальные необходимые значения (см. значения по умолчанию OAuth URL по провайдеру).

Мастер узла сети

Новый Мастер узла сети предоставляет пошаговый интерфейс с подсказками для настройки цели мониторинга (устройство, приложение, служба и т. д.) в Zabbix. Он упрощает настройку новых или существующих узлов сети, проводя пользователей через ключевые шаги, такие как выбор шаблона, установка агента Zabbix, добавление интерфейсов узла сети и многое другое.

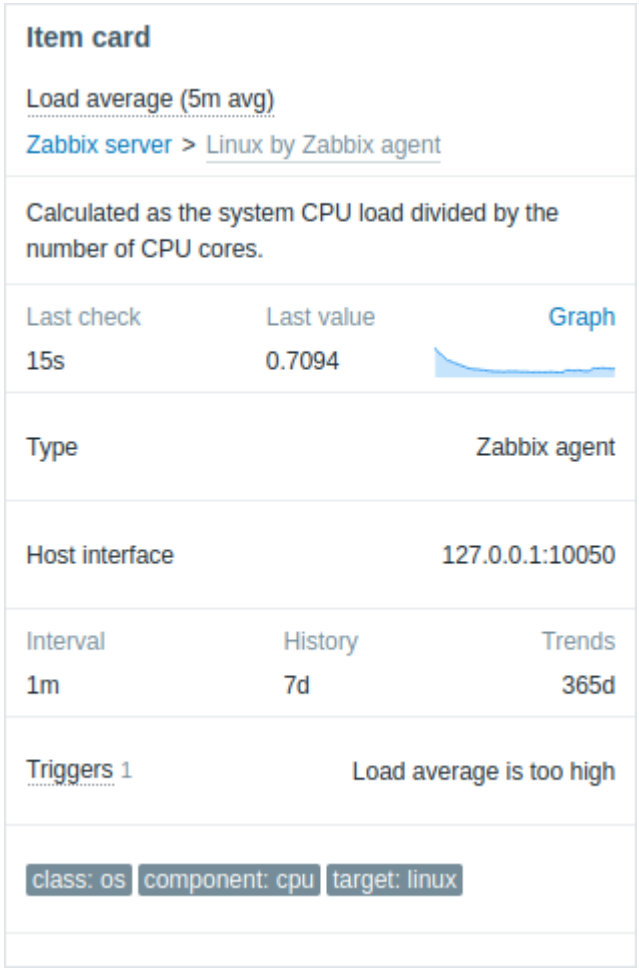



Доступ к Мастеру узла сети можно получить из раздела Сбор данных > Узлы сети или из меню узла сети в различных разделах веб-интерфейса.

Если вы обновляетесь с более ранней версии Zabbix, шаблоны необходимо обновить, чтобы они работали с Мастером узла сети. Инструкции см. в разделе Обновление шаблона.

**Виджеты** Карточка элемента данных

Виджет Карточка элемента данных был добавлен в набор виджетов панели мониторинга и предоставляет удобный способ быстро просматривать полную информацию об одном элементе данных.




История элемента данных

В виджете История элемента данных положение прокрутки теперь будет внизу, если новые значения настроены на отображение внизу. Это полезно для просмотра последних значений журналов.

Редактирование виджетов в реальном времени

При редактировании виджетов теперь можно предварительно просматривать изменения конфигурации виджета в реальном времени. Кроме того, формы настройки виджетов теперь можно перетаскивать, что позволяет перемещать их по мере необходимости.

Обратите внимание, что предварительный просмотр графика в форме виджета графика был удален.

**Элементы данных** Элемент данных ICMP ping с опцией повторной попытки

Для мониторинга доступности узла сети с помощью ICMP ping и возможностью изменять число повторных попыток добавлена новая простая проверка `icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]`.

**Функции** Отслеживание временных меток

Для отслеживания временных меток добавлены новые функции истории:

- `firstclock` - временная метка самого старого значения в пределах заданного периода оценки;
- `lastclock` - временная метка N-го по новизне значения в пределах заданного периода оценки;
- `logtimestamp` - временная метка сообщения журнала для N-го по новизне значения элемента данных журнала.

**Макросы** Отслеживание времени значения элемента данных

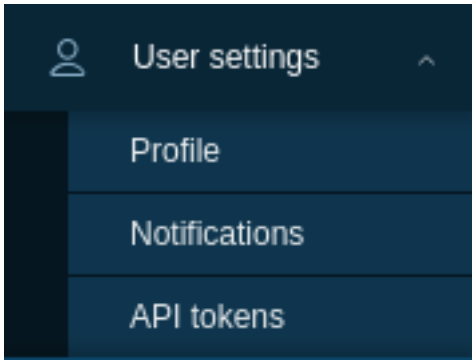
Для отслеживания времени значения элемента данных добавлены новые макросы:

| Макрос | Описание |
|---|---|
| {ITEM.LASTVALUE.AGE} | Время, прошедшее между последним получением значения элемента данных и вычислением макроса. |
| {ITEM.LASTVALUE.DATE} | Дата, когда было получено последнее значение элемента данных. |
| {ITEM.LASTVALUE.TIME} | Время, когда было получено последнее значение элемента данных. |

| Макрос | Описание |
|---|---|
| {ITEM.LASTVALUE.TIMESTAMP} | Метка времени, когда было получено последнее значение элемента данных. |
| {ITEM.VALUE.AGE} | Время, прошедшее между получением значения элемента данных и вычислением макроса. |
| {ITEM.VALUE.DATE} | Дата, когда было получено значение элемента данных. |
| {ITEM.VALUE.TIME} | Время, когда было получено значение элемента данных. |
| {ITEM.VALUE.TIMESTAMP} | Метка времени, когда было получено значение элемента данных. |

**Уведомления** Отдельный раздел меню для уведомлений пользователя

Для лучшей наглядности пользовательских медиа уведомления пользователя теперь имеют собственный раздел меню в *Настройках пользователя*.




Новый раздел *Уведомления* содержит две вкладки - *Медиа* и *Уведомления веб-интерфейса*, которые ранее были частью раздела профиля пользователя.

Управление собственными способами оповещения пользователя

По умолчанию всем пользователям теперь разрешено управлять собственными способами оповещения.

Однако права пользователя на изменение сведений о своих способах оповещения могут быть предоставлены или отозваны в зависимости от его роли пользователя (см. параметр *Создание и редактирование собственных способов оповещения*).

Кроме того, права пользователя Super admin на изменение сведений о способах оповещения других пользователей также могут быть предоставлены или отозваны в зависимости от его роли пользователя (см. параметр *Создание и редактирование способов оповещения пользователей*).

**Карты сети** Автоматическое скрытие подписей

Теперь можно настроить отображение подписей элементов карты/ссылок только при наведении на них курсора или при их выборе. Этот параметр помогает уменьшить визуальную перегруженность на картах, где много элементов расположены близко друг к другу.

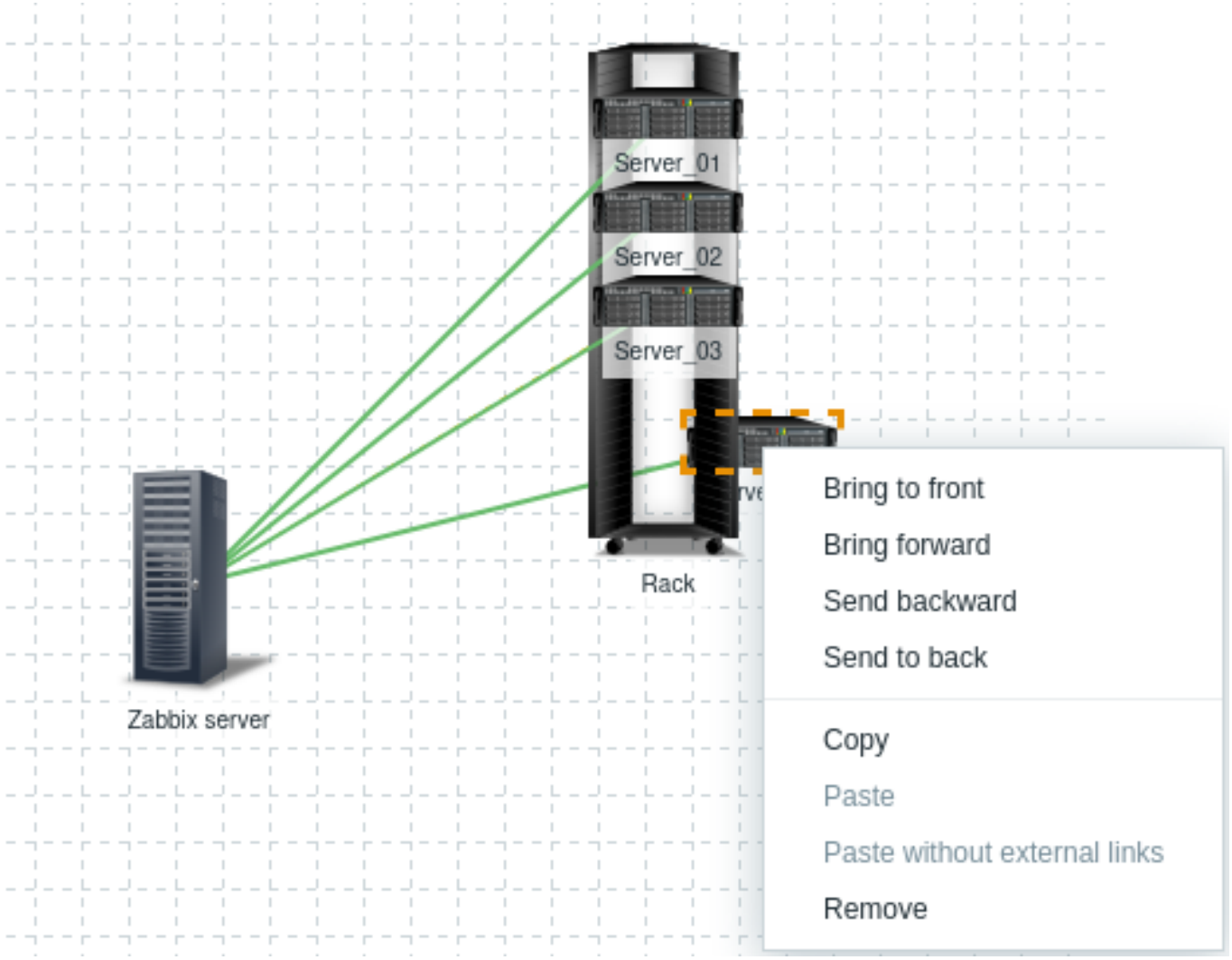
Автоматическое скрытие подписей можно настроить глобально для всех элементов карты/ссылок или для отдельного элемента карты/ссылки.

Пропорциональное масштабирование фоновых изображений

Теперь можно пропорционально масштабировать фоновые изображения, чтобы они соответствовали размеру карты.

Порядок элементов

Теперь можно переместить один элемент перед другим (или наоборот), щелкнув по элементу правой кнопкой мыши и выбрав параметры *Переместить вперед*/*Переместить на передний план* или *Переместить назад*/*Переместить на задний план*.



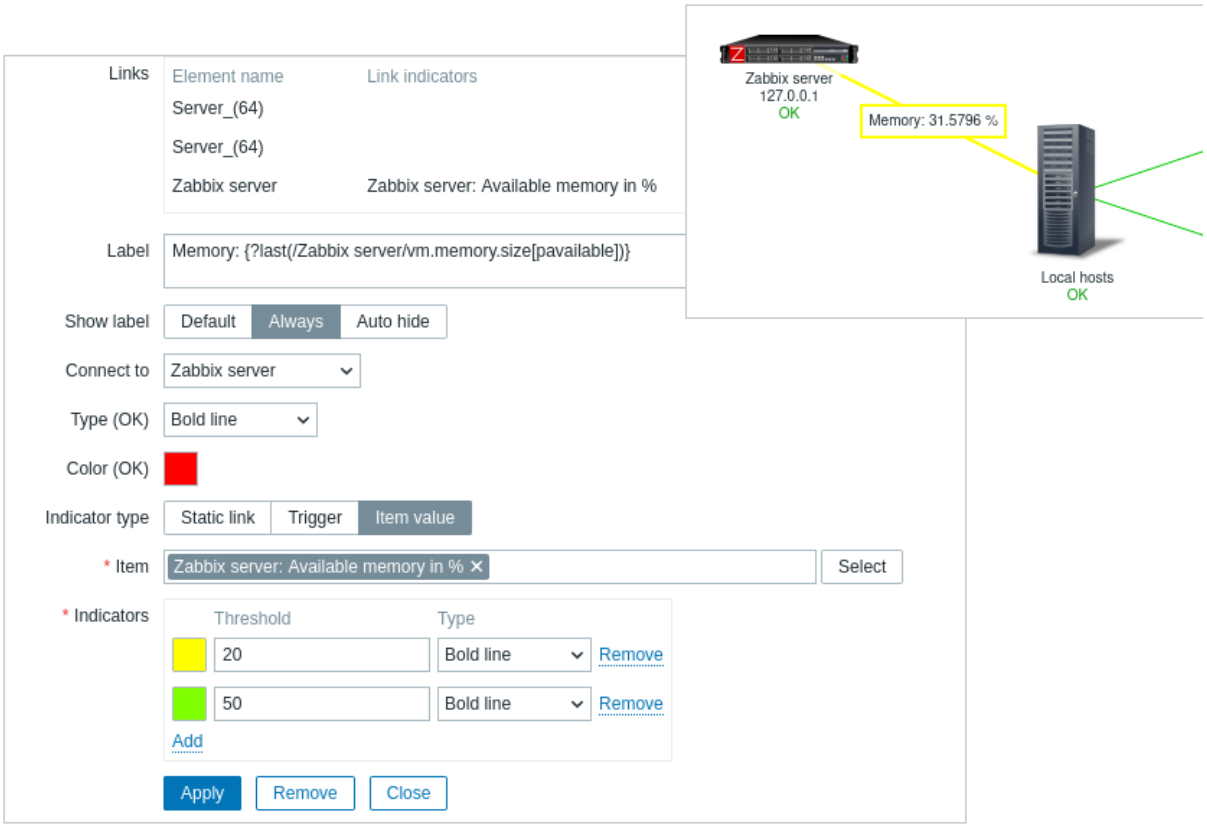

Кроме того, если элементы карты содержат и связь между ними, и подсветку, связь теперь будет отображаться под подсветкой.

Индикаторы ссылок на основе значения элемента данных

В предыдущих версиях можно было настраивать стиль и цвет ссылки, если какой-либо триггер переходил в состояние проблемы. Теперь аналогичная функциональность доступна на основе значений элементов данных.

Можно настроить стиль и цвет ссылки, если значение элемента данных:

- достигает заданного порога (для числовых элементов данных);
- соответствует регулярному выражению (для текстовых элементов данных).




**Процессы** Управление кэшем истории

В некоторых сценариях сбора данных отдельные элементы данных могут временно блокировать кэш истории сервера/прокси. Это может задерживать запись исторических данных в базу данных и замедлять работу системы. Чтобы упростить управление кэшем истории, были внедрены следующие улучшения:

- **Ручная очистка кэша:** Новая команда времени выполнения `history_cache_clear=target` для Zabbix сервер/прокси позволяет вручную очистить кэш истории для конкретного элемента данных по его ID.
- **Автоматическая очистка кэша:** Когда вы отключаете элемент данных, он немедленно удаляется из кэша истории (за исключением его последнего значения, которое сохраняется для журналов). Аналогично, когда вы отключаете узел сети, все его элементы данных удаляются из кэша истории (за исключением их последних значений).
- **Журналирование диагностики кэша:** Когда кэш истории заполнен, Zabbix сервер/прокси теперь записывает диагностическую информацию о кэше истории, начиная с DebugLevel=3 (ранее DebugLevel=4). Журнал содержит элементы данных с наибольшим количеством значений в кэше истории.

Для долгосрочной стабильности системы убедитесь, что сбор данных сбалансирован с доступными ресурсами (производительность базы данных, размер кэша, интервалы сбора, параметры элемента журнала и т. д.). Вы можете отслеживать кэш истории Zabbix с помощью внутреннего элемента zabbix[wcache]. Также можно рассмотреть возможность увеличения размера кэша истории для Zabbix сервер/прокси.

Статистика транзакций history syncer

Заголовок процесса history syncer теперь отображает подробную статистику о транзакциях history syncer для сервера и прокси Zabbix.

**Безопасность** TLS-шифрование между веб-интерфейсом и сервером

Теперь можно шифровать обмен данными между веб-интерфейсом Zabbix и сервером Zabbix с использованием TLS. Эта функция управляется новыми параметрами в конфигурации сервера (TLSListen, TLSFrontendAccept, TLSFrontendCertIssuer, TLSFrontendCertSubject, FrontendAllowedIP).

Разрешение макросов хранилища секретов сервером/прокси независимо

Теперь можно настроить, чтобы значения макросов секретов хранилища извлекались Zabbix server и Zabbix proxy независимо, если для параметра *Resolve secret vault macros by* установлено значение "Zabbix server and proxy".

**Плагины** Настройка пути к пользовательским запросам для загружаемых плагинов

Параметр `Plugins.*.CustomQueriesPath` в файлах конфигурации плагинов Zabbix agent 2 для MySQL, Oracle и PostgreSQL, а также `Plugins.MSSQL.CustomQueriesDir` для MSSQL, теперь имеет значение по умолчанию. Эта

функциональность по умолчанию отключена и может быть включена с помощью недавно добавленного параметра `Plugins.*.CustomQueriesEnabled`.

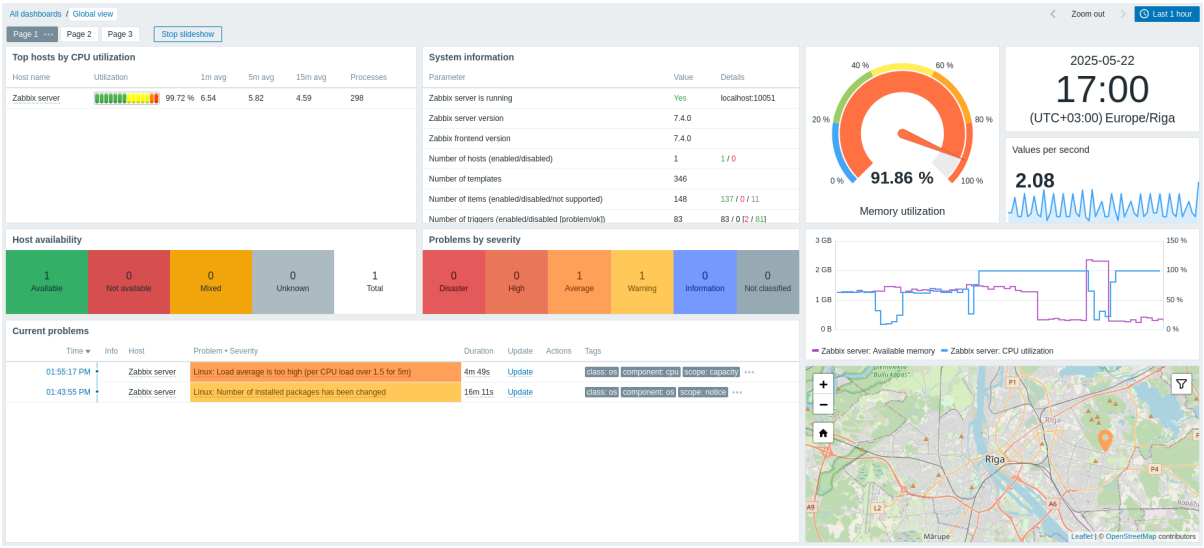
**Веб-интерфейс** Встроенная проверка в формах

Следующие формы веб-интерфейса теперь поддерживают встроенную проверку, отображая любые ошибки ввода сразу после заполнения полей:

- Шаблон конфигурации
- Узел сети конфигурации
- Элемент данных и прототип элемента данных конфигурации
- Триггер и прототип триггера конфигурации
- Модуль администрирования

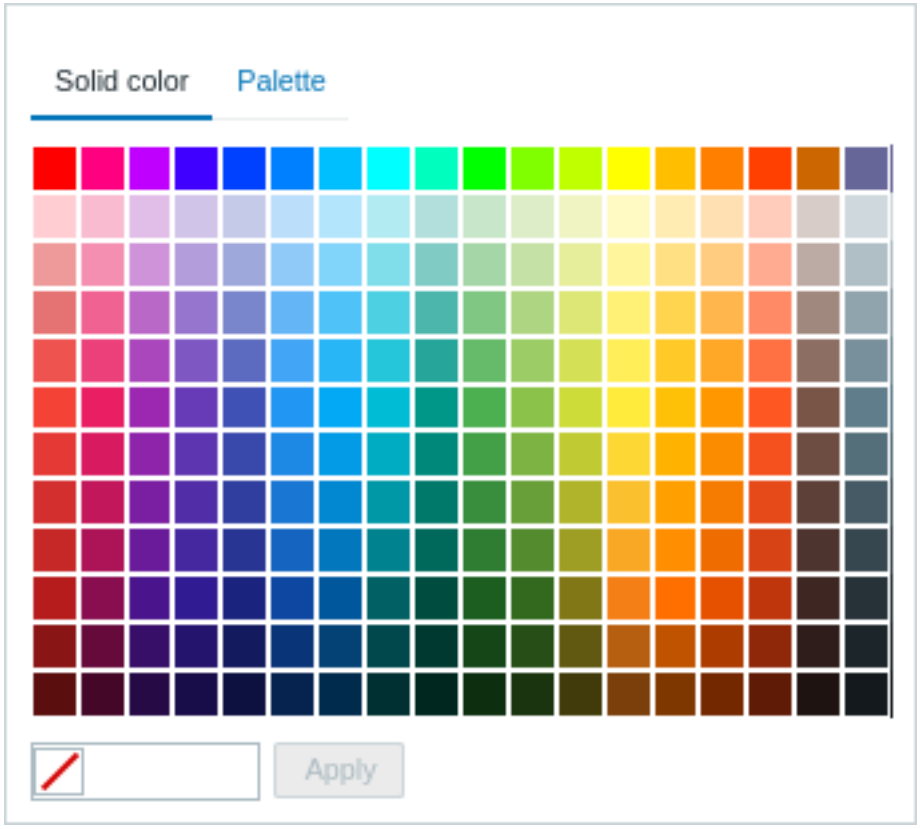
Обновлена панель по умолчанию

Панель dashboard по умолчанию *Global view* в новых установках Zabbix была обновлена и теперь включает последние виджеты панели.

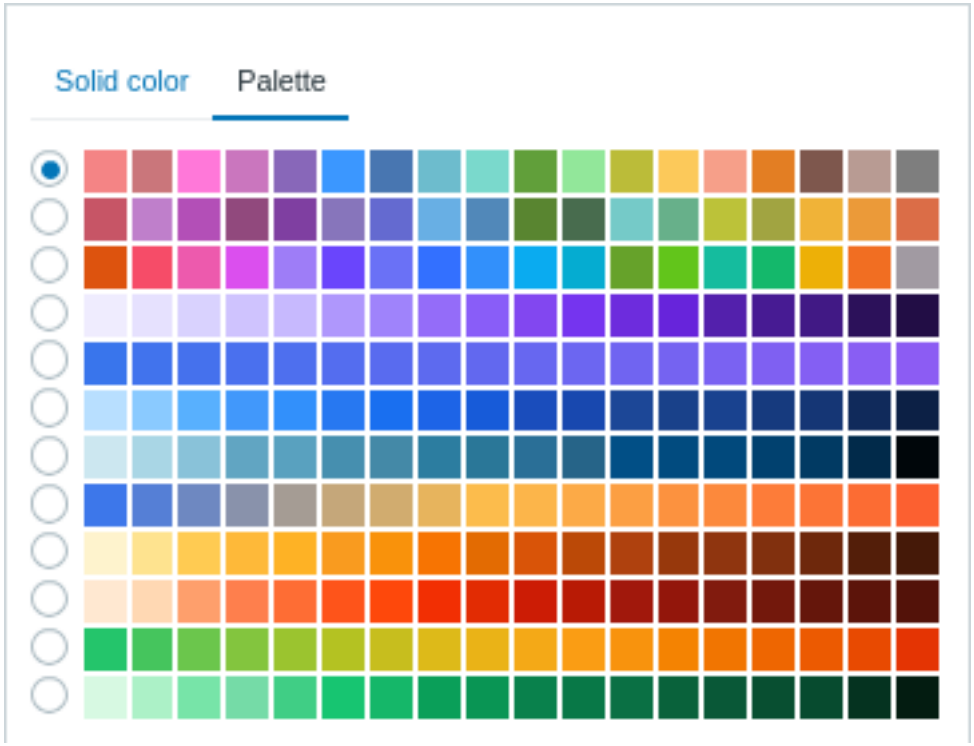



Улучшенный выбор цвета с поддержкой палитры

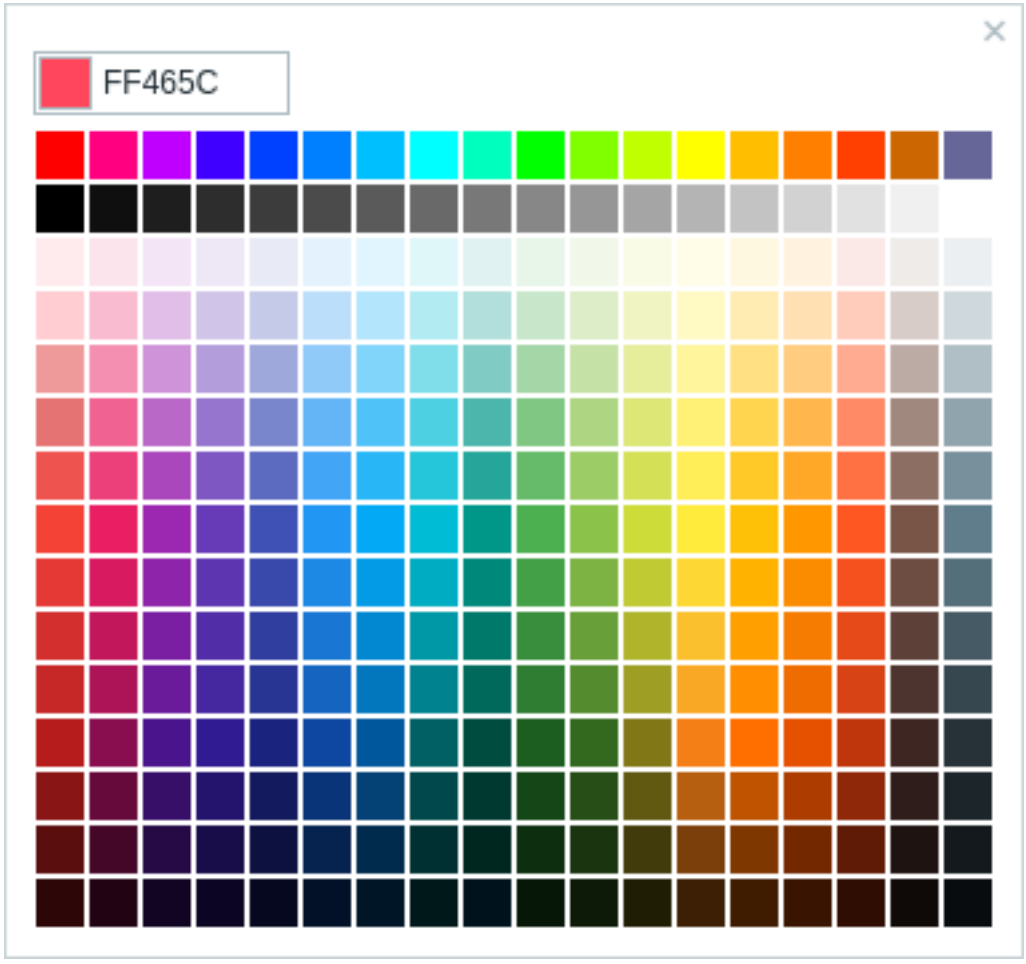
Средство выбора цвета в виджетах graph и pie chart было переработано для повышения удобства использования. Помимо сплошных цветов и ввода hex-кода, теперь можно перейти на вкладку *Palette* и выбрать цвет из предопределенных рядов цветов — каждая серия получает свой отдельный оттенок для более наглядного различия. Средство выбора также поддерживает полную навигацию с клавиатуры и проверку ввода в реальном времени.




B Zabbix 7.4 (*Solid color* tab)




B Zabbix 7.4 (*Color palette* tab)




До Zabbix 7.4

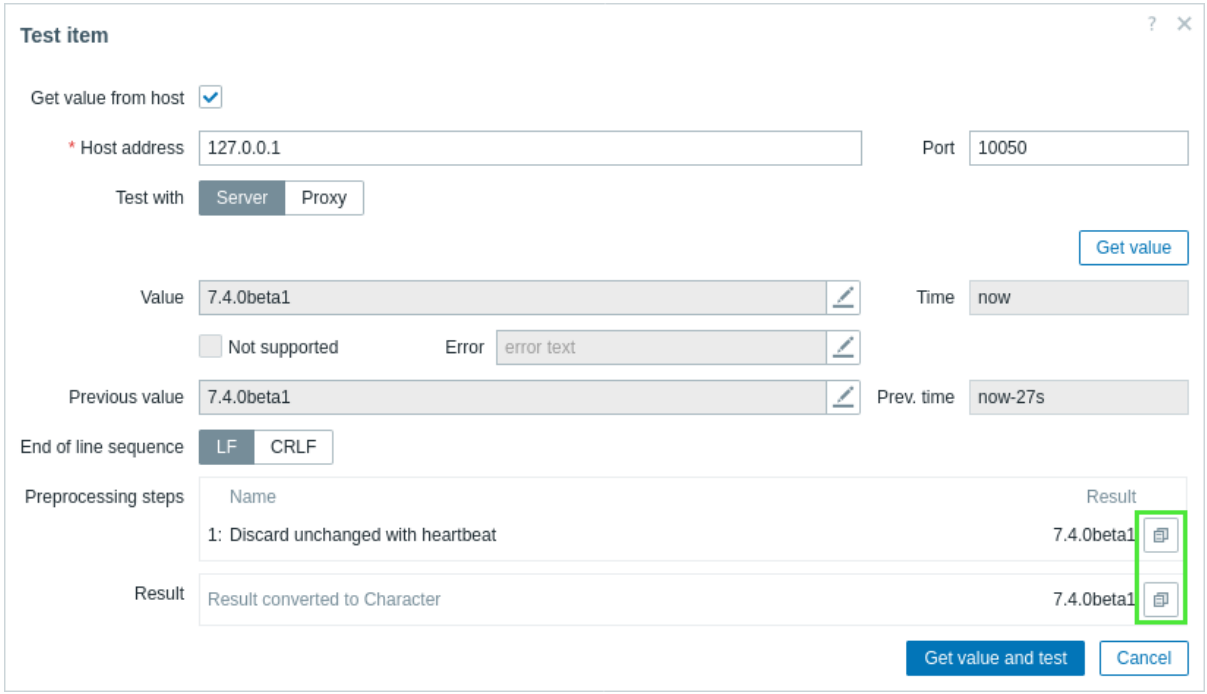
Модальные формы

Некоторые формы веб-интерфейса теперь открываются в модальных (всплывающих) окнах:

- График конфигурации
- Прототип графика конфигурации

Более удобное копирование результатов тестирования предварительной обработки

Теперь стало проще копировать значения в результатах тестирования предварительной обработки с помощью добавленной кнопки *Copy to clipboard*.




Обратите внимание, что аналогичная кнопка теперь заменяет ссылку *Copy* для созданных API tokens.

**Документация** Сводные страницы документации для минорных выпусков

Документация к выпускам минорных версий в рамках одного крупного выпуска Zabbix теперь будет собираться на отдельных страницах документации для новых возможностей и заметок по обновлению соответственно.

### 6 Что нового в Zabbix 7.4.x

Эта страница содержит сводную информацию о новых функциях, добавленных в минорные выпуски основной версии Zabbix.

См. также раздел Что нового основной версии.

Информацию об изменениях в существующих шаблонах и о новых шаблонах, охватывающих различные устройства, сервисы и решения партнёров, см. в разделе Изменения шаблонов.

**Что нового в Zabbix 7.4.12** Элементы тревог VMware включают имена сущностей

JSON, возвращаемый элементами `vmware.*.alarms.get[]`, теперь включает поле `name` внутри объекта `entity` для каждой тревоги. Это поле предоставляет имя затронутой сущности, а в запросе на изменение явно указывается фактическое имя узла сети для нативных ключей элементов `vmware.*.alarms.get`. Существующий объект `entity` уже возвращает идентификатор сущности, UUID и тип.

Если у вас есть скрипты, которые разбирают вывод элемента тревог VMware, возможно, вам потребуется обновить их, чтобы они обрабатывали дополнительное поле.

Поддержка MySQL 9.7

Максимальная поддерживаемая версия MySQL теперь — 9.7.X.

Поддержка TimescaleDB 2.27

Максимальная поддерживаемая версия для TimescaleDB теперь 2.27.X.

**Что нового в Zabbix 7.4.11** В этой минорной версии нет функциональных изменений.

**Что нового в Zabbix 7.4.10** Элементы тревог VMware возвращают сведения об объекте

Возвращаемый JSON элементов `vmware.*.alarms.get[]` теперь включает объект `entity` для каждой тревоги. Этот объект возвращает ID, UUID и тип затронутого объекта (узел сети, виртуальная машина или хранилище данных).

Если у вас есть скрипты, которые разбирают вывод элемента тревоги VMware, возможно, потребуется обновить их, чтобы они обрабатывали дополнительные поля.

**Что нового в Zabbix 7.4.9** Интеграции вебхуков

- Новая интеграция позволяет связать Zabbix с IBM Maximo Service Request с использованием медиа-типа Zabbix вебхук.
- Медиа-тип GLPI webhook был обновлен для поддержки RESTful API v2 с авторизацией OAuth2 (включена по умолчанию), необязательной поддержки устаревшего REST API v1 через флаг конфигурации, создания тикетов для событий проблем и настраиваемого пользователем сопоставления срочности.

Поддержка MySQL 9.6

Максимальная поддерживаемая версия MySQL теперь — 9.6.X.

Поддержка TimescaleDB 2.26

Максимальная поддерживаемая версия для TimescaleDB теперь 2.26.X.

Поддержка MariaDB 12.2

Максимальная поддерживаемая версия MariaDB теперь — 12.2.X.

Поддержка PHP 8.5

Максимальная поддерживаемая версия PHP теперь 8.5.

Поддержка SQLite 3.53

Максимальная поддерживаемая версия для SQLite теперь 3.53.

**Что нового в Zabbix 7.4.8** Поддержка TimescaleDB 2.25

Максимальная поддерживаемая версия для TimescaleDB теперь 2.25.X.

**Что нового в Zabbix 7.4.7** Поддержка TimescaleDB 2.24

Максимальная поддерживаемая версия для TimescaleDB теперь 2.24.X.

**Что нового в Zabbix 7.4.6** См. изменения, нарушающие совместимость, связанные с ограниченными правами при создании файлов журнала.

Поддержка TimescaleDB 2.23

Максимальная поддерживаемая версия для TimescaleDB теперь 2.23.X.

Поддержка MySQL 9.5

Максимальная поддерживаемая версия для MySQL теперь 9.5.X.

Расширенные параметры командной строки для MSI-установщика агента

Пакеты MSI-установщика Zabbix agent (как для Zabbix agent, так и для Zabbix agent 2) теперь поддерживают следующие параметры при выполнении установки из командной строки:

- `HOSTNAMEITEM` - указывает элемент данных агента Zabbix, используемый для получения имени узла сети
- `UNSAFEUSERPARAMETERS` - позволяет передавать в аргументах для пользовательских параметров все символы (например, ' " ` * ?)
- `STARTAGENTS` - указывает количество предварительно созданных экземпляров `zabbix_agentd`, обрабатывающих пассивные проверки
- `DONOTSTART` - предотвращает запуск службы агента Zabbix MSI-установщиком
- `SOURCEIP` - указывает исходный IP-адрес для исходящих подключений к серверу Zabbix/прокси или для подключений, выполняемых при запуске определенных элементов данных (например, `web.page.get`, `net.tcp.port`)

Максимальное время ожидания для элементов данных ICMP ping

Для элементов данных icmpping* теперь применяется жестко заданное максимальное возможное время ожидания в 600 секунд.

**Что нового в Zabbix 7.4.5** В этой минорной версии нет никаких функциональных изменений.

**Что нового в Zabbix 7.4.4** **Внимание!** Обновление до этой версии не рекомендуется из-за ошибок, связанных с графиками и плагином MySQL для агента Zabbix agent 2.

Поддержка PostgreSQL 18

Максимальная поддерживаемая версия PostgreSQL теперь — 18.X.

Поддержка TimescaleDB 2.22

Максимальная поддерживаемая версия для TimescaleDB теперь — 2.22.X.

Плагин Redis — поддержка TLS и проверка при запуске

Для плагина Redis для Zabbix agent 2 добавлена поддержка TLS.

Была реализована проверка конфигурации TLS плагина при запуске, а также улучшены сообщения проверки и ошибок. Некорректная логика конфигурации (например, использование типа подключения `verify_full` без указания `TLSCAFile`) может помешать запуску Zabbix agent 2.

**Что нового в Zabbix 7.4.3** Поддержка имени TNS для плагина Oracle

В плагин Oracle в Zabbix агент 2 добавлена нативная поддержка имени TNS Oracle. Ключ TNS или значение TNS можно указать на уровне [параметров] ключа элемента данных (/manual/config/items/itemtypes/zabbix_agent/zabbix_agent2#oracleping) или [конфигурации] плагина (/manual/appendix/config/zabbix_agent2_plugins/oracle_plugin) (для именованной сессии или параметров по умолчанию).

Поддержка имени TNS позволяет задавать описание подключения в кластерной среде БД с одним узлом сети Zabbix. Дополнительные сведения см. в README плагина Oracle.

Увеличено максимальное время ожидания для zabbix_get и zabbix_js

Максимальное значение параметра `timeout` утилит командной строки zabbix_get и zabbix_js увеличено до 600 секунд.

Поддержка MariaDB 12.0

Максимальная поддерживаемая версия для MariaDB теперь 12.0.X.

Подсветка всей строки в представлении ”Проблемы” и виджете ”Проблемы”

Параметр *Подсветка всей строки* теперь доступен в некратком представлении Мониторинг > Проблемы и в виджете панели Проблемы.

**Что нового в Zabbix 7.4.2** Поддержка дополнительных ролей для плагина Oracle

Элементы данных плагина Oracle для Zabbix агента 2 теперь в параметрах подключения поддерживают дополнительные роли, такие как SysBACKUP, SysDG, SysKM или SysRAC.

Автоматическая перезагрузка учётных данных SNMPv3

Проверки элементов данных walk[] и get[] агента SNMP теперь автоматически перезагружают обновлённые настройки аутентификации и конфиденциальности SNMPv3 при изменении соответствующего интерфейса SNMPv3. Команда ручной перезагрузки кэша -R snmp_cache_reload остаётся доступной для устранения неполадок и для устройств, которые не полностью соответствуют спецификациям SNMPv3.

smart.disk.discovery — новый параметр **тип**

Элемент данных smart.disk.discovery (плагин S.M.A.R.T. Zabbix агента 2) теперь принимает необязательный параметр **тип** для указания значения дисков для сканирования.

**Что нового в Zabbix 7.4.1** Поддержка TimescaleDB 2.21

Максимальная поддерживаемая версия для TimescaleDB теперь 2.21.X.

Диагностическая информация о предварительной обработке на сервере/прокси

В раздел предварительной обработки диагностической информации о работе сервера/прокси были добавлены разделы с верхними элементами, отсортированные по количеству полученных значений и размеру. Идентификаторы верхних элементов с наибольшим временем обработки выделяются.

Эта информация доступна при запуске сервера Zabbix или прокси с параметром diaginfo runtime, например `zabbix_server -R diaginfo=preprocessing`.

Обход менеджера предобработки для элементов данных без предварительной обработки

Чтобы снизить нагрузку на обработку, значения элементов данных, не имеющих ни предобработки, ни зависимых элементов данных, добавляются непосредственно в кэш истории или отправляются в менеджер LLD (см. Подробности

предварительной обработки). Ранее все значения элементов данных проходили через менеджер предобработки независимо от конфигурации предобработки.

Обновлена интеграция вебхука GLPI

Тип медиа GLPI теперь поддерживает аутентификацию с использованием токенов приложения.

## 2. Определения

**Обзор** В этом разделе вы можете узнать значение некоторых терминов, которые часто используются в Zabbix.

**Определения** ***узел сети***

*- любое физическое или виртуальное устройство, приложение, служба или любая другая логически связанная совокупность контролируемых параметров.*

***группа узлов сети***

*- логическая группировка узлов сети. Группы узлов сети используются при назначении прав доступа к узлам сети для разных групп пользователей.*

***элемент данных***

*- определенная порция данных, которую вы хотите получать от узла сети, метрика данных.*

***предобработка значения***

*- преобразование полученного значения метрики* перед сохранением в базу данных.

***триггер***

*- логическое выражение, которое определяет порог проблемы и используется для "оценки" данных, полученных в элементах данных.*

Когда полученные данные выше порога, триггеры переходят из состояния 'Ok' в состояние 'Problem'. Когда полученные данные ниже порога, триггеры остаются в состоянии 'Ok' или возвращаются в него.

***шаблон***

*- набор сущностей (элементы данных, триггеры, графики, правила обнаружения низкого уровня, веб-сценарии), готовый к применению к одному или нескольким узлам сети.*

Задача шаблонов - ускорить развертывание задач мониторинга на узле сети; а также упростить массовое внесение изменений в задачи мониторинга. Шаблоны связываются напрямую с отдельными узлами сети.

***группа шаблонов***

*- логическая группировка шаблонов. Группы шаблонов используются при назначении прав доступа к шаблонам для разных групп пользователей.*

***событие***

*- единичный факт возникновения чего-либо, заслуживающего внимания, например изменения состояния триггера или выполнения обнаружения/авторегистрации агента.*

***тег события***

*- предопределенный маркер для события.* Он может использоваться в корреляции событий, грануляции прав доступа и т. д.

***корреляция событий***

*- метод гибкого и точного сопоставления проблем с их устранением.*

Например, вы можете определить, что проблема, сообщенная одним триггером, может быть устранена другим триггером, который даже может использовать другой метод сбора данных.

***проблема***

*- триггер, находящийся в состоянии "Problem".*

***обновление проблемы***

*- возможности управления проблемами, предоставляемые Zabbix, такие как добавление комментария, подтверждение, изменение серьезности или ручное закрытие.*

***действие***

*- предопределенный способ реагирования на событие.*

Действие состоит из операций (например, отправки уведомления) и условий (*когда* выполняется операция)

***эскалация***

*- пользовательский сценарий выполнения операций в рамках действия; последовательность отправки уведомлений/выполнения удаленных команд.*

***макрос***

*- заполнитель переменной, который подставляет конкретное значение в зависимости от контекста.*

***средство доставки***

*- способ доставки уведомлений; канал доставки.*

***уведомление***

*- сообщение о некотором событии, отправленное пользователю через выбранный канал доставки.*

***удаленная команда***

*- предопределенная команда, которая автоматически выполняется на контролируемом узле сети при выполнении некоторого условия.*

***веб-сценарий***

*- один или несколько HTTP-запросов для проверки доступности веб-сайта.*

***веб-интерфейс***

*- веб-интерфейс, предоставляемый Zabbix.*

***панель***

*- настраиваемый раздел веб-интерфейса, отображающий сводки и визуализации* важной информации в визуальных элементах, называемых виджетами.

***виджет***

*- визуальный элемент, отображающий информацию определенного типа и источника* (сводку, карту, график, часы и т. д.), используемый на панели.

***Zabbix API***

*- Zabbix API позволяет использовать протокол JSON RPC для создания, обновления и получения объектов Zabbix (таких как узлы сети, элементы данных, графики и другие) или выполнения любых других пользовательских задач.*

***сервер Zabbix***

*- центральный процесс программного обеспечения Zabbix, который выполняет мониторинг, взаимодействует с прокси Zabbix и агентами, вычисляет триггеры, отправляет уведомления; центральное хранилище данных.*

***прокси Zabbix***

*- процесс, который может собирать данные от имени сервера Zabbix, снимая часть нагрузки по обработке с сервера.*

***агент Zabbix***

*- процесс, развертываемый на контролируемых объектах для активного мониторинга локальных ресурсов и приложений.*

***Zabbix agent 2***

*- агент Zabbix нового поколения для активного мониторинга локальных ресурсов и приложений, позволяющий использовать пользовательские плагины для мониторинга.*

> **Attention:**
> Поскольку Zabbix agent 2 разделяет с Zabbix agent многие функции, термин ”Zabbix agent” в документации обозначает оба варианта - Zabbix agent и Zabbix agent 2, если функциональное поведение одинаково. Zabbix agent 2 отдельно упоминается только там, где его функциональность отличается.

***шифрование***

- *поддержка зашифрованных коммуникаций между компонентами Zabbix (сервер, прокси, агент, утилиты zabbix_sender и zabbix_get)* с использованием протокола Transport Layer Security (TLS).

***авторегистрация агента***

- *автоматизированный процесс, при котором сам агент Zabbix регистрируется* как узел сети и начинает мониторинг.

***сетевое обнаружение***

- *автоматическое обнаружение сетевых устройств*.

***обнаружение низкого уровня***

- *автоматическое обнаружение сущностей низкого уровня на конкретном устройстве* (например, файловых систем, сетевых интерфейсов и т. д.).

***правило обнаружения низкого уровня***

- *набор определений для автоматического обнаружения сущностей низкого уровня* на устройстве.

***прототип элемента данных***

- *метрика с определенными параметрами в виде переменных, готовая к обнаружению низкого уровня*. После обнаружения низкого уровня переменные автоматически подставляются вместо реальных обнаруженных параметров, и метрика автоматически начинает собирать данные.

***прототип триггера***

- *триггер с определенными параметрами в виде переменных, готовый к обнаружению низкого уровня*. После обнаружения низкого уровня переменные автоматически подставляются вместо реальных обнаруженных параметров, и триггер автоматически начинает оценивать данные.

*Прототипы* некоторых других сущностей Zabbix также используются при обнаружении низкого уровня - прототипы графиков, прототипы узлов сети, прототипы групп узлов сети.

## 3. Процессы Zabbix

**Обзор** Zabbix состоит из нескольких процессов, которые работают вместе для мониторинга систем и отсылки оповещений. Каждый процесс выполняет определённую роль в инфраструктуре мониторинга.

**Основные процессы**

- Сервер — центральный процесс, который выполняет опрос и сбор данных, вычисляет триггеры и отсылает оповещения пользователям.
- Агент — собирает данные с наблюдаемых узлов сети.
- Агент 2 — агент нового поколения с архитектурой плагинов и расширенными возможностями.
- Прокси — собирает данные мониторинга с наблюдаемых устройств и отправляет информацию на Zabbix сервер.
- Java gateway — обеспечивает мониторинг Java-приложений.

**1. Сервер**

Обзор

Сервер Zabbix — это центральный процесс программного обеспечения Zabbix.

Сервер выполняет опрос и прием данных, вычисляет триггеры, отправляет уведомления пользователям. Это центральный компонент, которому агенты Zabbix и прокси передают данные о доступности и целостности систем. Сервер также может удаленно проверять сетевые службы (например, веб-серверы и почтовые серверы) с помощью простых проверок служб.

Сервер является центральным хранилищем, в котором сохраняются все конфигурационные, статистические и эксплуатационные данные, и именно он в Zabbix активно оповещает администраторов, когда возникают проблемы в любой из контролируемых систем.

Работа базового сервера Zabbix разделена на три отдельных компонента: сервер Zabbix, веб-интерфейс и хранилище базы данных.

Вся информация о конфигурации Zabbix хранится в базе данных, с которой взаимодействуют и сервер, и веб-интерфейс. Например, когда вы создаете новый элемент данных с помощью веб-интерфейса (или API), он добавляется в таблицу items в базе данных. Затем примерно раз в минуту сервер Zabbix запрашивает таблицу items, чтобы получить список активных элементов данных, который затем сохраняется в кэше внутри сервера Zabbix. Именно поэтому изменения, внесенные в веб-интерфейсе Zabbix, могут отображаться в разделе последних данных только через две минуты.

Запущенный сервер

Если установлен из пакета

Zabbix сервер работает как демон. Его можно запустить, выполнив:

```
systemctl start zabbix-server
```

Эта команда будет работать на большинстве систем GNU/Linux. На других системах вам, возможно, потребуется выполнить:

```
/etc/init.d/zabbix-server start
```

Аналогично, для остановки/перезапуска/просмотра состояния используйте следующие команды:

```
systemctl stop zabbix-server
systemctl restart zabbix-server
systemctl status zabbix-server
```

Запуск вручную

Если указанный выше способ не работает, необходимо запустить его вручную. Найдите путь к бинарному файлу zabbix_server и выполните:

```
zabbix_server
```

С Zabbix server можно использовать следующие параметры командной строки:

```
-c --config <file>                              (       : /usr/local/etc/zabbix_server.conf)
-f --foreground                       Zabbix server
-R --runtime-control <option>
-T --test-config
-h --help
-V --version
```

Примеры запуска Zabbix server с параметрами командной строки:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf
zabbix_server --help
zabbix_server -V
```

Управление во время работы

Параметры управления во время работы:

| Option | Description | Target |
|---|---|---|
| config_cache_reload | Перезагрузить кэш конфигурации. Игнорируется, если кэш в данный момент загружается. | |
| history_cache_diff[=<target>] | Отобразить кэш истории для элемента данных, указанного по его ID. Затрагивает все значения элемента данных, кроме первого и последнего значения. | **target** - ID элемента данных |
| diaginfo[=<section>] | Собрать диагностическую информацию в файле журнала сервера. | historycache - статистика кэша истории;<br>valuecache - статистика кэша значений;<br>preprocessing - статистика менеджера предварительной обработки;<br>alerting - статистика менеджера оповещений;<br>lld - статистика менеджера LLD;<br>locks - список мьютексов (пусто в системах *BSD*);<br>connector - статистика для коннекторов с наибольшей очередью. |

| Option | Description | Target |
|---|---|---|
| `ha_status` | Записать в журнал состояние кластера высокой доступности (HA). | |
| `ha_remove_node=target` | Удалить узел высокой доступности (HA), указанный по имени или ID.<br>Обратите внимание, что активные/резервные узлы удалить нельзя. | **target** - имя или ID узла (можно получить, выполнив `ha_status`). |
| `ha_set_failover_delay=delay` | Установить задержку переключения высокой доступности (HA).<br>Поддерживаются суффиксы времени, например 10s, 1m. | |
| `proxy_config_cache_reload[=target]` | Перезагрузить конфигурации прокси. | **target** - список имен прокси, разделенных запятыми.<br>Если target не указан, конфигурация будет перезагружена для всех прокси. |
| `secrets_reload` | Перезагрузить секреты из Vault. | |
| `service_cache_reload` | Перезагрузить кэш менеджера служб. | |
| `snmp_cache_reload` | Перезагрузить кэш SNMP — очистить свойства SNMP engine (engine time, engine boots, engine id, credentials) для всех узлов сети.<br>Используйте для принудительной глобальной очистки кэша при устранении проблем с SNMP. | |
| `housekeeper_execute` | Запустить процедуру housekeeping.<br>Игнорируется, если процедура housekeeping уже выполняется. | |
| `trigger_housekeeper_execute` | Запустить процедуру housekeeping для триггеров.<br>Игнорируется, если процедура housekeeping для триггеров уже выполняется.<br>Пока процедура housekeeping для триггеров не запущена, проблемы, вызванные триггерами, которые к этому моменту были удалены, все еще могут создавать проблемы служб и назначать их службам. Если в вашей конфигурации много правил расчета состояния служб, основанных на часто обнаруживаемых/необнаруживаемых триггерах, рассмотрите возможность увеличения частоты процедуры housekeeping путем настройки серверного параметра конфигурации `ProblemHousekeepingFrequency`. | |
| `log_level_increase[=target]` | Увеличить уровень журналирования, затрагивает все процессы, если target не указан.<br>Не поддерживается в системах *BSD*. | **process type** - все процессы указанного типа (например, `poller`).<br>См. все типы процессов сервера.<br>**process type,N** - тип процесса и номер (например, `poller,3`).<br>**pid** - идентификатор процесса (1 to 65535). Для больших значений укажите target как 'process type,N'. |
| `log_level_decrease[=target]` | Уменьшить уровень журналирования, затрагивает все процессы, если target не указан.<br>Не поддерживается в системах *BSD*. | |

| Option | Description | Target |
|---|---|---|
| prof_enable[=<target>] | Включить профилирование.<br>Затрагивает все процессы, если target не указан.<br>Включенное профилирование предоставляет сведения обо всех rwlocks/mutexes по имени функции. | **process type** - все процессы указанного типа (например, history syncer)<br>Поддерживаемые типы процессов в качестве целей профилирования: 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, preprocessing worker, proxy poller, self-monitoring, service manager, snmp trapper, task manager, timer, trapper, unreachable poller, vmware collector.<br>**process type,N** - тип процесса и номер (например, history syncer,1).<br>**pid** - идентификатор процесса (1 to 65535). Для больших значений укажите target как 'process type,N'.<br>**scope** - rwlock, mutex, processing можно использовать с типом процесса и номером (например, history syncer,1,processing) или со всеми процессами данного типа (например, history syncer,rwlock). |
| prof_disable[=<target>] | Отключить профилирование.<br>Затрагивает все процессы, если target не указан. | **process type** - все процессы указанного типа (например, history syncer).<br>Поддерживаемые типы процессов в качестве целей профилирования: см. prof_enable.<br>**process type,N** - тип процесса и номер (например, history syncer,1).<br>**pid** - идентификатор процесса (1 to 65535). Для больших значений укажите target как 'process type,N'. |

Пример использования управления во время выполнения для перезагрузки кэша конфигурации сервера:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R config_cache_reload
```

Примеры использования управления во время выполнения для перезагрузки конфигурации прокси:

```
### :
zabbix_server -R proxy_config_cache_reload

### Proxy1 Proxy2:
zabbix_server -R proxy_config_cache_reload=Proxy1,Proxy2
```

Пример использования управления во время выполнения для очистки кэша истории для элемента данных:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R history_cache_clear=42243
```

Примеры использования управления во время выполнения для сбора диагностической информации:

```
### :
zabbix_server -R diaginfo

### :
zabbix_server -R diaginfo=historycache
```

Пример использования управления во время выполнения для перезагрузки SNMP-кэша:

```
zabbix_server -R snmp_cache_reload
```

**Attention:**
Когда интерфейс SNMPv3 обновляется через веб-интерфейс Zabbix, Zabbix в большинстве случаев автоматически перезагружает новые учетные данные SNMPv3 для этого интерфейса; используйте `-R snmp_cache_reload` только если опрос по-прежнему не выполняется после изменения учетных данных (например, из-за несоответствий engineBoots/engineID или устройств, не соответствующих RFC), либо когда требуется принудительно очистить глобальный SNMP-кэш для устранения неполадок.

Пример использования управления во время выполнения для запуска housekeeper:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R housekeeper_execute
```

Примеры использования управления во время выполнения для изменения уровня журнала:

```
###                             :
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase

###                               poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2

###                            PID 1234:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234

###                               http poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_decrease="http poller"
```

Пример установки задержки переключения при отказе HA на минимальное значение 10 секунд:

```
zabbix_server -R ha_set_failover_delay=10s
```

Пользователь процесса

Zabbix сервер спроектирован для запуска от непривилегированного пользователя (non-root). Он будет работать от любого непривилегированного пользователя, от которого был запущен. Таким образом, вы можете запускать сервер от имени любого непривилегированного пользователя, без каких либо последствий.

Если вы попытаетесь запустить сервер от имени «root», он сразу переключится на пользователя «zabbix», который должен присутствовать в вашей системе. Единственный способ запустить сервер от пользователя «root» — соответствующим образом отредактировать параметр «AllowRoot» в файле конфигурации сервера.

Если Zabbix сервер и агент работают на одном сервере, то рекомендуется использовать разных пользователей для запуска сервера и для запуска агента. В противном случае, если сервер и агент запущены под одним пользователем, агент будет иметь доступ к файлу конфигурации сервера и любой пользователь с правами Администратора в Zabbix может с лёгкостью получить, например, пароль от базы данных.

Файл конфигурации

См. параметры файла конфигурации для получения подробной информации о настройке `zabbix_server`.

Скрипты запуска

Скрипты используются для автоматического запуска/остановки процессов Zabbix при включении/выключении системы. Скрипты располагаются в директории misc/init.d.

Типы процессов и потоков сервера

- `agent poller` - асинхронный процесс опроса для пассивных проверок с рабочим потоком;
- `alert manager` - менеджер очереди оповещений;
- `alert syncer` - процесс записи оповещений в БД;
- `alerter` - процесс отправки уведомлений;
- `availability manager` - процесс обновления доступности узлов сети;
- `browser poller` - процесс опроса для проверок элементов данных браузера;
- `configuration syncer` - процесс управления кэшем данных конфигурации в памяти;
- `configuration syncer worker` - процесс разрешения и синхронизации значений пользовательских макросов в именах элементов данных;
- `connector manager` - процесс-менеджер для коннекторов;
- `connector worker` - процесс обработки запросов от менеджера коннекторов;
- `discovery manager` - процесс-менеджер для обнаружения устройств;
- `discovery worker` - процесс обработки задач обнаружения от менеджера обнаружения;
- `escalator` - процесс эскалации действий;

- `ha manager` - процесс управления высокой доступностью;
- `history poller` - процесс обработки вычисляемых проверок, требующих подключения к базе данных;
- `history syncer` - процесс записи истории в БД;
- `housekeeper` - процесс удаления устаревших данных (истории и трендов элементов данных, пользовательских сессий, событий и т. д.), а также данных, оставшихся после удаленных объектов;
- `http agent poller` - асинхронный процесс опроса для HTTP-проверок с рабочим потоком;
- `http poller` - процесс опроса для веб-мониторинга;
- `icmp pinger` - процесс опроса для проверок icmpping;
- `internal poller` - процесс опроса для внутренних проверок;
- `ipmi manager` - менеджер процесса опроса IPMI;
- `ipmi poller` - процесс опроса для проверок IPMI;
- `java poller` - процесс опроса для проверок Java;
- `lld manager` - процесс-менеджер задач низкоуровневого обнаружения;
- `lld worker` - рабочий процесс задач низкоуровневого обнаружения;
- `odbc poller` - процесс опроса для проверок ODBC;
- `poller` - обычный процесс опроса для пассивных проверок;
- `preprocessing manager` - менеджер задач предварительной обработки с потоками рабочих процессов предварительной обработки;
- `preprocessing worker` - поток для предварительной обработки данных;
- `proxy poller` - процесс опроса для пассивных прокси;
- `proxy group manager` - менеджер балансировки нагрузки и высокой доступности прокси;
- `report manager`- менеджер задач планового формирования отчетов;
- `report writer` - процесс формирования плановых отчетов;
- `self-monitoring` - процесс сбора внутренних статистических данных сервера;
- `service manager` - процесс управления сервисами путем получения информации о проблемах, тегах проблем и восстановлении проблем от history syncer, task manager и alert manager;
- `snmp poller` - асинхронный процесс опроса для проверок SNMP с рабочим потоком (`walk[OID]` и `get[OID]` только для элементов данных);
- `snmp trapper` - trapper для SNMP-ловушек;
- `task manager` - процесс удаленного выполнения задач, запрошенных другими компонентами (например, закрытие проблемы, подтверждение проблемы, немедленная проверка значения элемента данных, функциональность удаленной команды);
- `timer` - таймер для обработки обслуживаний;
- `trapper` - trapper для активных проверок, ловушек, связи с прокси;
- `trigger housekeeper` - процесс удаления проблем и событий, сгенерированных триггерами, которые были впоследствии удалены;
- `unreachable poller` - процесс опроса для недоступных устройств;
- `vmware collector` - сборщик данных VMware, отвечающий за сбор данных из служб VMware.

Файл журнала сервера можно использовать для наблюдения за этими типами процессов.

Начиная с Zabbix 7.4.6, файл журнала сервера создается с правами чтения и записи только для владельца файла. Кроме того, файл доступен для чтения группе владельца. Все остальные права запрещены.

Различные типы процессов сервера Zabbix можно отслеживать с помощью внутреннего элемента данных `zabbix[process,<type>,`

Статистика транзакций history syncer

Заголовок процесса history syncer отображает подробную статистику по транзакциям history syncer:

```
205182 ?        S      0:00  zabbix_server: history syncer #2 [processed 0 values, 0+0 triggers in 0.00002
205183 ?        S      0:00  zabbix_server: history syncer #3 [processed 18 values, 7+0 triggers in 0.0026
205184 ?        S      0:00  zabbix_server: history syncer #4 [processed 0 values, 0+0 triggers in 0.00002
```

В "A+B triggers":

- A - триггеры, обработанные из-за значений истории;
- B - триггеры, обработанные из-за таймеров.

Временные показатели в `processed...in N (<timings>) sec`:

- Время, затраченное на запись значений элементов данных в базу данных;
- Время, затраченное на обновление данных элемента данных (состояние, ошибки, инвентарь узла сети и т. д.);
- Время, затраченное на запись трендов в базу данных;
- Время, затраченное на вычисление триггеров;
- Время, затраченное на обработку событий и действий.

Поддерживаемые платформы

В связи с требованиями безопасности и критически важного характера работы сервера, UNIX является единственной операционной системой, которая может обеспечить необходимую производительность, отказоустойчивость и гибкость. Zabbix работает с ведущими на рынке версиями операционных систем.

Zabbix сервер протестирован на следующих платформах:

- Linux
- Solaris
- AIX
- HP-UX
- Mac OS X
- FreeBSD
- OpenBSD
- NetBSD
- SCO Open Server

**Note:**
Также Zabbix может работать и на других Unix-подобных операционных системах.

Региональные настройки (локаль)

Обратите внимание, что серверу требуется локаль UTF-8, чтобы некоторые текстовые элементы данных интерпретировались корректно. Большинство современных Unix-подобных систем уже имеют локаль UTF-8 по умолчанию, тем не менее, есть некоторые системы где это необходимо указывать вручную.

### 1. Кластер высокой доступности

Обзор

Высокая доступность (HA) обычно требуется в критически важных инфраструктурах, которые могут позволить себе практически нулевое время простоя. Поэтому для любого сервиса, который может отказать, должен быть предусмотрен вариант аварийного переключения, чтобы взять на себя его функции в случае сбоя текущего сервиса.

Zabbix предлагает **встроенное** решение для высокой доступности, которое легко настроить и для которого не требуется предварительный опыт работы с HA. Встроенная HA Zabbix может быть полезна как дополнительный уровень защиты от программных/аппаратных сбоев сервера Zabbix или для сокращения времени простоя из-за обслуживания.

В режиме высокой доступности Zabbix несколько серверов Zabbix работают как узлы кластера. Пока один сервер Zabbix в кластере активен, остальные находятся в режиме ожидания и готовы при необходимости взять на себя его функции.

Переход на Zabbix HA не является необратимым. Вы можете в любой момент вернуться к автономной работе.

См. также: Детали реализации

Включение высокой доступности

Запуск Zabbix server как узла кластера

Для запуска Zabbix server как узла кластера в конфигурации сервера требуются два параметра:

- Параметр `HANodeName` должен быть указан для каждого Zabbix server, который будет узлом HA-кластера.

Это уникальный идентификатор узла (например, `zabbix-node-01`), по которому сервер будет указан в конфигурациях агента и прокси. Если не указать `HANodeName`, сервер будет запущен в автономном режиме.

- Параметр `NodeAddress` должен быть указан для каждого узла.

Параметр `NodeAddress` (`address:port`) будет использоваться веб-интерфейсом Zabbix для подключения к активному узлу сервера. `NodeAddress` должен соответствовать IP-адресу или FQDN-имени соответствующего Zabbix server.

Перезапустите все Zabbix server после внесения изменений в файлы конфигурации. Теперь они будут запущены как узлы кластера. Новый статус серверов можно увидеть в *Reports* > *System information*, а также выполнив:

```
zabbix_server -R ha_status
```

Эта команда во время работы запишет текущий статус HA-кластера в лог Zabbix server (и в stdout):

```
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
```

Подготовка веб-интерфейса

Убедитесь, что `address:port` сервера Zabbix не задан в конфигурации веб-интерфейса (находится в `conf/zabbix.conf.php` в каталоге файлов веб-интерфейса).

```
// Uncomment and set to desired values to override Zabbix hostname/IP and port.
// $ZBX_SERVER                  = '';
// $ZBX_SERVER_PORT             = '';
```

Веб-интерфейс Zabbix автоматически определит активный узел, считывая настройки из таблицы nodes в базе данных Zabbix. Адрес узла сети активного узла будет использоваться как адрес сервера Zabbix.

Конфигурация прокси

Узлы кластера HA (серверы) должны быть перечислены в конфигурации либо пассивного, либо активного прокси Zabbix.

Для пассивного прокси имена узлов должны быть перечислены в параметре Server parameter прокси и разделены **запятой**.

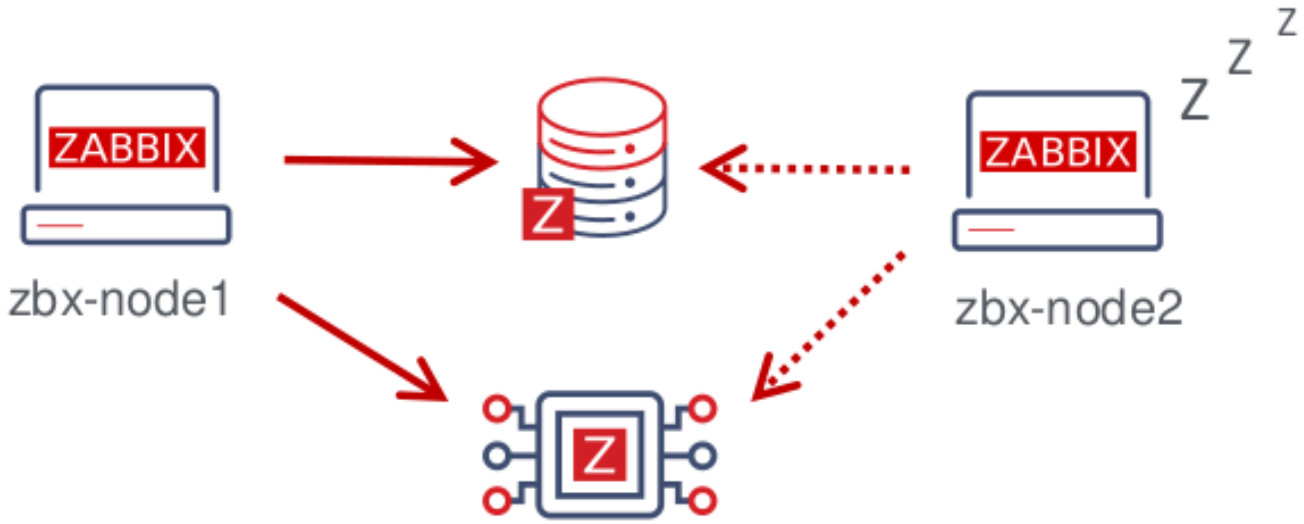
```
Server=zabbix-node-01,zabbix-node-02
```

Для активного прокси имена узлов должны быть перечислены в параметре Server parameter прокси и разделены **точкой с запятой**.

```
Server=zabbix-node-01;zabbix-node-02
```

Конфигурация агента

Узлы кластера HA (серверы) должны быть перечислены в конфигурации Zabbix agent или Zabbix agent 2.




Чтобы включить пассивные проверки, имена узлов должны быть перечислены в параметре Server parameter, разделенные **запятой**.

`Server=zabbix-node-01,zabbix-node-02`

Чтобы включить активные проверки, имена узлов должны быть перечислены в параметре ServerActive parameter. Обратите внимание, что для активных проверок узлы должны быть отделены от любых других серверов запятой, а сами узлы должны быть разделены **точкой с запятой**, например:

`ServerActive=zabbix-node-01;zabbix-node-02`

Переключение на резервный узел

Zabbix автоматически выполнит переключение на другой узел, если активный узел остановится. Для выполнения переключения должен быть как минимум один узел в статусе ожидания.

Насколько быстро произойдет переключение? Все узлы обновляют время последнего доступа (а также статус, если он изменился) каждые 5 секунд. Итак:

- Если активный узел завершает работу и успевает сообщить свой статус как "stopped", другой узел возьмет управление в течение **5 секунд**.
- Если активный узел завершает работу/становится недоступным, не успев обновить свой статус, узлы в статусе ожидания будут ждать **задержку переключения** + 5 секунд, прежде чем взять управление на себя.

Задержка переключения настраивается, поддерживаемый диапазон — от 10 секунд до 15 минут (по умолчанию одна минута). Чтобы изменить задержку переключения, можно выполнить:

`zabbix_server -R ha_set_failover_delay=5m`

Управление HA-кластером

Текущим состоянием HA-кластера можно управлять с помощью специальных параметров управления во время выполнения:

- `ha_status` - записать статус HA-кластера в журнал сервера Zabbix (и в stdout);
- `ha_remove_node=target` - удалить HA-узел, идентифицируемый по его `<target>` - имени или ID узла (имя/ID можно получить из вывода команды `ha_status`), например:

`zabbix_server -R ha_remove_node=zabbix-node-02`

Обратите внимание, что активные/резервные узлы удалить нельзя.

- `ha_set_failover_delay=delay` - задать задержку переключения HA (от 10 секунд до 15 минут; поддерживаются суффиксы времени, например 10s, 1m).

Состояние узла можно отслеживать:

- В *Reports* > *System information*.
- В виджете панели *System information*.
- С помощью параметра `ha_status` управления во время выполнения на сервере (см. выше).

Внутренний элемент данных `zabbix[cluster,discovery,nodes]` можно использовать для обнаружения узлов, так как он возвращает JSON с информацией об узлах высокой доступности.

Отключение HA-кластера

Чтобы отключить кластер высокой доступности:

1. Создайте резервные копии файлов конфигурации.
2. Остановите резервные узлы сети.
3. Удалите параметр HANodeName на активном основном сервере.
4. Перезапустите основной сервер (он запустится в автономном режиме).

Обновление HA-кластера

Чтобы выполнить обновление HA-узлов до новой major-версии:

1. Остановите все узлы.
2. Создайте полную резервную копию базы данных.
3. Если в базе данных используется репликация, убедитесь, что все узлы синхронизированы и не имеют проблем. Не выполняйте обновление, если репликация нарушена.
4. Выберите один узел, который будет выполнять обновление базы данных, измените его конфигурацию на автономный режим, закомментировав `HANodeName`, и выполните обновление.
5. Убедитесь, что обновление базы данных полностью завершено (*System information* должно показывать, что сервер Zabbix запущен).
6. Перезапустите узел в HA-режиме.

7. Обновите и запустите остальные узлы (переводить их в автономный режим не требуется, так как к этому моменту база данных уже обновлена).

При обновлении minor-версии достаточно обновить первый узел, убедиться, что он обновлен и работает, а затем начать обновление следующего узла.

Детали реализации

Кластер высокой доступности (HA) — это решение, которое включается по желанию, и оно поддерживается для сервера Zabbix. Нативное решение HA разработано так, чтобы быть простым в использовании, оно работает между площадками и не имеет специальных требований к базам данных, которые распознает Zabbix. Пользователи могут использовать либо нативное решение HA Zabbix, либо стороннее решение HA — в зависимости от того, что лучше соответствует требованиям к высокой доступности в их среде.

Решение состоит из нескольких экземпляров или узлов `zabbix_server`. Каждый узел:

- Настраивается отдельно.
- Использует одну и ту же базу данных.
- Может иметь несколько режимов: active, standby, unavailable, stopped.

Одновременно активным может быть только один узел. Узел в режиме standby выполняет только один процесс — HA manager. Узел standby не выполняет сбор данных, обработку или другие обычные действия сервера; он не прослушивает порты; у него минимальное число подключений к базе данных.

И активные, и standby-узлы обновляют время своего последнего доступа каждые 5 секунд. Каждый узел standby отслеживает время последнего доступа активного узла. Если время последнего доступа активного узла превышает `failover delay` секунд, узел standby переключается в активный режим и назначает ранее активному узлу статус `unavailable`.

Активный узел отслеживает собственное подключение к базе данных — если оно потеряно более чем на `failover delay`−5 секунд, он должен остановить всю обработку и переключиться в режим standby. Активный узел также отслеживает статус узлов standby — если время последнего доступа узла standby превышает `failover delay` секунд, этому узлу standby назначается статус `unavailable`.

Узлы спроектированы так, чтобы быть совместимыми между minor-версиями Zabbix.

**2. Агент**

Обзор

Агент Zabbix развертывается на контролируемом объекте для активного мониторинга локальных ресурсов и приложений (жесткие диски, память, статистика процессора и т. д.).

Агент собирает рабочую информацию локально и передает данные на сервер Zabbix для дальнейшей обработки. В случае сбоев (например, если жесткий диск заполнен или аварийно завершился процесс службы), сервер Zabbix может активно оповещать администраторов конкретной машины, на которой был зафиксирован сбой.

Агенты Zabbix отличаются высокой эффективностью благодаря использованию системных вызовов ОС для сбора статистической информации.

На узле сети можно запустить несколько экземпляров агента.

Пассивные и активные проверки

Агенты Zabbix могут выполнять пассивные и активные проверки:

- Пассивные проверки - агент Zabbix отвечает на запрос от сервера Zabbix (или прокси). Например, сервер запрашивает данные (например, загрузку CPU), а агент возвращает результат.
- Активные проверки - агент Zabbix собирает и отправляет данные, не ожидая запроса от сервера Zabbix (или прокси). Сначала он получает список элементов мониторинга от сервера (загрузка CPU, доступная память и т. д.), затем собирает необходимые данные и периодически отправляет новые значения обратно.

Тип проверки агента настраивается путем выбора соответствующего типа элемента данных. Агент Zabbix обрабатывает элементы данных типа *Zabbix agent* или *Zabbix agent (active)*.

Поддерживаемые платформы

Поддерживаемые платформы смотрите на странице Требования.

Агент в Unix-подобных системах

Агент Zabbix в Unix-подобных системах запускается на узле сети, который находится под мониторингом.

Установка

Агент Zabbix можно установить в системах на базе Linux одним из следующих способов:

- Пакеты Zabbix - выберите компонент Agent (после выбора версии Zabbix, дистрибутива ОС и версии ОС) и следуйте приведенным инструкциям;
- Исходные тексты Zabbix - загрузите исходные файлы и скомпилируйте агент Zabbix, настроив исходные тексты с параметром `--enable-agent`.

**Attention:**
Как правило, 32-разрядные агенты Zabbix работают в 64-разрядных системах, но в некоторых случаях могут не работать.

**Note:**
Предварительно скомпилированные двоичные файлы агента Zabbix доступны для загрузки для macOS, IBM AIX, FreeBSD, OpenBSD и Solaris. Устаревшие двоичные файлы, совместимые с текущей версией сервера/прокси Zabbix, доступны для NetBSD и HP-UX.

При установке из пакета

Zabbix агент работает как демон. Агента можно запустить, выполнив:

```
systemctl start zabbix-agent
```

Эта команда будет работать на большинстве систем GNU/Linux. На других системах вам, возможно, потребуется выполнить:

```
/etc/init.d/zabbix-agent start
```

Для остановки/перезапуска/просмотра состояния используйте следующие команды:

```
systemctl stop zabbix-agent
systemctl restart zabbix-agent
systemctl status zabbix-agent
```

Запуск вручную

Вы можете запустить Zabbix агент, найдя бинарный файл `zabbix_agentd` и запустив его напрямую; например:

```
zabbix_agentd
```

Агент на Windows системах

Zabbix агент на Windows работает как служба Windows.

Установка

Агент Zabbix можно установить в Windows одним из следующих способов:

- Предварительно скомпилированные двоичные файлы агента Zabbix - загрузите MSI-пакет установщика агента Zabbix и следуйте инструкциям на странице Установка агента Windows из MSI;
- Исходные тексты Zabbix - загрузите исходные файлы и следуйте инструкциям на странице Сборка агента Zabbix в Windows.

Дополнительные сведения об установке агента Zabbix (из ZIP-архива) в качестве службы Windows см. на странице Агент Zabbix в Microsoft Windows.

Параметры

Следующие параметры командной строки можно использовать с агентом Zabbix:

| Parameter | Description |
|---|---|
| **Агент UNIX и Windows** | |
| `-c --config <config-file>` | Путь к файлу конфигурации.<br>Этот параметр можно использовать, чтобы указать файл конфигурации, отличный от файла по умолчанию. |
| `-f --foreground` | Запускать агент Zabbix в режиме переднего плана (по умолчанию: true). |

| Parameter | Description |
|---|---|
| -p --print | Вывести известные элементы данных и завершить работу.<br>Обратите внимание: чтобы также вернуть результаты пользовательского параметра, необходимо указать файл конфигурации (если он не находится в расположении по умолчанию). |
| -t --test <item key> | Проверить указанный элемент данных и завершить работу.<br>Обратите внимание: чтобы также вернуть результаты пользовательского параметра, необходимо указать файл конфигурации (если он не находится в расположении по умолчанию). |
| -T --test-config | Проверить файл конфигурации и завершить работу. |
| -h --help | Показать справочную информацию. |
| -V --version | Показать номер версии. |
| **Только агент UNIX** | |
| -R --runtime-control <option> | Выполнить административные функции. См. управление во время выполнения. |
| **Только агент Windows** | |
| -m | Использовать несколько экземпляров агента (с параметрами -i, -d, -s, -x). |
| --multiple-agents | Чтобы различать имена служб экземпляров, каждое имя службы будет включать значение Hostname из указанного файла конфигурации. |
| -S --startup-type <value> | Установить тип запуска службы агента Zabbix для Windows. Допустимые значения:<br>automatic - *(по умолчанию)* запускать службу автоматически при старте Windows;<br>delayed - отложить запуск службы до завершения запуска служб, которые стартуют автоматически (доступно в Windows Server 2008/Vista и более поздних версиях);<br>manual - запускать службу вручную (пользователем или приложением);<br>disabled - отключить службу, чтобы ее нельзя было запустить пользователем или приложением.<br>Этот параметр можно использовать вместе с параметром -i или отдельно, чтобы изменить тип запуска уже установленной службы. |
| -i --install | Установить агент Zabbix для Windows как службу. |
| -d --uninstall | Удалить службу агента Zabbix для Windows. |
| -s --start | Запустить службу агента Zabbix для Windows. |
| -x --stop | Остановить службу агента Zabbix для Windows. |

Примеры использования параметров командной строки:

- Вывод всех встроенных элементов данных агента со значениями.
- Проверка пользовательского параметра с ключом mysql.ping, определенного в указанном файле конфигурации.
- Установка службы Zabbix Agent для Windows с использованием пути по умолчанию к файлу конфигурации C:\\Program Files\\Zabbix Agent\\zabbix_agentd.conf.
- Установка службы Zabbix Agent [Hostname] для Windows с использованием файла конфигурации zabbix_agentd.conf, расположенного в той же папке, что и исполняемый файл агента, и обеспечение уникальности имени службы путем добавления к нему значения Hostname из файла конфигурации.
- Изменение типа запуска установленной службы Zabbix Agent для Windows с использованием файла конфигурации zabbix_agentd.conf, расположенного в той же папке, что и исполняемый файл агента.

```
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
```

Управление во время работы

С помощью параметров управления во время работы можно изменить уровень логирования процессов агента.

| Option | Description | Target |
|---|---|---|
| `log_level_increase[=<target>]` | Увеличение уровня логирования.<br>Если цель не указана, это затрагивает все процессы. | Цель можно указать как:<br>**тип процесса** - все процессы указанного типа (например, `listener`)<br>См. все типы процессов агента.<br>**тип процесса,N** - тип процесса и номер (например, `listener,3`)<br>**pid** - идентификатор процесса (1 to 65535). Для больших значений указывайте цель как 'тип процесса,N'. |
| `log_level_decrease[=<target>]` | Уменьшение уровня логирования.<br>Если цель не указана, это затрагивает все процессы. | |
| `userparameter_reload` | Перезагрузить значения параметров `UserParameter` и `Include` из текущего файла конфигурации. | |

Примеры:

- Увеличение уровня логирования всех процессов
- Увеличение уровня логирования третьего процесса listener
- Увеличение уровня логирования процесса с PID 1234
- Уменьшение уровня логирования всех процессов active checks

```
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:**
Управление во время работы не поддерживается в OpenBSD, NetBSD и Windows.

Типы процессов агента

Типы процессов агента следующие:

- `active checks` - процесс для выполнения активных проверок;
- `collector` - процесс для сбора данных;
- `listener` - процесс для прослушивания пассивных проверок.

Файл журнала агента можно использовать для наблюдения за этими типами процессов.

Начиная с Zabbix 7.4.6, файл журнала агента создается с правами чтения и записи только для владельца файла. Кроме того, файл доступен для чтения группе владельца. Все остальные разрешения запрещены.

Пользователь процесса

Агент Zabbix в UNIX предназначен для работы от имени пользователя, не являющегося root. Он будет запускаться от имени того пользователя, не являющегося root, от которого был запущен. Таким образом, вы можете запускать агент от имени любого пользователя, не являющегося root, без каких-либо проблем.

Если вы попытаетесь запустить его от имени `root`, он переключится на жестко заданного пользователя `zabbix`, который должен присутствовать в вашей системе. Вы можете запускать агент от имени `root` только в том случае, если соответствующим образом измените параметр `AllowRoot` в файле конфигурации агента.

Файл конфигурации

Путь по умолчанию к файлу конфигурации:

- В UNIX по умолчанию используется `/usr/local/etc/zabbix_agentd.conf` или путь, заданный переменными compile-time `--sysconfdir` или `--prefix`.
- В Windows по умолчанию используется `C:\\Program Files\\Zabbix Agent\\zabbix_agentd.conf`.

Если на узле сети запущен один экземпляр агента, он может использовать файл конфигурации по умолчанию или файл конфигурации, указанный в командной строке. В случае нескольких экземпляров каждый экземпляр агента должен иметь собственный файл конфигурации (один из экземпляров может использовать файл конфигурации по умолчанию).

Подробнее о настройке агента Zabbix см. параметры файла конфигурации для агента Zabbix для Unix или агента Zabbix для Windows.

Региональные настройки (локаль)

Обратите внимание что агенту требуется локаль UTF-8, чтобы некоторые текстовые элементы данных агента смогли вернуть корректное содержимое. Большинство современных Unix-подобных систем имеют локаль UTF-8 установленной по умолчанию; тем не менее, есть некоторые системы, где это необходимо задать вручную.

Код выхода

Агент Zabbix возвращает 0 в случае успешного завершения и 1 в случае ошибки.

### 3. Агент 2

Обзор

Zabbix агент 2 — это новое поколение Zabbix агента, написанное на языке Go (с частичным использованием кода на C из Zabbix агента). Zabbix агент 2 разработан для:

- уменьшения количества TCP-соединений;
- улучшения параллельности проверок;
- лёгкого расширения при помощи плагинов, которые обеспечивают простые проверки с минимальным кодом и поддерживают сложные проверки, состоящие из длительно выполняемых скриптов и автономного сбора данных с периодической отправкой данных;
- функционирования в качестве замены Zabbix агента, поддерживая все предыдущие функции.

Все метрики, поддерживаемые Zabbix агентом 2, собираются плагинами.

Пассивные и активные проверки

Агент Zabbix 2 поддерживает пассивные и активные проверки, как и агент Zabbix. Кроме того, активные проверки агента Zabbix 2 поддерживают гибкие интервалы/интервалы по расписанию и параллельное выполнение проверок в рамках одного активного сервера.

> **Note:**
> По умолчанию после перезапуска агент Zabbix 2 запланирует первый сбор данных для активных проверок на условно случайное время в пределах интервала обновления элемента данных, чтобы предотвратить всплески нагрузки на ресурсы. Чтобы выполнять активные проверки, у которых нет интервала обновления типа *Scheduling*, сразу после перезапуска агента, задайте параметр `ForceActiveChecksOnStart` (на глобальном уровне) или `Plugins.<Plugin name>.System.ForceActiveChecksOnStart` (влияет только на проверки конкретного плагина) в файле конфигурации. Если параметр уровня плагина задан, он переопределяет глобальный параметр.

Проверка параллельного выполнения

Проверки из разных плагинов могут выполняться одновременно. Количество одновременных проверок в рамках одного плагина ограничено параметром емкости плагина. Для каждого плагина может быть задано жестко закодированное значение емкости (1000 по умолчанию), которое можно уменьшить с помощью параметра `Plugins.<PluginName>.System.Capaci` в параметре конфигурации `Plugins`.

Поддерживаемые платформы

Поддерживаемые платформы смотрите на странице Требования.

Агент 2 в Unix-подобных системах

Агент 2 Zabbix в Unix-подобных системах запускается на узле сети, который находится под мониторингом.

Установка

Zabbix агент 2 можно установить на системах на базе Linux одним из следующих способов:

- Пакеты Zabbix - выберите компонент Agent 2 (после выбора версии Zabbix, дистрибутива ОС и версии ОС) и следуйте инструкциям;
- Исходные тексты Zabbix - загрузите исходные файлы и скомпилируйте агент, выполнив настройку с параметром `--enable-agent2`.

**Note:**
Возможности мониторинга Zabbix агент 2 можно расширить с помощью загружаемых плагинов, которые доступны отдельно. Подробности см. в разделе Загружаемые плагины.

Если установлен как пакет

Zabbix agent 2 работает как процесс переднего плана и для работы в фоновом режиме использует внешний менеджер служб (например, systemd). В Linux у Zabbix agent 2 нет встроенной поддержки демонизации.

Агент можно запустить, выполнив:

```
systemctl start zabbix-agent2
```

Чтобы остановить, перезапустить или проверить состояние Zabbix agent 2, используйте следующие команды:

```
systemctl stop zabbix-agent2
systemctl restart zabbix-agent2
systemctl status zabbix-agent2
```

Запуск вручную

Вы можете запустить агент Zabbix, найдя исполняемый файл `zabbix_agent2` и запустив его напрямую; например:

```
zabbix_agent2
```

Агент 2 в системах Windows

Zabbix агент 2 работает как отдельный процесс; однако его также можно запускать как службу Windows.

Установка

Zabbix agent 2 можно установить в Windows одним из следующих способов:

- Предварительно скомпилированные двоичные файлы Zabbix agent - загрузите MSI-пакет установщика агента и следуйте инструкциям на странице Установка агента Windows из MSI;
- Исходные коды Zabbix - загрузите исходные файлы и следуйте инструкциям на странице Сборка Zabbix agent 2 в Windows.

**Note:**
Возможности мониторинга Zabbix agent 2 можно расширить с помощью загружаемых плагинов, которые доступны отдельно. Подробности см. на странице Загружаемые плагины.

Дополнительные сведения об установке Zabbix agent 2 (из ZIP-архива) в качестве службы Windows см. на странице Zabbix agent в Microsoft Windows.

Параметры

С Zabbix agent 2 можно использовать следующие параметры командной строки:

| Parameter | Description |
|---|---|
| **Агент UNIX и Windows** | |
| `-c --config <config-file>` | Путь к файлу конфигурации.<br>Вы можете использовать этот параметр, чтобы указать файл конфигурации, отличный от файла по умолчанию. |
| `-f --foreground` | Запустить Zabbix agent в режиме переднего плана (по умолчанию: true). |
| `-p --print` | Вывести известные элементы данных и выйти.<br>Обратите внимание: чтобы также вернуть результаты пользовательского параметра, необходимо указать файл конфигурации (если он не находится в расположении по умолчанию). |
| `-t --test <item key>` | Проверить указанный элемент данных и выйти.<br>Обратите внимание: чтобы также вернуть результаты пользовательского параметра, необходимо указать файл конфигурации (если он не находится в расположении по умолчанию). |
| `-T --test-config` | Проверить файл конфигурации и выйти. |
| `-h --help` | Вывести справочную информацию и выйти. |
| `-v --verbose` | Вывести отладочную информацию. Используйте этот параметр вместе с параметрами `-p` и `-t`. |

| Parameter | Description |
|---|---|
| -V --version | Вывести версию агента и сведения о лицензии. |
| -R --runtime-control <option> | Выполнить административные функции. См. управление во время выполнения. |
| **Только агент Windows** | |
| -m --multiple-agents | Использовать несколько экземпляров агента (с параметрами -i, -d, -s, -x).<br>Чтобы различать имена служб экземпляров, каждое имя службы будет включать значение Hostname из указанного файла конфигурации. |
| -S --startup-type <value> | Установить тип запуска службы Zabbix agent для Windows. Допустимые значения:<br>automatic - *(по умолчанию)* запускать службу автоматически при старте Windows;<br>delayed - отложить запуск службы до завершения запуска служб, которые стартуют автоматически;<br>manual - запускать службу вручную (пользователем или приложением);<br>disabled - отключить службу, чтобы ее нельзя было запустить пользователем или приложением.<br>Вы можете использовать этот параметр вместе с параметром -i или отдельно, чтобы изменить тип запуска уже установленной службы. |
| -i --install | Установить Zabbix agent для Windows как службу. |
| -d --uninstall | Удалить службу Zabbix agent для Windows. |
| -s --start | Запустить службу Zabbix agent для Windows. |
| -x --stop | Остановить службу Zabbix agent для Windows. |

Конкретные примеры использования параметров командной строки:

- Вывести все встроенные элементы данных агента со значениями.
- Проверить пользовательский параметр с ключом mysql.ping, определенным в указанном файле конфигурации.
- Установить службу Zabbix Agent для Windows, используя путь по умолчанию к файлу конфигурации C:\\Program Files\\Zabbix Agent 2\\zabbix_agent2.conf.
- Изменить тип запуска установленной службы Zabbix Agent для Windows, используя файл конфигурации zabbix_agent2.conf, расположенный в той же папке, что и исполняемый файл агента.

```
zabbix_agent2 --print
zabbix_agent2 -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
zabbix_agent2.exe -i
zabbix_agent2.exe -c zabbix_agent2.conf -S delayed
```

Управление во время выполнения

Управление во время выполнения предоставляет несколько возможностей для удаленного управления.

| Option | Description |
|---|---|
| log_level_increase | Увеличить уровень журналирования. |
| log_level_decrease | Уменьшить уровень журналирования. |
| metrics | Вывести список доступных метрик. |
| version | Показать версию агента. |
| userparameter_reload | Перезагрузить значения параметров UserParameter и Include из текущего файла конфигурации. |
| help | Показать справочную информацию по управлению во время выполнения. |

Примеры:

- Увеличение уровня журналирования для агента 2.
- Вывести параметры управления во время выполнения.

```
zabbix_agent2 -R log_level_increase
zabbix_agent2 -R help
```

Начиная с Zabbix 7.4.1, команды управления во время выполнения для агента 2 выводят результат в stdout (стандартный вывод) вместо stderr (стандартный поток ошибок).

Файл конфигурации

Путь по умолчанию к файлу конфигурации:

- В UNIX по умолчанию используется `/usr/local/etc/zabbix_agent2.conf` или путь, заданный переменными время компиляции `--sysconfdir` или `--prefix`.
- В Windows по умолчанию используется `C:\\Program Files\\Zabbix Agent 2\\zabbix_agent2.conf`.

Если на узле сети запущен один экземпляр агента, он может использовать файл конфигурации по умолчанию или файл конфигурации, указанный в командной строке. В случае нескольких экземпляров каждый экземпляр агента должен иметь собственный файл конфигурации (один из экземпляров может использовать файл конфигурации по умолчанию).

Параметры конфигурации Zabbix agent 2 в основном совпадают с параметрами Zabbix agent, за некоторыми исключениями:

| Parameters | Description |
|---|---|
| Специфичные для Zabbix agent 2 параметры конфигурации | |
| `ControlSocket` | Путь к сокету управления во время выполнения. Zabbix agent 2 использует сокет управления для команд во время выполнения. |
| `EnablePersistentBuffer`, `PersistentBufferFile`, `PersistentBufferPeriod` | Эти параметры используются для настройки постоянного хранилища в Zabbix agent 2 для активных элементов данных. |
| `ForceActiveChecksOnStart` | Определяет, должен ли агент выполнять активные проверки сразу после перезапуска или распределять их равномерно во времени. |
| `Plugins` | У плагинов могут быть собственные параметры в формате `Plugins.<Plugin name>.<Parameter>=<value>`. Распространенный параметр плагина — `System.Capacity`, который задает предел числа проверок, выполняемых одновременно. |
| `StatusPort` | Порт, на котором Zabbix agent 2 будет ожидать HTTP-запросы состояния и отображать список настроенных плагинов и некоторые внутренние параметры. |
| Параметры конфигурации Zabbix agent, не поддерживаемые в Zabbix agent 2 | |
| `AllowRoot`, `User` | Не поддерживаются в Zabbix agent 2, поскольку он не поддерживает демонизацию. |
| `EnableRemoteCommands` | Устарел; вместо него в обоих агентах используются параметры `AllowKey` и `DenyKey`. |
| `ListenBacklog` | Используется в Zabbix agent для определения максимального числа ожидающих подключений в очереди TCP. Не поддерживается в Zabbix agent 2. |
| `LoadModule`, `LoadModulePath` | Загружаемые модули не поддерживаются в Zabbix agent 2. |
| `MaxLinesPerSecond`, `LogRemoteCommands` | В Zabbix agent 2 вместо них используются `Plugins.Log.MaxLinesPerSecond` и `Plugins.SystemRun.LogRemoteCommands`. |

| Parameters | Description |
|---|---|
| StartAgents | Этот параметр используется в Zabbix agent для увеличения параллелизма пассивных проверок или их отключения. В Zabbix agent 2 параллелизм настраивается на уровне плагина и может быть ограничен параметром capacity. Пассивные проверки в Zabbix Agent 2 отключены, если не указан параметр Server. |

Подробнее см. параметры файла конфигурации для Zabbix agent2.

Коды выхода

Zabbix agent 2 также может быть скомпилирован с более старыми версиями OpenSSL (1.0.1, 1.0.2).

В этом случае Zabbix предоставляет мьютексы для блокировки в OpenSSL. Если блокировка или разблокировка мьютекса завершается сбоем, то сообщение об ошибке выводится в стандартный поток ошибок (STDERR), а Agent 2 завершает работу с кодом возврата 2 или 3 соответственно.

**4. Прокси**

Обзор

Прокси Zabbix — это процесс, который может собирать данные мониторинга с одного или нескольких контролируемых устройств и отправлять информацию на сервер Zabbix, фактически работая от имени сервера. Все собранные данные буферизуются локально, а затем передаются на сервер Zabbix, к которому относится прокси.

Развертывание прокси необязательно, но может быть очень полезным для распределения нагрузки одного сервера Zabbix. Если данные собирают только прокси, обработка на сервере требует меньше ресурсов CPU и операций ввода-вывода диска.

Прокси Zabbix — это идеальное решение для централизованного мониторинга удаленных площадок, филиалов и сетей без локальных администраторов.

Для прокси Zabbix требуется отдельная база данных.

> **Attention:**
> Обратите внимание, что поддерживаемые для прокси Zabbix базы данных — SQLite, MySQL и PostgreSQL.

См. также: Использование прокси в распределенной среде

Запущенный прокси

Если установлен из пакета

Zabbix прокси работает как демон. Его можно запустить, выполнив:

```
systemctl start zabbix-proxy
```

Эта команда будет работать на большинстве систем GNU/Linux. На других системах вам, возможно, потребуется выполнить:

```
/etc/init.d/zabbix-proxy start
```

Аналогично, для остановки/перезапуска/просмотра состояния Zabbix прокси, используйте следующие команды:

```
systemctl stop zabbix-proxy
systemctl restart zabbix-proxy
systemctl status zabbix-proxy
```

Запуск вручную

Если вышеописанное не сработало, необходимо запустить его вручную. Найдите путь к бинарному файлу zabbix_proxy и выполните:

```
zabbix_proxy
```

С Zabbix proxy можно использовать следующие параметры командной строки:

```
-c --config <file>
-f --foreground                         Zabbix proxy
-R --runtime-control <option>
-T --test-config
-h --help
-V --version
```

Примеры запуска Zabbix proxy с параметрами командной строки:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf
zabbix_proxy --help
zabbix_proxy -V
```

Управление во время выполнения

Параметры управления во время выполнения:

| Option | Description | Target |
|---|---|---|
| `config_cache_reload` | Перезагрузить кэш конфигурации.<br>Игнорируется, если кэш в данный момент загружается.<br>Активный прокси Zabbix подключится к серверу Zabbix и запросит данные конфигурации.<br>Пассивный прокси Zabbix запросит данные конфигурации у сервера Zabbix при следующем подключении сервера к прокси. | |
| `history_cache_clear=target` | Очистить кэш истории для элемента данных, указанного по его ID.<br>Затрагивает все значения элемента данных, кроме первого и последнего значения. | **target** - ID элемента данных. |
| `diaginfo[=<section>]` | Собрать диагностическую информацию в файле журнала прокси. | `historycache` - статистика кэша истории;<br>`preprocessing` - статистика менеджера предварительной обработки;<br>`locks` - список мьютексов (пусто в системах *BSD*). |
| `snmp_cache_reload` | Перезагрузить кэш SNMP - очистить свойства SNMP engine (engine time, engine boots, engine id, credentials) для всех узлов сети.<br>Используйте для принудительной глобальной очистки кэша при устранении проблем SNMP. | |
| `housekeeper_execute` | Запустить процедуру housekeeping.<br>Игнорируется, если процедура housekeeping уже выполняется. | |
| `log_level_increase[=target]` | Увеличить уровень журналирования, влияет на все процессы, если target не указан.<br>Не поддерживается в системах *BSD*. | **process type** - все процессы указанного типа (например, `poller`).<br>См. все типы процессов прокси.<br>**process type,N** - тип процесса и номер (например, `poller,3`).<br>**pid** - идентификатор процесса (1 to 65535).<br>Для больших значений укажите target как 'process type,N'. |
| `log_level_decrease[=target]` | Уменьшить уровень журналирования, влияет на все процессы, если target не указан.<br>Не поддерживается в системах *BSD*. | |

| Option | Description | Target |
|---|---|---|
| `prof_enable[=<target>]` | Включить профилирование.<br>Влияет на все процессы, если target не указан.<br>Включенное профилирование предоставляет сведения обо всех rwlocks/mutexes по имени функции. | **process type** - все процессы указанного типа (например, `history syncer`).<br>См. все типы процессов прокси.<br>**process type,N** - тип процесса и номер (например, `history syncer,1`)<br>**pid** - идентификатор процесса (1 to 65535). Для больших значений укажите target как 'process type,N'.<br>**scope** - `rwlock`, `mutex`, `processing` можно использовать с типом процесса и номером (например, `history syncer,1,processing`) или со всеми процессами типа (например, `history syncer,rwlock`). |
| `prof_disable[=<target>]` | Отключить профилирование.<br>Влияет на все процессы, если target не указан. | **process type** - все процессы указанного типа (например, `history syncer`).<br>См. все типы процессов прокси.<br>**process type,N** - тип процесса и номер (например, `history syncer,1`).<br>**pid** - идентификатор процесса (1 to 65535). Для больших значений укажите target как 'process type,N'. |

Пример использования управления во время выполнения для перезагрузки кэша конфигурации прокси:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R config_cache_reload
```

Пример использования управления во время выполнения для очистки кэша истории для элемента данных:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R history_cache_clear=42243
```

Примеры использования управления во время выполнения для сбора диагностической информации:

```
### :
zabbix_proxy -R diaginfo

### :
zabbix_proxy -R diaginfo=historycache
```

Пример использования управления во время выполнения для перезагрузки кэша SNMP:

```
zabbix_proxy -R snmp_cache_reload
```

**Attention:**
Когда интерфейс SNMPv3 обновляется через веб-интерфейс Zabbix, Zabbix в большинстве случаев автоматически перезагружает новые учетные данные SNMPv3 для этого интерфейса; используйте `-R snmp_cache_reload` только если опрос по-прежнему не выполняется после изменения учетных данных (например, из-за несоответствий engineBoots/engineID или устройств, не соответствующих RFC), либо когда требуется принудительно очистить глобальный кэш SNMP для устранения неполадок.

Пример использования управления во время выполнения для запуска housekeeper:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R housekeeper_execute
```

Примеры использования управления во время выполнения для изменения уровня журналирования:

```
### :
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase

### poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=poller,2

### PID 1234:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=1234
```

```
###                             http poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_decrease="http poller"
```

Пользователь процесса

Прокси Zabbix предназначен для запуска от имени пользователя, не являющегося root. Он будет работать от имени того пользователя, не являющегося root, под которым был запущен. Таким образом, вы можете запускать прокси от имени любого пользователя, не являющегося root, без каких-либо проблем.

Если вы попытаетесь запустить его от имени `root`, он переключится на жестко заданного пользователя `zabbix`, который должен присутствовать в вашей системе. Вы можете запускать прокси от имени `root` только в том случае, если соответствующим образом измените параметр `AllowRoot` в файле конфигурации прокси.

Файл конфигурации

См. параметры файла конфигурации для получения подробной информации о настройке `zabbix_proxy`.

Типы процессов и потоки прокси

- `agent poller` - асинхронный процесс опроса для пассивных проверок с рабочим потоком;
- `availability manager` - процесс для обновления доступности узлов сети;
- `browser poller` - опросчик для проверок элементов данных браузера;
- `configuration syncer` - процесс для управления кэшем конфигурационных данных в памяти;
- `data sender` - отправитель данных прокси;
- `discovery manager` - процесс-менеджер для обнаружения устройств;
- `discovery worker` - процесс для обработки задач обнаружения от discovery manager;
- `history syncer` - запись истории в БД;
- `housekeeper` - процесс для удаления устаревшей истории элементов данных;
- `http agent poller` - асинхронный процесс опроса для HTTP-проверок с рабочим потоком;
- `http poller` - опросчик для мониторинга веба;
- `icmp pinger` - опросчик для проверок icmpping;
- `internal poller` - опросчик для внутренних проверок;
- `ipmi manager` - менеджер опросчиков IPMI;
- `ipmi poller` - опросчик для проверок IPMI;
- `java poller` - опросчик для проверок Java;
- `odbc poller` - опросчик для проверок ODBC;
- `poller` - обычный опросчик для пассивных проверок;
- `preprocessing manager` - менеджер задач предварительной обработки с потоками рабочих предварительной обработки;
- `preprocessing worker` - поток для предварительной обработки данных;
- `self-monitoring` - процесс для сбора внутренней статистики сервера;
- `snmp poller` - асинхронный процесс опроса для проверок SNMP с рабочим потоком (только элементы данных `walk[OID]` и `get[OID]`);
- `snmp trapper` - trapper для SNMP traps;
- `task manager` - процесс для удаленного выполнения задач, запрошенных другими компонентами (например, закрытие проблемы, подтверждение проблемы, проверка значения элемента данных сейчас, функциональность удаленной команды);
- `trapper` - trapper для активных проверок, traps и связи с прокси;
- `unreachable poller` - опросчик для недоступных устройств;
- `vmware collector` - сборщик данных VMware, отвечающий за сбор данных из служб VMware.

Файл журнала прокси можно использовать для наблюдения за этими типами процессов.

Начиная с Zabbix 7.4.6, файл журнала прокси создается с правами на чтение и запись только для владельца файла. Кроме того, файл доступен для чтения группе владельца. Все остальные права запрещены.

Различные типы процессов прокси Zabbix можно отслеживать с помощью внутреннего `zabbix[process,<type>,<mode>,<state>]` элемента данных.

Статистика транзакций history syncer

Заголовок процесса history syncer отображает подробную статистику по транзакциям history syncer.

```
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
```

Временные значения в `processed...in N (<timings>) sec` означают:

- Время, затраченное на запись значений элементов данных в базу данных.
- Время, затраченное на обновление данных элемента данных (состояние, ошибки).

Поддерживаемые платформы

Прокси Zabbix работает на том же списке поддерживаемых платформ, что и сервер Zabbix.

Буфер памяти

Буфер памяти позволяет сохранять новые данные (значения элементов данных, сетевое обнаружение, авторегистрацию узлов сети) в буфере и загружать их на сервер Zabbix без обращения к базе данных. Буфер памяти был введен для прокси в Zabbix 7.0.

В установках до Zabbix 7.0 собранные данные сохранялись в базе данных перед загрузкой на сервер Zabbix. Для таких установок это остается поведением по умолчанию после обновления до Zabbix 7.0.

Для оптимальной производительности рекомендуется настроить использование буфера памяти на прокси. Это можно сделать, изменив значение ProxyBufferMode с disk (жестко заданное значение по умолчанию для существующих установок) на hybrid (рекомендуется) или memory. Также необходимо задать размер буфера памяти (параметр ProxyMemoryBufferSize).

В гибридном режиме буфер защищен от потери данных: несохраненные данные записываются в базу данных, если прокси остановлен, буфер переполнен или данные слишком старые. Когда все значения будут записаны в базу данных, прокси снова начнет использовать буфер памяти.

В режиме памяти буфер памяти будет использоваться, однако защиты от потери данных нет. Если прокси остановлен или память переполнится, несохраненные данные будут отброшены.

Гибридный режим (ProxyBufferMode=hybrid) применяется ко всем новым установкам начиная с Zabbix 7.0.

Дополнительные параметры, такие как ProxyMemoryBufferSize и ProxyMemoryBufferAge, определяют размер буфера памяти и максимальный возраст данных в буфере соответственно.

Обратите внимание, что при конфликтующей конфигурации прокси выведет ошибку и не запустится, например, если:

- ProxyBufferMode установлен в hybrid или memory, а ProxyMemoryBufferSize равен 0.
- ProxyBufferMode установлен в hybrid или memory, а ProxyLocalBuffer не равен 0.

Региональные настройки (локаль)

Обратите внимание, что прокси серверу требуется локаль UTF-8, чтобы некоторые текстовые элементы данных интерпретировались корректно. Большинство современных Unix-подобных систем уже имеют локаль UTF-8 по умолчанию; тем не менее, есть некоторые системы, где это необходимо указывать вручную.

Расчет очередей во время обслуживания

> **Attention:**
> Прокси Zabbix не учитывает периоды обслуживания; подробности см. в разделе Расчет очередей во время обслуживания.

**5. Java gateway**

Обзор

Java gateway Zabbix можно установить из исходного кода или пакетов.

Нативная поддержка мониторинга приложений JMX существует в виде демона Zabbix под названием "Zabbix Java gateway". Zabbix Java gateway — это демон, написанный на Java. Чтобы узнать значение определенного счетчика JMX на узле сети, сервер Zabbix выполняет запрос к Zabbix Java gateway, который использует JMX management API для удаленного запроса к нужному приложению. Для приложения не требуется установка дополнительного ПО, его достаточно запустить с параметром -Dcom.sun.management.jmxremote в командной строке.

Java gateway принимает входящие подключения от сервера Zabbix или прокси и может использоваться только как "passive proxy". В отличие от прокси Zabbix, его также можно использовать из прокси Zabbix (прокси Zabbix нельзя объединять в цепочку). Доступ к каждому Java gateway настраивается непосредственно в файле конфигурации сервера Zabbix или прокси, поэтому для каждого сервера Zabbix или прокси можно настроить только один Java gateway. Если у узла сети будут элементы данных типа *JMX agent* и элементы данных другого типа, только элементы данных типа *JMX agent* будут передаваться в Java gateway для получения.

Когда элемент данных должен быть обновлен через Java gateway, сервер Zabbix или прокси подключится к Java gateway и запросит значение, которое Java gateway, в свою очередь, получит и передаст обратно серверу или прокси. Таким образом, Java gateway не кэширует никакие значения.

Сервер Zabbix или прокси имеет специальный тип процессов, которые подключаются к Java gateway; их количество определяется параметром StartJavaPollers. Внутри Java gateway запускает несколько потоков, количество которых определяется параметром START_POLLERS. На стороне сервера, если подключение занимает более "Timeout" секунд, оно будет завершено, но Java gateway при этом может все еще быть занят получением значения из счетчика JMX. Чтобы решить эту проблему, в Java gateway есть параметр TIMEOUT, который позволяет задать тайм-аут для сетевых операций JMX.

Сервер или прокси Zabbix будет по возможности объединять запросы к одной цели JMX в общие пакеты (это зависит от интервалов элементов данных) и отправлять их в Java gateway в одном соединении для повышения производительности.

Рекомендуется, чтобы значение StartJavaPollers было меньше или равно START_POLLERS; в противном случае могут возникать ситуации, когда в Java gateway не будет доступных потоков для обработки входящих запросов. В таких случаях Java gateway использует ThreadPoolExecutor.CallerRunsPolicy, что означает, что основной поток будет обрабатывать входящий запрос и временно не будет принимать новые запросы.

Если вы пытаетесь отслеживать Java-приложения на базе Wildfly с помощью Java gateway Zabbix, установите последнюю версию jboss-client.jar, доступную на странице загрузки Wildfly.

**1 Установка из пакетов RHEL**

Обзор

Если Zabbix Java Gateway установлен из пакетов, следующая информация вам поможет в настройке Zabbix Java gateway.

Настройка и запуск Java gateway

Параметры конфигурации Zabbix Java gateway можно изменить в файле:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Подробнее см. параметры конфигурации Zabbix Java gateway.

Чтобы запустить Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Чтобы автоматически запускать Zabbix Java gateway при загрузке системы:

RHEL 7 и новее:

```
systemctl enable zabbix-java-gateway
```

RHEL до 7:

```
chkconfig --level 12345 zabbix-java-gateway on
```

Настройка сервера для использования с Java gateway

Когда Java gateway запущен и работает, необходимо указать Zabbix server, где найти Zabbix Java gateway. Для этого задаются параметры JavaGateway и JavaGatewayPort в файле конфигурации сервера. Если узел сети, на котором работает приложение JMX, контролируется Zabbix proxy, то параметры подключения следует указать в файле конфигурации прокси вместо этого.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

По умолчанию сервер не запускает процессы, связанные с мониторингом JMX. Однако, если вы хотите использовать его, необходимо указать количество предварительно порожденных экземпляров Java pollers. Это делается так же, как задаются обычные pollers и trappers.

```
StartJavaPollers=5
```

Не забудьте перезапустить сервер или прокси после завершения настройки.

Отладка Java gateway

Файл журнала Java gateway Zabbix находится здесь:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Если вы хотите повысить уровень журналирования, отредактируйте файл:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

и измените info в level="info" на debug или даже trace (для глубокой диагностики):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
      <root level="info">
              <appender-ref ref="FILE" />
      </root>

</configuration>
```

Мониторинг JMX

См. страницу Мониторинг JMX для получения более подробной информации.

**2 Установка из пакетов Debian/Ubuntu**

Обзор

Если Zabbix Java Gateway установлен из пакетов, следующая информация вам поможет в настройке Zabbix Java gateway.

Настройка и запуск Java gateway

Конфигурацию Java gateway можно изменить в файле:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Дополнительные сведения см. в разделе параметры конфигурации Zabbix Java gateway.

Чтобы запустить Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Чтобы автоматически запускать Zabbix Java gateway при загрузке системы:

```
systemctl enable zabbix-java-gateway
```

Настройка сервера для использования Java gateway

После запуска Java gateway необходимо указать Zabbix серверу, где найти Zabbix Java gateway. Для этого в файле конфигурации сервера задаются параметры `JavaGateway` и `JavaGatewayPort`. Если узел сети, на котором работает приложение JMX, контролируется через Zabbix прокси, то параметры подключения следует указать в файле конфигурации прокси.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

По умолчанию сервер не запускает никаких процессов, связанных с мониторингом JMX. Однако если вы хотите использовать его, необходимо указать количество предварительно порожденных экземпляров Java pollers. Это делается так же, как задаются обычные pollers и trappers.

```
StartJavaPollers=5
```

Не забудьте перезапустить сервер или прокси после завершения настройки.

Отладка Java gateway

Файл журнала Java gateway Zabbix:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Если вы хотите повысить уровень журналирования, отредактируйте файл:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

и измените `info` в `level="info"` на debug или даже `trace` (для глубокой диагностики):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
      <root level="info">
              <appender-ref ref="FILE" />
      </root>

</configuration>
```

Мониторинг JMX

См. страницу Мониторинг JMX для получения более подробной информации.

### 3 Установка из исходных кодов

Обзор

Если Zabbix Java Gateway установлен из исходных кодов, следующая информация поможет вам настроить Zabbix Java gateway.

Обзор файлов

Если вы получили Java gateway из исходных кодов, у вас должен был получиться набор shell-скриптов, JAR- и конфигурационных файлов в `$PREFIX/sbin/zabbix_java`. Назначение этих файлов приведено ниже.

`bin/zabbix-java-gateway-$VERSION.jar`

Сам JAR-файл Java gateway.

```
lib/logback-core-1.5.16.jar
lib/logback-classic-1.5.16.jar
lib/slf4j-api-2.0.16.jar
lib/android-json-4.3_r3.1.jar
```

Зависимости Java gateway: библиотека Logback, SLF4J и Android JSON.

```
lib/logback.xml
lib/logback-console.xml
```

Конфигурационные файлы для Logback.

```
shutdown.sh
startup.sh
```

Удобные скрипты для запуска и остановки Java gateway.

`settings.sh`

Конфигурационный файл, который подключается указанными выше скриптами startup и shutdown.

Настройка и запуск Java gateway

По умолчанию Java gateway прослушивает порт 10052. Если вы планируете запускать Java gateway на другом порту, вы можете указать это в скрипте `settings.sh`. См. описание файла конфигурации Java gateway, чтобы узнать, как задать этот и другие параметры.

> **Warning:**
> Порт 10052 не зарегистрирован в IANA.

После того как вы настроите параметры, вы можете запустить Java gateway, выполнив скрипт `startup.sh`:

```
./startup.sh
```

Аналогично, когда Java gateway больше не нужен, выполните скрипт `shutdown.sh`, чтобы остановить его:

```
./shutdown.sh
```

Обратите внимание, что в отличие от server или proxy, Java gateway является легковесным и не требует базы данных.

Настройка сервера для использования с Java gateway

После запуска Java gateway необходимо указать Zabbix сервер, где найти Zabbix Java gateway. Для этого в файле конфигурации сервера задаются параметры `JavaGateway` и `JavaGatewayPort`. Если узел сети, на котором работает JMX-приложение, отслеживается через Zabbix прокси, то параметры подключения следует указать в файле конфигурации прокси.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

По умолчанию сервер не запускает никаких процессов, связанных с мониторингом JMX. Однако если вы хотите использовать его, необходимо указать количество предварительно порожденных экземпляров Java pollers. Это делается так же, как при указании обычных pollers и trappers.

```
StartJavaPollers=5
```

Не забудьте перезапустить сервер или прокси после завершения настройки.

Отладка Java gateway

Если возникают какие-либо проблемы с Java gateway или сообщение об ошибке, которое вы видите в веб-интерфейсе для элемента данных, недостаточно информативно, возможно, вам стоит посмотреть файл журнала Java gateway.

По умолчанию Java gateway записывает свою активность в файл `/tmp/zabbix_java.log` с уровнем журналирования "info". Иногда этой информации недостаточно, и требуется информация с уровнем журналирования "debug". Чтобы повысить уровень журналирования, измените файл `lib/logback.xml` и значение атрибута level тега `<root>` на `debug`:

```
<root level="debug">
  <appender-ref ref="FILE" />
</root>
```

Обратите внимание, что в отличие от сервера Zabbix или прокси Zabbix, после изменения файла `logback.xml` перезапуск Zabbix Java gateway не требуется - изменения в `logback.xml` будут применены автоматически. Когда отладка будет завершена, вы можете вернуть уровень журналирования на "info".

Если вы хотите вести журнал в другой файл или в совершенно другой носитель, например в базу данных, настройте файл `logback.xml` в соответствии со своими потребностями. Подробнее см. в Руководстве Logback.

Иногда для целей отладки полезно запускать Java gateway как консольное приложение, а не как демон. Чтобы сделать это, закомментируйте переменную `PID_FILE` в `settings.sh`. Если `PID_FILE` опущена, сценарий `startup.sh` запускает Java gateway как консольное приложение и заставляет Logback использовать файл `lib/logback-console.xml`, который не только записывает журнал в консоль, но и включает уровень журналирования "debug".

Наконец, обратите внимание, что поскольку Java gateway использует SLF4J для журналирования, вы можете заменить Logback на любой другой фреймворк, поместив соответствующий JAR-файл в каталог lib. Подробнее см. в Руководстве SLF4J.

Мониторинг JMX

См. страницу Мониторинг JMX для получения дополнительных сведений.

### 6. Sender

Обзор

Zabbix sender — это утилита командной строки, которая может использоваться для отправки данных о производительности на сервер Zabbix для обработки.

Обычно утилита используется в длительно выполняющихся пользовательских скриптах для периодической отправки данных о доступности и производительности.

Для отправки результатов напрямую на сервер Zabbix или прокси необходимо настроить тип элемента данных *Zabbix trapper*.

Также см. библиотеку Python для Zabbix, в которой встроена функциональность, позволяющая работать как Zabbix sender.

Запуск Zabbix sender

Пример запуска Zabbix UNIX sender:

```
cd bin
./zabbix_sender -z zabbix -s "Linux DB3" -k db.connections -o 43
```

где:

- `z` - хост сервера Zabbix (можно также использовать IP-адрес);
- `s` - техническое имя контролируемого узла сети (как зарегистрировано во веб-интерфейсе Zabbix);
- `k` - ключ элемента данных;
- `o` - значение для отправки.

**Attention:**
Параметры, содержащие пробелы, должны быть заключены в двойные кавычки.

Zabbix sender можно использовать для отправки нескольких значений из входного файла. Дополнительные сведения см. в manpage Zabbix sender.

Если указан файл конфигурации, Zabbix sender использует все адреса, определенные в конфигурационном параметре агента `ServerActive`, для отправки данных. Если отправка на один адрес завершается неудачей, sender пытается

отправить данные на другие адреса. Если отправка пакетных данных на один адрес завершается неудачей, последующие пакеты на этот адрес не отправляются.

Zabbix sender принимает строки в кодировке UTF-8 (как для систем Unix-подобного типа, так и для Windows) без начального байта порядка (BOM) в файле.

Zabbix sender в Windows можно запускать аналогичным образом:

```
zabbix_sender.exe [options]
```

Сценарии отправки в реальном времени `zabbix_sender` будут собирать несколько значений, переданных ему с небольшим интервалом, и отправлять их на сервер в одном соединении. Значение, которое находится не дальше чем на 0.2 секунды от предыдущего значения, может быть помещено в тот же стек, но максимальное время опроса по-прежнему составляет 1 секунду.

> **Note:**
> Zabbix sender завершит работу, если в указанном файле конфигурации присутствует недопустимая запись параметра (не в формате `parameter=value`).

Запуск Zabbix sender с низкоуровневым обнаружением

Пример запуска Zabbix sender для отправки значения в формате JSON для низкоуровневого обнаружения:

```
./zabbix_sender -z 192.168.1.113 -s "Zabbix server" -k trapper.discovery.item -o '[{"{#FSNAME}":"/","{#FST
```

Чтобы это работало, правило низкоуровневого обнаружения должно иметь тип элемента данных *Zabbix trapper* (в этом примере — с ключом `trapper.discovery.item`).

**7. Get**

Обзор

Zabbix get — это утилита командной строки, которая может использоваться для подключения к Zabbix агенту и получения от него запрашиваемой информации.

Утилита обычно используется для диагностики Zabbix агентов.

Смотрите также библиотеку Python для Zabbix, которая имеет встроенный функционал для действий, аналогичных утилите Zabbix get.

Выполнение Zabbix get

Пример выполнения Zabbix get в UNIX для получения значения загрузки процессора с агента:

```
cd bin
./zabbix_get -s 127.0.0.1 -p 10050 -k system.cpu.load[all,avg1]
```

Ещё один пример выполнения утилиты Zabbix get для получения строки с веб-сайта:

```
cd bin
./zabbix_get -s 192.168.1.1 -p 10050 -k "web.page.regexp[www.example.com,,,\"USA: ([a-zA-Z0-9.-]+)\",,\1]"
```

Обратите внимание, что в этом примере ключ элемента данных содержит пробелы, поэтому для выделения ключа элемента данных используются двойные кавычки. Двойные кавычки не являются частью ключа элемента данных; они будут обрезаны оболочкой командной строки и не будут переданы Zabbix агенту.

Если ключ элемента данных не поддерживается, Zabbix get вернёт код завершения 1.

Zabbix get принимает следующие параметры командной строки:

```
-s --host <         IP>                    IP       .
-p --port <      >                           ,             (       : 10050).
-I --source-address <IP    >          IP                    .
-t --timeout <    >                          .           : 1-600     (       : 30    ).
-k -key <           >                      ,                    .
-P --protocol <    >               ,                      .     :
                                    auto —                        JSON,                  ,
                                    json —                        JSON;
                                    plaintext —                                    ,
-h --help                                          .
```

```
-V --version                                    .

--tls-connect <    >                            .     :
                                          unencrypted —                 (       );
                                          psk —                    TLS          (pre-shared key, PSK);
                                          cert —                   TLS        ;
--tls-ca-file <CA-  >                           ,              CA                              .
--tls-crl-file <CRL-  >                         ,                    .
--tls-agent-cert-issuer <    >                              .
--tls-agent-cert-subject <    >                             .
--tls-cert-file <cert-  >                       ,                          .
--tls-key-file <key-  >                         ,               (private key).
--tls-psk-identity <       -PSK>       (             )    ,                   PSK.
--tls-psk-file <PSK-  >                         ,              (pre-shared key).
--tls-cipher13 <   -   >                 OpenSSL 1.1.1          TLS 1.3.
--tls-cipher <   -   >                   GnuTLS (   TLS 1.2     )           OpenSSL (       TLS 1.2).
```

Смотрите также страницу помощи по Zabbix get для получения более подробной информации.

Zabbix get в Windows можно запустить аналогичным образом:

```
zabbix_get.exe [   ]
```

#### 8. JS

Обзор

zabbix_js — это утилита командной строки, которую можно использовать для тестирования встроенных скриптов.

Эта утилита выполняет пользовательский скрипт со строковым параметром и выводит результат. Скрипты выполняются с использованием встроенного механизма скриптинга Zabbix.

В случае ошибок компиляции или выполнения zabbix_js выведет сообщение об ошибке в stderr и завершит работу с кодом 1.

Использование

```
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
```

zabbix_js принимает следующие параметры командной строки:

```
-s, --script script-file
                                                  "-",                stdin
-i, --input input-file

                                          -p
                                                  "-",                stdin
-p, --param input-param                     :       ,
                                          -i
-l, --loglevel log-level
                                    : 0-5
-t, --timeout timeout                 -
                                         : 1-600      (       : 10     )
-h, --help
-V, --version
-w <webdriver url>
```

Примеры:

Пример 1: Выполнение скрипта с переменной, содержащей значение

Скрипт (script-file.js):

```
return value;
```

Использование:

```
zabbix_js -s script-file.js -p example_value
```

Вывод: `example_value`

Пример 2: Выполнение скрипта с файлом, содержащим входные данные

Файл со скриптом (`script-file.js`):

```
return value;
```

Файл с входными данными (`example.txt`):

```

```

Использование:

```
zabbix_js -s script-file.js -i example.txt
```

Вывод:

Пример 3: Выполнение скрипта с вводом, прочитанным из stdin

Файл со скриптом (`script-file.js`):

```
return value;
```

Использование:

```
zabbix_js -s script-file.js -i -
```

Ввод (stdin):

```
          stdin
```

Вывод: `stdin`

**9. Веб-сервис**

Обзор

Веб-служба Zabbix — это процесс, который используется для взаимодействия с внешними веб-службами. В настоящее время веб-служба Zabbix используется для создания и отправки запланированных отчетов, а в будущем планируется добавить дополнительную функциональность.

Сервер Zabbix подключается к веб-службе по HTTP(S). Для веб-службы Zabbix требуется, чтобы Google Chrome был установлен на том же узле сети; в некоторых дистрибутивах служба также может работать с Chromium (см. известные проблемы).

Установка

Официальный пакет zabbix-web-service доступен в репозитории Zabbix.

Чтобы собрать Zabbix web service из исходных кодов, укажите параметр конфигурации `--enable-webservice`.

Чтобы настроить Zabbix web service, измените параметры конфигурационного файла `zabbix_web_service.conf`.

> **Attention:**
> Настоятельно рекомендуется настроить шифрование между сервером Zabbix и Zabbix web service с использованием сертификатов. По умолчанию данные, передаваемые между сервером Zabbix и Zabbix web service, не шифруются, что может привести к несанкционированному доступу.

## 4. Установка и первые шаги

Для установки Zabbix с нуля выполните следующие действия:

1. Убедитесь, что необходимое программное обеспечение установлено и запущено.

2. Установите Zabbix.
3. Установите веб-интерфейс Zabbix.
4. Начните мониторинг.

Более подробная информация о каждом шаге приведена ниже.

**Необходимое программное обеспечение** Zabbix хорошо работает со следующим программным обеспечением (часто называемым стеком LAMP):

- **Операционная система Linux:** Ubuntu, Debian, CentOS или другие
- **Веб-сервер:** Apache или NGINX
- **Сервер баз данных:** MySQL/MariaDB или PostgreSQL
- **PHP**

На большинстве современных систем Linux PHP может быть уже установлен. В этом случае вам может потребоваться только настроить сервер баз данных и веб-сервер.

Более подробную информацию, например о поддерживаемых версиях программного обеспечения, см. в разделе Необходимое программное обеспечение.

**Установка Zabbix** **Самый простой** способ установки Zabbix — из **пакетов дистрибутива** (альтернативно, из контейнеров). Пакеты устанавливают незначительные зависимости программного обеспечения.

Точные инструкции по установке, адаптированные для вашей операционной системы, доступны на **сайте Zabbix**.

Выберите вашу платформу и следуйте приведенным инструкциям. Для начала рекомендуется установить сервер Zabbix, агент Zabbix 2 и веб-интерфейс. Позже вы можете добавить прокси Zabbix и дополнительные агенты.

Часть процесса установки — создание базы данных (для хранения данных конфигурации и истории) и импорт в нее начальных данных.

**Установка веб-интерфейса Zabbix** Выполните все шаги **мастера установки** веб-интерфейса.

Вам будет предложено ввести пароль базы данных (созданный на предыдущем шаге), чтобы веб-интерфейс мог взаимодействовать с базой данных.

**Начало мониторинга** Войдите в веб-интерфейс Zabbix.

Используйте **краткие руководства** для начала мониторинга популярных ресурсов (Linux, Apache, MySQL или других). В этих кратких руководствах объясняется, с какими шаблонами нужно работать, чтобы начать мониторинг популярных ресурсов.

Вы можете начать с руководства по мониторингу узла сети Linux.

### 1. Получение Zabbix

Обзор

Имеется четыре способа получения Zabbix:

- Установка из дистрибутивных пакетов
- Загрузка последней версии архива исходного кода и самостоятельная сборка
- Установка из контейнеров
- Загрузка готового решения Zabbix

Для загрузки последней версии дистрибутивных пакетов, исходного кода или готового решения Zabbix, перейдите на страницу загрузки Zabbix, на которой имеются прямые ссылки на последние доступные версии.

Получение исходного кода Zabbix

Существует несколько способов получить исходный код Zabbix:

- Вы можете скачать стабильные выпущенные версии с официального сайта Zabbix
- Вы можете скачать ночные сборки с страницы разработчиков на официальном сайте Zabbix
- Вы можете получить последнюю версию разработки из системы репозитория исходного кода Git:
  - Основное расположение полного репозитория находится по адресу https://git.zabbix.com/scm/zbx/zabbix.git
  - Master и поддерживаемые выпуски также зеркалируются на Github по адресу https://github.com/zabbix/zabbix

Для клонирования репозитория должен быть установлен Git-клиент. Официальный пакет командной строки Git-клиента обычно называется **git** в дистрибутивах. Например, чтобы установить его в Debian/Ubuntu, выполните:

```
sudo apt-get update
sudo apt-get install git
```

Чтобы получить весь исходный код Zabbix, перейдите в каталог, в который вы хотите поместить код, и выполните:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

## 2. Требования

Оборудование

### Память

Zabbix требуется как физическая память, так и память на диске. Очевидно, что объём необходимой памяти на диске зависит от количества наблюдаемых узлов сети и наблюдаемых параметров. Если вы планируете достаточно долгосрочное хранение истории наблюдаемых параметров, то потребуется по крайней мере несколько гигабайт для хранения данных истории в базе данных. Каждый из процессов демона Zabbix требует несколько подключений к серверу базы данных. Объём памяти, требуемый на каждое из подключений к базе данных, зависит от настроек базы данных.

**Note:**
Чем больше оперативной памяти вам доступно, тем быстрее работает база данных (а следовательно, и Zabbix).

### CPU

Zabbix и, в частности, база данных Zabbix могут требовать значительных ресурсов CPU в зависимости от количества контролируемых параметров и выбранного движка базы данных.

### Другое оборудование

Для использования встроенных в Zabbix SMS уведомлений потребуется последовательный порт передачи данных и GSM модем. Конвертер USB-to-serial также будет работать.

Примеры аппаратной конфигурации

В таблице приведены примеры аппаратной конфигурации при условии использования платформы **Linux/BSD/Unix**.

Это примеры размера и аппаратной конфигурации, с которых можно начать. Каждая установка Zabbix уникальна. Обязательно выполните тестирование производительности вашей системы Zabbix в тестовой или среде разработки, чтобы вы могли полностью понять свои требования до развертывания установки Zabbix в производственной среде.

| Размер установки | Контролируемые метрики[1] | Ядра CPU/vCPU | Память (GiB) | База данных | Amazon EC2[2] |
|---|---|---|---|---|---|
| Маленькая | 1 000 | 2 | 8 | MySQL Server, Percona Server, MariaDB Server, PostgreSQL | m6i.large/m6g.large |
| Средняя | 10 000 | 4 | 16 | MySQL Server, Percona Server, MariaDB Server, PostgreSQL | m6i.xlarge/m6g.xlarge |
| Большая | 100 000 | 16 | 64 | MySQL Server, Percona Server, MariaDB Server, PostgreSQL | m6i.4xlarge/m6g.4xlarge |
| Очень большая | 1 000 000 | 32 | 96 | MySQL Server, Percona Server, MariaDB Server, PostgreSQL | m6i.8xlarge/m6g.8xlarge |

[1] 1 метрика = 1 элемент данных + 1 триггер + 1 график<br> [2] Пример с универсальными экземплярами Amazon EC2, использующими архитектуру ARM64 или x86_64; при оценке и тестировании установки Zabbix перед

развертыванием в производственной среде следует выбрать подходящий тип экземпляра, например оптимизированный для вычислений/памяти/хранилища.

**Note:**
Фактическая конфигурация в значительной степени зависит от количества активных элементов данных и частоты обновления (подробности см. в разделе размер базы данных на этой странице). Для крупных установок настоятельно рекомендуется размещать базу данных на отдельном сервере.

Поддерживаемые платформы

В связи с требованиями безопасности и критически важным характером работы системы мониторинга, единственной операционной системой, которая может обеспечить необходимую производительность, отказоустойчивость и гибкость является операционная система UNIX. Zabbix работает на лидирующих на рынке версиях.

Компоненты Zabbix доступны и протестированы на следующих платформах:

| Платформа | Сервер | Агент | Агент 2 | Комментарий |
|---|---|---|---|---|
| Linux | x | x | x | |
| Windows | - | x | x | Zabbix агент поддерживается на всех версиях Windows для рабочих станций и серверов, начиная с XP (64-битная)/Server 2003. |
| Mac OS X | x | x | - | |
| IBM AIX | x | x | - | Zabbix агент не будет работать на платформах AIX ниже версий 6.1 TL07 / 7.1 TL01. |
| FreeBSD | x | x | - | |
| OpenBSD | x | x | - | |
| Solaris | x | x | - | |
| NetBSD | x | x | - | |
| HP-UX | x | x | - | |

**Note:**
Zabbix сервер/агент может работать также и на других Unix-подобных операционных системах.

**Attention:**
Zabbix отключает дампы памяти, если скомпилирован с шифрованием, и не запустится, если система не позволяет отключение дампов памяти.

Требуемое программное обеспечение

Zabbix построен на основе современных веб-серверов, ведущих систем управления базами данных и языка сценариев PHP.

Стороннее внешнее сопутствующее ПО

Если указано как обязательное, требуемое ПО/библиотека строго необходимо. Необязательные компоненты нужны для поддержки некоторых конкретных функций.

| Software | Mandatory status | Supported versions | Comments |
|---|---|---|---|
| *MySQL/Percona* | One of | 8.0.30-9.7.X | Требуется, если MySQL (или Percona) используется как backend-база данных Zabbix. Требуется движок InnoDB.<br><br>Добавлена поддержка версий MySQL:<br>- 9.5.X начиная с Zabbix 7.4.6;<br>- 9.6.X начиная с Zabbix 7.4.9;<br>- 9.7.X начиная с Zabbix 7.4.12.<br><br>Мы рекомендуем использовать библиотеку C API (libmysqlclient) для сборки server/proxy. |

| Software | Mandatory status | Supported versions | Comments |
|---|---|---|---|
| *MariaDB* | | 10.5.00-12.2.X | Требуется движок InnoDB.<br><br>Рекомендуемая версия — 11.4.<br><br>Мы рекомендуем использовать библиотеку MariaDB Connector/C для сборки server/proxy.<br><br>Добавлена поддержка версий MariaDB:<br>- 12.0.X начиная с Zabbix 7.4.3;<br>- 12.2.X начиная с Zabbix 7.4.9.<br><br>См. также: Возможные взаимные блокировки с MariaDB и Доступ к элементам интерфейса с MariaDB 10.5.1-10.5.9. |
| *PostgreSQL* | | 13.0-18.X | Требуется, если PostgreSQL используется как backend-база данных Zabbix.<br>В зависимости от размера установки может потребоваться увеличить параметр конфигурации PostgreSQL *work_mem* (значение по умолчанию — 4MB), чтобы объем памяти, используемый базой данных для конкретной операции, был достаточным и выполнение запросов не занимало слишком много времени.<br><br>Добавлена поддержка версий PostgreSQL:<br>- 18.X начиная с Zabbix 7.4.4. |
| *TimescaleDB* for PostgreSQL | | 2.13.0-2.27.X | Требуется, если TimescaleDB используется как расширение базы данных PostgreSQL. Убедитесь, что установлена Community Edition TimescaleDB, поддерживающая сжатие.<br><br>Обратите внимание, что PostgreSQL 15 поддерживается начиная с TimescaleDB 2.10.<br>Также см. документацию TimescaleDB для подробностей о совместимости версий PostgreSQL и TimescaleDB.<br><br>Добавлена поддержка версий TimescaleDB:<br>- 2.20.X, 2.21.X начиная с Zabbix 7.4.1;<br>- 2.22.X начиная с Zabbix 7.4.4;<br>- 2.23.X начиная с Zabbix 7.4.6;<br>- 2.24.X начиная с Zabbix 7.4.7;<br>- 2.25.X начиная с Zabbix 7.4.8;<br>- 2.26.X начиная с Zabbix 7.4.9;<br>- 2.27.X начиная с Zabbix 7.4.12. |
| *SQLite* | Optional | 3.3.5-3.53.X | SQLite поддерживается только с прокси Zabbix. Требуется, если SQLite используется как база данных прокси Zabbix.<br><br>Поддержка версий SQLite:<br>- 3.53.X начиная с Zabbix 7.4.9. |
| *Elasticsearch* | | 7.X | Elasticsearch поддерживается только с серверами Zabbix, исключительно для хранения исторических данных.<br>Поддержка Elasticsearch в настоящее время является экспериментальной. См. также требуемое ПО для server/proxy. |
| *smartmontools* | | 7.1 or later | Требуется для Zabbix agent 2. |
| *who* | | | Требуется для плагина подсчета пользователей. |
| *dpkg* | | | Требуется для плагина system.sw.packages. |
| *pkgtool* | | | Требуется для плагина system.sw.packages. |
| *rpm* | | | Требуется для плагина system.sw.packages. |
| *pacman* | | | Требуется для плагина system.sw.packages. |
| *q applets* | | | `qlist` и `qsize` как часть q applets требуются для плагина system.sw.packages в Gentoo Linux. |

**Note:**
Хотя Zabbix может работать с базами данных, доступными в операционных системах, для наилучшего опыта мы рекомендуем использовать базы данных, установленные из официальных репозиториев разработчика базы данных.

Веб-интерфейс

Минимальная поддерживаемая ширина экрана для веб-интерфейса Zabbix составляет 1200px.

Если указано как обязательное, требуемое программное обеспечение/библиотека строго необходимо. Необязательные нужны для поддержки некоторых конкретных функций.

| Software | Mandatory status | Supported versions | Comments |
|---|---|---|---|
| *PHP* | Yes | 8.0.0 - 8.5.X | Добавлена поддержка версий PHP:<br>- 8.5.X начиная с Zabbix 7.4.9. |
| *Apache*<br>*Nginx* | One of | 2.4 or later<br>1.20 or later | |
| *MySQL* | One of | See Third-party external surround-ing software | |
| *PostgreSQL* | | | |
| ***PHP extensions*** | | | |
| *mysqli* | Yes | | Требуется, если MySQL используется в качестве базовой базы данных Zabbix. |
| *pgsql* | | | Требуется, если PostgreSQL используется в качестве базовой базы данных Zabbix. |
| *bcmath* | | | php-bcmath (*--enable-bcmath*) |
| *mbstring* | | | php-mbstring (*--enable-mbstring*) |
| *sockets* | | | php-net-socket (*--enable-sockets*); требуется для поддержки пользовательских скриптов. |
| *gd* | | 2.0.28 or later | php-gd (если поставляется дистрибьютором как отдельный пакет); расширение PHP GD должно поддерживать изображения PNG (*--with-png-dir*), изображения JPEG (*--with-jpeg-dir*) и FreeType 2 (*--with-freetype-dir*). Может потребоваться версия 2.3.0 или новее, чтобы избежать возможного наложения текста на графиках для некоторых языков веб-интерфейса. |
| *libxml* | | 2.6.15 or later | php-xml (если поставляется дистрибьютором как отдельный пакет) |
| *xmlwriter* | | | php-xmlwriter (если поставляется дистрибьютором как отдельный пакет) |
| *xmlreader* | | | php-xmlreader (если поставляется дистрибьютором как отдельный пакет) |
| *ctype* | | | php-ctype (*--enable-ctype*) |
| *session* | | | php-session (если поставляется дистрибьютором как отдельный пакет) |
| *ldap* | No | | php-ldap; требуется для LDAP-аутентификации. |
| *openssl* | | | php-openssl; требуется для SAML-аутентификации. |
| *gettext* | | | php-gettext (*--with-gettext*); требуется для переводов. |
| *cURL* | | 7.19.4 or later | php-curl; требуется для Duo Universal Prompt MFA и SMTP authentication. |

Сторонние библиотеки веб-интерфейса, поставляемые с Zabbix:

| Library | Mandatory status | Supplied version | Comments |
|---|---|---|---|
| jQuery JavaScript Library | Yes | 3.6.0 | Библиотека JavaScript, упрощающая кроссбраузерную разработку. |
| jQuery UI | | 1.12.1 | Набор взаимодействий пользовательского интерфейса, эффектов, виджетов и тем, построенный на основе jQuery. |
| SAML PHP Toolkit | | 4.3.1 | Набор PHP-инструментов, добавляющий поддержку аутентификации SAML 2.0 для входа в Zabbix.<br><br>История поставляемых версий:<br>- 4.0.0 начиная с Zabbix 7.4.0;<br>- 4.3.1 начиная с Zabbix 7.4.8 (рекомендуется). |
| Duo Universal PHP library | | 1.1.2 | Библиотека PHP, добавляющая поддержку многофакторной аутентификации Duo Universal Prompt для Zabbix.<br><br>История поставляемых версий:<br>- 1.0.2 начиная с Zabbix 7.4.0;<br>- 1.1.0 начиная с Zabbix 7.4.7;<br>- 1.1.2 начиная с Zabbix 7.4.10 (рекомендуется). |
| Symfony Yaml Component | | 5.1.0 | Добавляет поддержку экспорта и импорта элементов конфигурации Zabbix в формате YAML. |

**Note:**
Zabbix также может работать на более ранних версиях Apache, MySQL и PostgreSQL.

**Attention:**
Для шрифтов, отличных от стандартного DejaVu, может потребоваться функция PHP imagerotate. Если она отсутствует, эти шрифты могут отображаться некорректно при показе графика. Эта функция доступна только если PHP скомпилирован со встроенным GD, чего нет в Debian и других дистрибутивах.

Сторонние библиотеки, используемые для написания и отладки кода веб-интерфейса Zabbix:

| Library | Mandatory status | Minimum version | Description |
|---|---|---|---|
| Composer | No | 2.4.1 | Менеджер пакетов уровня приложения для PHP, который предоставляет стандартный формат для управления зависимостями PHP-программного обеспечения и требуемых библиотек. |
| PHPUnit | | 8.5.29 | Фреймворк модульного тестирования PHP для тестирования веб-интерфейса Zabbix. |
| SASS | | 3.4.22 | Язык сценариев-препроцессор, который интерпретируется и компилируется в каскадные таблицы стилей (CSS). |

Веб-браузер на стороне клиента

Cookies и JavaScript должны быть включены.

Поддерживаются последние стабильные версии Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari и Opera.

**Warning:**
Реализована политика одного источника для IFrames, что означает, что Zabbix веб-интерфейс нельзя поместить во фреймы на другом домене.<br><br> Тем не менее, помещённые в Zabbix фреймы страницы будут иметь доступ к веб-интерфейсу Zabbix (посредством JavaScript), если помещённая во фрейм страница и веб-интерфейс Zabbix располагаются на одном домене. Такая станица как `http://secure-zabbix.com/cms/page.html`, при помещении в панели на `http://secure-zabbix.com/zabbix/`, будет иметь полный доступ к Zabbix посредством JS.

Сервер/прокси

Если указано как обязательное, требуемое программное обеспечение/библиотека строго необходимо. Необязательные нужны для поддержки некоторых конкретных функций.

| Requirement | Mandatory status | Description |
|---|---|---|
| *libpcre2* | Yes | Библиотека PCRE2 требуется для поддержки Perl Compatible Regular Expression (PCRE).<br>Поддерживается PCRE2 v10.x. |
| *libevent* | | Требуется для межпроцессного взаимодействия. Версия 2.0.10 или выше. |
| *libevent-pthreads* | | Требуется для межпроцессного взаимодействия. |
| *libpthread* | | Требуется для поддержки mutex и read-write lock (может входить в состав libc). |
| *libresolv* | | Требуется для разрешения DNS-имен (может входить в состав libc). |
| *libiconv* | | Требуется для преобразования текстовой кодировки/формата. Обязательно для Zabbix server в Linux. |
| *libz* | | Требуется для поддержки сжатия. |
| *libm* | | Математическая библиотека. Требуется только для Zabbix server. |
| *libmysqlclient* | One of | Требуется, если используется MySQL. |
| *libmariadb* | | Требуется, если используется MariaDB. |
| *libpq5* | | Требуется, если используется PostgreSQL; версия *libpq5* должна совпадать с версией используемой базы данных PostgreSQL или быть выше нее. |
| *libsqlite3* | | Требуется, если используется SQLite. Требуется только для Zabbix proxy. |
| *libOpenIPMI* | No | Требуется для поддержки IPMI. Требуется только для Zabbix server. |
| *libssh2* or *libssh* | | Требуется для проверок SSH. Версия 1.8.0 или выше (libssh2); 0.9.0 или выше (libssh). |
| *libcurl* | | Требуется для следующих функций:<br>- Мониторинг веб-сайтов, мониторинг VMware и элементы данных HTTP agent (для всех: версия 7.19.1 или выше);<br>- элементы данных Zabbix agent web.page.* (версия 7.19.1 или выше; см. также требования к агент);<br>- SMTP-аутентификация (Basic: версия 7.20.0 или выше; OAuth: версия 7.33 или выше; см. также требования к веб-интерфейс);<br>- Elasticsearch (версия 7.28.0 или выше).<br>Рекомендуется версия 7.28.0 или выше для всех функций.<br>Чтобы использовать обновленные возможности cURL для элементов данных `web.page.*`, перезапустите Zabbix server/proxy.<br>Для SMTP-аутентификации используйте пакет `libcurl-full` во время выполнения. |
| *libxml2* | | Требуется для мониторинга VMware и предварительной обработки XML XPath. |
| *net-snmp* | | Требуется для поддержки SNMP. Версия 5.3.0 или выше.<br>Поддержка надежных протоколов шифрования (AES192/AES192C, AES256/AES256C) доступна начиная с библиотеки net-snmp 5.8; в системах на базе RHEL 8+ рекомендуется использовать net-snmp 5.8.15 или более позднюю версию. |
| *libunixodbc* | | Требуется для мониторинга баз данных. |
| *libgnutls* or *libopenssl* | | Требуется при использовании шифрования.<br>Минимальные версии: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1 |
| *libldap* | | Требуется для поддержки LDAP. |
| *fping* | | Требуется для элементов данных ICMP ping. |
| *c-ares* | | Требуется для асинхронного разрешения DNS, если Zabbix настроен с опцией `--with-ares`. В противном случае будет использоваться *libevent*.<br>Минимальная версия: 1.16.0 |

Агент

| Requirement | Mandatory status | Description |
|---|---|---|
| *libpcre2* | Yes | Требуется библиотека PCRE2 для поддержки Perl Compatible Regular Expression (PCRE).<br>Поддерживается PCRE2 v10.x.<br>Требуется для мониторинга журналов. Также требуется в Windows. |
| *libpthread* | | Требуется для поддержки mutex и read-write lock (может быть частью libc). Не требуется в Windows. |
| *libresolv* | | Требуется для разрешения DNS (может быть частью libc). Не требуется в Windows. |

| Requirement | Mandatory status | Description |
|---|---|---|
| *libiconv* | | Требуется для преобразования текстовой кодировки/формата в UTF-8 в элементах данных журнала, содержимого файлов, регулярных выражений файлов и элементах данных regmatch (может быть частью libc). Не требуется в Windows. |
| *libgnutls* or *libopenssl* | No | Требуется при использовании encryption.<br>Минимальные версии: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1<br>В Microsoft Windows требуется OpenSSL 1.1.1 или более поздняя версия. |
| *libldap* | | Требуется, если используется LDAP. Не поддерживается в Windows. |
| *libcurl* | | Требуется для расширенной поддержки элементов данных Zabbix agent web.page.*.<br>Без libcurl доступна базовая функциональность (например, web.page.get[http://example.com]). С libcurl агент поддерживает дополнительные возможности, такие как HTTP URL с учетными данными (например, http://user:password@example.com) и HTTPS URL.<br>Требуется версия 7.19.1 или выше (рекомендуется 7.28.0 или выше).<br>Чтобы использовать обновленные возможности cURL, перезапустите Zabbix agent. |
| *libmodbus* | | Требуется только при использовании мониторинга Modbus.<br>Версия 3.0 или выше. |

Агент 2

| Requirement | Mandatory status | Description |
|---|---|---|
| *Go* | Yes | Требуется для сборки Zabbix agent 2 и его плагинов из исходного кода.<br>Поддерживается Go 1.24.10 или более поздняя версия. Инструкции по установке см. на go.dev.<br>Библиотеки Go, используемые Zabbix agent 2 и его плагинами, перечислены в репозитории Zabbix Git (библиотеки, помеченные как `indirect` в репозитории, являются зависимостями других требуемых библиотек):<br>- Zabbix agent 2<br>- Поддержка плагинов<br>- PostgreSQL<br>- MongoDB<br>- MSSQL<br>- Ember+<br>- NVIDIA GPU<br>- Пример плагина |
| *libpcre2* | Yes | Для поддержки регулярных выражений, совместимых с Perl (PCRE) требуется библиотека PCRE2.<br>Поддерживается PCRE2 v10.x.<br>Требуется для мониторинга журналов. Также требуется в Windows. |
| *libopenssl* | No | Требуется при использовании шифрования.<br>На платформах UNIX требуется OpenSSL 1.0.1 или более поздняя версия.<br>Библиотека OpenSSL должна иметь включенную поддержку PSK. LibreSSL не поддерживается.<br>В системах Microsoft Windows требуется OpenSSL 1.1.1 или более поздняя версия. |

Веб-сервис

Для создания отчётов по расписанию с использованием веб-сервиса Zabbix поддерживается последняя стабильная версия Google Chrome.

Версия Go, требуемая для построения веб-сервиса, совпадает с той, которая используется для Zabbix агента 2.

Java gateway

Если вы получили Zabbix из репозитория исходных кодов или скачали архив, то необходимые зависимости уже включены в дерево исходного кода.

Если вы получили Zabbix из пакетов вашего дистрибутива, то необходимые зависимости обеспечиваются системой управления пакетами.

В обоих вышеупомянутых случаях программное обеспечение готово к использованию, и скачивать какие-либо дополнительные файлы не нужно.

Однако, если вы хотите использовать другие версии этих зависимостей (например, если вы готовите пакет для определённого дистрибутива Linux), ниже приведён список версий библиотек, для которых подтверждена работоспособность Java gateway. Zabbix также может работать с другими версиями этих библиотек.

Следующая таблица содержит список JAR файлов, которые поставляются вместе в Java gateway в оригинальном коде:

| Библиотека | Признак обязательности | Поставляемая версия | Комментарии |
|---|---|---|---|
| android-json | Да | 4.3r1 | JSON (JavaScript Object Notation) — это легковесный формат обмена данными. Это совместимая с org.json реализация Android, которая была извлечена из Android SDK. |
| logback-classic | | 1.5.16 | |
| logback-core | | 1.5.16 | |
| slf4j-api | | 2.0.16 | |

Java gateway может быть скомпилирован с использованием Oracle Java или с использованием OpenJDK (версия 1.6 или новее) с открытым исходным кодом. Пакеты, поставляемые Zabbix, скомпилированы с использованием OpenJDK. Следующая таблица перечисляет пакеты OpenJDK, использованные для сборки пакетов Zabbix в зависимости от дистрибутива:

| Дистрибутив | Пакет OpenJDK |
|---|---|
| AlmaLinux 9 | `java-11-openjdk-devel` (amd64: 11.0.19.0.7-4; arm64: 11.0.20.0.8-3) |
| 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 9 | `java-11-openjdk-devel` (amd64, arm64: 11.0.18.0.10-3) |
| 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 12 | `default-jdk-headless` (amd64, arm64: 2:1.17-74) |
| Debian 11 | `default-jdk-headless` (amd64: 2:1.11-72) |
| 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 9 | `java-11-openjdk-devel` (amd64: 11.0.19.0.7-4.0.1; arm64: 11.0.20.0.8-2.0.1) |
| 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 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 9 | `java-11-openjdk-devel` (amd64: 11.0.19.0.7-4; arm64: 11.0.20.0.8-3) |
| 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 9 | `java-11-openjdk-devel` (amd64: 11.0.19.0.7-4; arm64: 11.0.20.0.8-3) |
| Rocky Linux 8 | `java-1.8.0-openjdk-devel` (amd64: 1.8.0.372.b07-4; arm64: 1.8.0.382.b05-2) |
| 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 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) |

Номера портов по умолчанию

Для конфигурации по умолчанию применим следующий список открытых портов по компонентам:

| Компонент Zabbix | Номер порта | Протокол | Тип подключения |
|---|---|---|---|
| Zabbix агент | 10050 | TCP | по требованию |
| Zabbix агент 2 | 10050 | TCP | по требованию |
| Zabbix сервер | 10051 | TCP | по требованию |

| Компонент Zabbix | Номер порта | Протокол | Тип подключения |
|---|---|---|---|
| Zabbix прокси | 10051 | TCP | по требованию |
| Zabbix Java gateway | 10052 | TCP | по требованию |
| Zabbix веб-сервис | 10053 | TCP | по требованию |
| Zabbix веб-интерфейс | 80 | HTTP | по требованию |
| | 443 | HTTPS | по требованию |
| Zabbix траппер | 10051 | TCP | по требованию |

**Note:**
Эти номера портов должны быть открыты в брандмауэре, чтобы подключения Zabbix были разрешены. Исходящие TCP подключения обычно не требуют явных настроек брандмауэра.

Размер базы данных

Данные конфигурации Zabbix требуют фиксированного количества дискового пространства и сильно не увеличиваются.

Размер базы данных Zabbix в основном зависит от следующих переменных, которые определяют объём хранимых исторических данных:

- Количество обрабатываемых значений в секунду

Это среднее количество новых значений, которые Zabbix сервер получает каждую секунду. Например: если имеется 3000 элементов данных с интервалом проверки 60 секунд, то количество обрабатываемых запросов за секунду рассчитывается 3000/60 = **50**.

Это означает, что каждую секунду в базу данных Zabbix добавляется 50 новых значений.

- Настройки очистки истории

Zabbix хранит значения за определённый период времени, обычно несколько недель или месяцев. Каждое новое значение требует определенного объёма дискового пространства для данных и индексов.

Таким образом, если требуется хранение 30 дней истории и каждую секунду мы получаем 50 новых значений, общее количество значений будет приблизительно (**30***24*3600)* **50** = 129.600.000 или около 130M значений.

В зависимости от типа базы данных, типа полученных значений (с плавающей точкой, целое число, строки, файлы журналов и т.д.) может потребоваться от 40 байт до сотен байт дискового пространства для хранения одного значения. Обычно одно значение занимает около 90 байт для числового элемента данных[2]. В нашем случае это означает, что 130M значений потребуют 130M * 90 байт = **10.9ГБ** дискового пространства.

**Note:**
Размер значений текстовых/журнальных элементов данных невозможно предугадать точно, но можно ожидать около 500 байт на значение.

- Настройки очистки динамики изменений

Zabbix хранит ежечасную статистику значений max/min/avg/count для каждого элемента данных в таблице **trends**. Эти данные используются для отслеживания динамики изменений и для графиков при отображении большого периода времени. Период в 1 час не является настраиваемым.

Базе данных Zabbix, в зависимости от типа базы данных, требуется около 90 байт на один элемент динамики изменений. Предположим, что требуется хранить динамику изменений в течение 5 лет. Значения 3000 элементов данных потребуют 3000*24*365* **90** = **2.2ГБ** за год, или **11ГБ** за 5 лет.

- Настройки очистки событий

Каждое событие Zabbix требует около 250 байт дискового пространства[1]. Сложно точно оценить количество событий, ежедневно генерируемых Zabbix сервером. В самом худшем случае, можно предположить, что Zabbix генерирует одно событие в секунду.

По каждому событию восстановления создается запись в event_recovery. Обычно большая часть событий восстанавливается, поэтому мы можем предположить, что в event_recovery будет по одной записи по каждому событию. Это означает дополнительные 80 байт на каждое событие.

Опционально, у событий могут быть теги, каждая запись тега требует приблизительно 100 байт дискового пространства[1]. Количество тегов на событие (#теги) зависит от конфигурации. Таким образом, каждая запись потребует дополнительно #теги * 100 байт дискового пространства.

Это означает, что для того, чтобы хранить события 3 года, потребуется 3*365*24*3600* (250+80+#tags*100) = **~30ГБ**+#теги*100 байт дискового пространства[2].

**Note:**
[1] Больше, когда имеются не-ASCII имена событий, тегов и значения.<br> [2] Приблизительные размеры основаны на MySQL и могут отличаться для других баз данных.

Представленная ниже таблица содержит формулы для расчёта требуемого дискового пространства для системы мониторинга Zabbix:

| Параметр | Формула для расчёта занимаемого места(в байтах) |
|---|---|
| *Конфигурация Zabbix* | Фиксированный размер. Обычно 10МБ или меньше. |
| *История* | дней*(элементов данных/частота обновления)*24*3600*байт<br>элементы данных : количество элементов данных<br>дней : количество дней хранения истории<br>частота обновления : среднее значение периода проверки элементов данных<br>байт : количество байт, требуемых для одного значения, зависит от типа базы данных, около 90 байт |
| *Динамика изменений* | дней*(элементов данных/3600)*24*3600*байт<br>элеменов данных : количество элементов данных<br>дней : количество дней хранения динамики изменений<br>байт : количество байт, требуемых для одного значения, зависит от типа базы данных, около 90 байт. |
| *События* | дней*событий*24*3600*байт<br>событий : количество событий в секунду. Одно (1) событие в секунду в худшем случае.<br>дней : количество дней хранения событий<br>байт : количество байт, требуемых для одного события, зависит от типа базы данных, обычно примерно 330 + среднее количество тегов на событие * 100 байт. |

Итак, общее количество требуемого места на жёстком диске рассчитывается:

**Конфигурация + История + Динамика изменений + События**

Дисковое пространство НЕ будет использовано сразу же после установки Zabbix. Размер базы данных будет постепенно увеличиваться и остановится по достижении определённого момента, зависящего от настроек очистки базы данных.

Синхронизация времени

Очень важно иметь точное системное время на сервере с запущенным Zabbix. ntpd [en] — один из самых популярных демонов синхронизации времени узла с временем на остальных серверах. Настоятельно рекомендуется поддерживать синхронизированное время на всех системах, где работают компоненты Zabbix.

Требования к сети

Следующий список открытых портов по компонентам применим к конфигурации по умолчанию.

| Компоненты | Порт |
|---|---|
| Веб-интерфейс | http на 80, https на 443 |
| Сервер | 10051 (для использования с активными прокси/агентами) |
| Активный прокси | 10051 |
| Пассивный прокси | 10051 |
| Агент 2 | 10050 |
| Траппер | |
| JavaGateway | 10052 |
| Веб-сервис | 10053 |

**Note:**
Чтобы разрешить внешние коммуникации с Zabbix, номера портов должны быть открыты на брандмауэре. Исходящие TCP-соединения обычно не требуют явных настроек брандмауэра.

### 3. Установка из исходных кодов

Вы можете получить самую свежую версию Zabbix, скомпилировав её из исходных кодов.

Здесь представлено пошаговое руководство по установке Zabbix из исходных кодов.

Установка демонов Zabbix

1 Загрузите исходный архив

Перейдите на страницу загрузки Zabbix и загрузите исходный архив. После загрузки распакуйте исходные файлы, выполнив:

```
tar -zxvf zabbix-7.4.0.tar.gz
```

> **Note:**
> Укажите в команде правильную версию Zabbix. Она должна соответствовать имени загруженного архива.

2. Создайте учётную запись пользователя

Все процессы демонов Zabbix запускаются под учётной записью непривилегированного системного пользователя. Если демон Zabbix запущен под учётной записью непривилегированного пользователя, он продолжит работу под этим пользователем.

В конфигурации по умолчанию, если демон запущен как `root`, он переключится на учётную запись пользователя `zabbix`, которая должна существовать. Чтобы создать системного пользователя и группу `zabbix`, выполните команды, перечисленные ниже.

Системы на основе RedHat:

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

Системах на основе Debian:

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

Для веб-интерфейса Zabbix создавать отдельную учетную запись пользователя не нужно.

Рекомендации по безопасности

Если Zabbix сервер и агент работают на одной машине, то рекомендуется запускать их под **отдельными учётными записями**. Запуск сервер и агента от имени одного и того же пользователя позволяет агенту получить доступ к файлу конфигурации сервера, что может раскрыть конфиденциальную информацию, например пароль базы данных, любому пользователю уровня Администратора в Zabbix.

> **Attention:**
> Запуск Zabbix из-под `root`, `bin` или любой другой учётной записи со специальными правами является риском для безопасности.

Домашний каталог (необязательно)

Zabbix процессам не требуется домашний каталог, поэтому его создание обычно не рекомендуется. Однако, если вы используете некоторый функционал, которому требуется наличие домашнего каталога (например, хранение учётных данных MySQL в `$HOME/.my.cnf`), вы можете создать его с помощью команд, перечисленных ниже.

На системах на основе RedHat, выполните:

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

На системах на основе Debian, выполните:

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

3. Создайте базу данных Zabbix

Для демонов Zabbix сервера и прокси, а так же для веб-интерфейса Zabbix, требуется база данных. Она не требуется для работы Zabbix агента.

Для создания схемы базы данных и вставки набора данных предоставляются SQL скрипты. Для базы данных Zabbix прокси требуется только создание схемы данных, в то время как для Zabbix сервера помимо схемы необходимо также загрузить набор данных поверх схемы.

После создания базы данных Zabbix, перейдите к следующим шагам компиляции Zabbix.

4 Настройка исходных кодов

Для сборки Zabbix server, Zabbix proxy или Zabbix agent требуется C99 с расширениями GNU. Эту версию можно явно указать, задав CFLAGS="-std=gnu99":

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

**Note:**
Если установка выполняется из репозитория Zabbix Git, сначала необходимо выполнить:
`./bootstrap.sh`

При настройке исходных кодов для Zabbix server или proxy необходимо указать используемый тип базы данных. За один раз с процессом server или proxy может быть скомпилирован только один тип базы данных.

Чтобы увидеть все поддерживаемые параметры конфигурации, в каталоге распакованных исходных кодов Zabbix выполните:

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

Чтобы настроить исходные коды для Zabbix server и agent, можно выполнить примерно такую команду:

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

Чтобы настроить исходные коды для Zabbix server (с PostgreSQL и т. д.), можно выполнить:

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

Чтобы настроить исходные коды для Zabbix proxy (с SQLite и т. д.), можно выполнить:

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

Чтобы настроить исходные коды для Zabbix agent, можно выполнить:

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

или, для Zabbix agent 2:

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

**Note:**
Для сборки Zabbix agent 2 должна быть установлена поддерживаемая версия Go.

Примечания по параметрам компиляции:

- `--enable-agent` - компилирует Zabbix agent, а также командные утилиты Zabbix get и Zabbix sender.
- `--with-libcurl` - требуется для мониторинга виртуальных машин, аутентификации SMTP и элементов данных `web.page.*` Zabbix agent. См. также: Требования (libcurl).
- `--with-libxml2` - требуется для мониторинга виртуальных машин.
- `--with-libpcre2[=DIR]` - Zabbix всегда компилируется с библиотекой PCRE2; этот параметр позволяет только указать пользовательский путь установки PCRE2.
- `--with-mysql=/path/to/mysql_config` - указывает путь к определенной конфигурации клиентской библиотеки MySQL. Полезно, когда установлено несколько версий MySQL или MariaDB.
- `--enable-static` - статически связывает библиотеки (не поддерживается в Solaris). Используйте этот параметр, если планируете распространять скомпилированные двоичные файлы для систем без необходимых библиотек. Не рекомендуется при сборке Zabbix server. Чтобы собрать server статически, требуется статическая версия каждой внешней библиотеки. Скрипт configure не проверяет это автоматически.
- `--with-stacksize=<value>` - задает размер стека на поток в килобайтах (например, `--with-stacksize=512`). Вы можете увеличить это значение, если Zabbix аварийно завершает работу или зависает из-за переполнения стека (например, во время предобработки в системах с низкими значениями ограничения размера стека потоков по умолчанию).

**Attention:**
Если `./configure` завершается с ошибкой из-за отсутствующих библиотек или других проблем, проверьте файл `config.log` для получения подробной информации об ошибке.
Например, если отсутствует `libssl`, непосредственное сообщение об ошибке может вводить в заблуждение:
```
checking for main in -lmysqlclient... no
configure: error: Not found mysqlclient library
```
В этом случае `config.log` показывает истинную причину:
```
/usr/bin/ld: cannot find -lssl
/usr/bin/ld: cannot find -lcrypto
```

См. также:

- Сборка Zabbix с поддержкой шифрования
- Известные проблемы компиляции

5 Сборка и установка всего

**Note:**
Если выполняется установка из репозитория Zabbix Git, сначала необходимо выполнить:
```
$ make dbschema
```

```
make install
```

Этот шаг следует выполнять от имени пользователя с достаточными правами (обычно `root` или с использованием `sudo`).

При выполнении `make install` по умолчанию будут установлены двоичные файлы демона (zabbix_server, zabbix_agentd, zabbix_proxy) в /usr/local/sbin, а двоичные файлы клиента (zabbix_get, zabbix_sender) в /usr/local/bin.

**Note:**
Чтобы указать другое расположение вместо `/usr/local`, используйте ключ `--prefix` на предыдущем шаге настройки исходных текстов, например, `--prefix=/home/zabbix`. В этом случае двоичные файлы демона будут установлены в `<prefix>/sbin`, а утилиты — в `<prefix>/bin`. Страницы man будут установлены в `<prefix>/share`.

6. Просмотрите и отредактируйте файлы конфигурации

- отредактируйте файл конфигурации Zabbix агента **/usr/local/etc/zabbix_agentd.conf**

Вам нужно сконфигурировать этот файл на каждом узле сети, на котором установлен zabbix_agentd.

В файле вы должны указать **IP адрес** Zabbix сервера. Подключения с остальных хостов будут отклонены.

- отредактируйте файл конфигурации Zabbix сервера **/usr/local/etc/zabbix_server.conf**

Вы должны указать имя базы данных, пользователя и пароль (если используется).

Остальные параметры подойдут со значениями по умолчанию, если у вас небольшая инсталляция (до десятка наблюдаемых узлов сети). Вы должны изменить параметры по умолчанию, если вы хотите увеличить производительность Zabbix сервера (или прокси).

- если вы установили Zabbix прокси, то отредактируйте файл конфигурации Zabbix прокси **/usr/local/etc/zabbix_proxy.conf**

Вы должны указать IP адрес сервера и имя прокси (должно быть известно серверу), а также имя базы данных, пользователя и пароль (если используется).

**Note:**
В случае SQLite необходимо указать полный путь к файлу базы данных; пользователь и пароль к БД не требуются.

7. Запустите демоны

Запустите zabbix_server на стороне сервера.

```
zabbix_server
```

**Note:**
Удостоверьтесь, что в вашей системе разрешено выделение 36МБ (или немногим больше) разделяемой памяти, в противном случае сервер может не запуститься и вы увидите сообщение: «Cannot allocate shared memory for <type of cache>», — в файле журнала сервера. Такое может случится в FreeBSD, Solaris 8.

Запустите zabbix_agentd на всех наблюдаемых машинах.

```
zabbix_agentd
```

**Note:**
Убедитесь, что ваша система позволяет выделить 2 МБ разделяемой памяти (shared memory), в противном случае агент может не запуститься и вы увидите сообщение: «Cannot allocate shared memory for collector», — в журнале агента. Такое может произойти в Solaris 8.

Если вы установили Zabbix прокси, запустите zabbix_proxy.

```
zabbix_proxy
```

Установка веб-интерфейса Zabbix

Копирование PHP-файлов

Веб-интерфейс Zabbix написан на PHP, поэтому для его запуска нужен веб-сервер с поддержкой PHP.
Установка выполняется простым копированием PHP-файлов из каталога ui в каталог HTML-документов веб-сервера.

Распространенные расположения каталогов HTML-документов для веб-серверов Apache включают:

- /usr/local/apache2/htdocs (каталог по умолчанию при установке Apache из исходного кода)
- /srv/www/htdocs (OpenSUSE, SLES)
- /var/www/html (Debian, Ubuntu, Fedora, RHEL)

Рекомендуется использовать подкаталог вместо корня HTML.
Чтобы создать подкаталог и скопировать в него файлы веб-интерфейса Zabbix, выполните следующие команды, заменив <htdocs> на фактический каталог:

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

Если вы планируете использовать любой язык, кроме английского, см. Установка дополнительных языков веб-интерфейса для получения инструкций.

Установка веб-интерфейса

См. страницу Установка веб-интерфейса для получения информации о мастере установки веб-интерфейса Zabbix.

Установка загружаемых плагинов Zabbix agent 2

Установка загружаемых плагинов Zabbix agent 2 требуется только в том случае, если вы хотите отслеживать цели, не охватываемые встроенными плагинами (например, серверы или кластеры MongoDB, PostgreSQL и его форки и т. д.). См. полный список загружаемых плагинов и встроенных плагинов.

**Attention:**
Перед установкой плагина, пожалуйста, ознакомьтесь с его файлом README. Он может содержать особые требования и инструкции по установке.

Чтобы установить из исходных кодов, сначала скачайте и распакуйте архив исходных кодов загружаемого плагина.

Чтобы скомпилировать плагин, перейдите в каталог распакованного плагина и выполните `make`:

```
make
```

**Note:**
Для сборки загружаемых плагинов Zabbix agent 2 должна быть установлена поддерживаемая версия Go.

Исполняемый файл плагина можно разместить в любом месте, если он может быть загружен Zabbix agent 2. Укажите путь к двоичному файлу плагина в файле конфигурации плагина, например в postgresql.conf для плагина PostgreSQL:

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

Путь к файлу конфигурации плагина должен быть указан в параметре Include файла конфигурации Zabbix agent 2:

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

Перейдите к разделу настройка для получения дополнительных сведений о настройке плагинов.

Загружаемые плагины, предоставляемые Zabbix, используют простые makefile со следующими целями сборки:

- `make` - собрать плагин
- `make clean` - удалить все файлы, созданные при сборке плагина
- `make check` - запустить самотестирование (требуется реальная цель мониторинга, например база данных PostgreSQL)
- `make style` - проверить стиль кода Go с помощью `golangci-lint`
- `make format` - форматировать код Go с помощью `go fmt`
- `make dist` - создать архив исходных кодов, включая все зависимости

Установка Java gateway

Устанавливать Java gateway требуется только в том случае, если вы хотите отслеживать приложения JMX. Java gateway является легковесным и не требует базы данных.

Чтобы установить из исходных кодов, сначала скачайте и распакуйте архив с исходным кодом.

Чтобы скомпилировать Java gateway, выполните скрипт `./configure` с опцией `--enable-java`. Рекомендуется указать опцию `--prefix`, чтобы задать путь установки, отличный от значения по умолчанию /usr/local, поскольку при установке Java gateway будет создано целое дерево каталогов, а не только один исполняемый файл.

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

Чтобы скомпилировать и упаковать Java gateway в файл JAR, выполните `make`. Обратите внимание, что для этого шага в вашем пути должны быть доступны исполняемые файлы `javac` и `jar`.

```
make
```

Теперь у вас есть файл zabbix-java-gateway-$VERSION.jar в каталоге src/zabbix_java/bin. Если вам удобно запускать Java gateway из каталога src/zabbix_java в каталоге дистрибутива, тогда вы можете перейти к инструкциям по настройке и запуску Java gateway. В противном случае убедитесь, что у вас достаточно прав, и выполните `make install`.

```
make install
```

Перейдите к настройке для получения дополнительных сведений о настройке и запуске Java gateway.

Установка веб-сервиса Zabbix

Установка веб-сервиса Zabbix требуется, только если вы хотите использовать отчёты по расписанию.

Чтобы установить из исходного кода, сначала загрузите и извлеките исходный код из архива.

Чтобы скомпилировать веб-сервис Zabbix, выполните скрипт `./configure` с опцией `--enable-webservice`.

> **Note:**
> Для сборки веб-сервиса Zabbix должна быть установлена поддерживаемая версия Go.

Выполните zabbix_web_service на машине, где был установлен веб-сервис:

```
zabbix_web_service
```

Перейдите к установке для получения более подробных сведений о настройке генерирования Регулярных отчётов.

### 1. Сборка Zabbix агента на Windows

Обзор

На этой странице показано, как собрать агент Zabbix из исходного кода в Windows 10 (64-разрядной версии).

Эти инструкции применимы к версиям Windows, поддерживающим Visual Studio 2022.

Для сборки агента Zabbix требуется:

- Компилятор C (входит в состав инструментов сборки (Build Tools) для Visual Studio 2022)
- OpenSSL (для функций шифрования в Zabbix)
- PCRE2 (Perl-совместимые регулярные выражения — Perl Compatible Regular Expressions; для функций сопоставления шаблонов регулярных выражений в Zabbix)

Вы можете собрать агента Zabbix одним из следующих методов:

- С использованием vcpkg — автоматизированный подход, упрощающий управление зависимостями с помощью менеджера пакетов C++.
- Ручная сборка — ручной подход, требующий установки всех зависимостей перед компиляцией агента.

В зависимости от ваших потребностей в мониторинге могут потребоваться дополнительные библиотеки. Подробнее смотрите Требования.

**Attention:**
Перед началом сборки учтите следующее:<br><br>
- Для выполнения команд используйте командную строку x64 Native Tools (входит в состав Build Tools for Visual Studio 2022), запущенную пользователем с правами на запись в защищённые папки.
- Рекомендуется создать рабочий каталог `C:\Zabbix` для всех исходных файлов и папок сборки. Однако скомпилированные компоненты следует устанавливать в `C:\Program Files\Zabbix\x64`.

Сборка агента Zabbix с помощью vcpkg

В этом разделе приведены инструкции по сборке агента Zabbix с помощью vcpkg — менеджера пакетов, который упрощает управление зависимостями и интеграцию с проектами на C++.

1. Скачайте и установите Build Tools for Visual Studio 2022. Во время установки обязательно выберите рабочую нагрузку *Desktop development with C++*, которая включает инструменты, необходимые для сборки агента с помощью vcpkg:

- C compiler (Microsoft Visual C++)
- NMake command-line tool
- vcpkg package manager
- x64 Native Tools Command Prompt

2. Инициализируйте vcpkg и установите зависимости, необходимые для сборки агента Zabbix (учтите, что это может занять некоторое время):

```
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. Скачайте архив исходного кода Zabbix и распакуйте его в `C:\Zabbix\zabbix-7.4.0`.

4. Перейдите в каталог сборки Zabbix (`C:\Zabbix\zabbix-7.4.0\build\win32\project`) и создайте следующий скрипт `build.bat`; убедитесь, что правильно указаны каталоги, в которых установлены OpenSSL и PCRE2:

```
:: Set vcpkg installation path:
set vcpkg=C:\Program Files\Zabbix\x64\x64-windows-static

:: Run the build process:
nmake -f Makefile CPU=AMD64 ^
    PCRE2INCDIR="%vcpkg%\include" ^
    PCRE2LIBDIR="%vcpkg%\lib" ^
    TLS=openssl ^
    TLSINCDIR="%vcpkg%\include" ^
    TLSLIBDIR="%vcpkg%\lib" ^
    LIBS="$(LIBS) Crypt32.lib" ^
    all
```

5. Скомпилируйте агент Zabbix, выполнив скрипт:

```
build.bat
```

После компиляции двоичные файлы компонентов Zabbix будут находиться в `C:\Zabbix\zabbix-7.4.0\bin\win64`. Файл конфигурации агента Zabbix находится в `C:\Zabbix\zabbix-7.4.0\conf`.

Чтобы запустить агент, скопируйте `zabbix_agent.exe` и его файл конфигурации в отдельную папку (например, `C:\Zabbix\agent`), а затем запустите агент:

```
mkdir C:\Zabbix\agent
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agent.exe C:\Zabbix\agent\
copy C:\Zabbix\zabbix-7.4.0\conf\zabbix_agent.win.conf C:\Zabbix\agent\
```

```
C:\Zabbix\agent\zabbix_agent.exe -c C:\Zabbix\agent\zabbix_agent.win.conf
```

Сборка Zabbix агента вручную

**Attention:**
Этот метод сборки агента Zabbix подходит пользователям, которым требуется полный контроль над средой сборки или которые находятся в ограниченной среде, где использование vcpkg невозможно.

В этом разделе содержатся инструкции по ручной сборке агента Zabbix, включая установку необходимых инструментов сборки и зависимостей (Perl, OpenSSL, PCRE2), а также компиляцию агента.

Установка инструментов сборки

1. Скачайте и установите [Инструменты сборки (Build Tools) для Visual Studio 2022] (https://visualstudio.microsoft.com/downloads/#build-tools-for-visual-studio-2022).

Во время установки обязательно выберите рабочую нагрузку *Desktop development with C++*, которая включает в себя инструменты, необходимые для ручной сборки агента:

- Компилятор C (Microsoft Visual C++)
- Инструмент командной строки NMake
- Командная строка x64 Native Tools Command Prompt

Установка OpenSSL

**Note:**
Чтобы скомпилировать агент Zabbix без поддержки TLS, перейдите к разделу Установка PCRE2.

1. Скачайте и установите Strawberry Perl (доступен в виде MSI-установщика).

Во время установки обязательно укажите `C:\Zabbix\Strawberry` в качестве папки для установки.

2. Установите модуль Perl `Text::Template`:

```
cpanm Text::Template
```

3. Убедитесь, что Netwide Assembler (NASM; требуется для компиляции OpenSSL) был скомпилирован во время установки Strawberry Perl:

```
nasm -v
#### NASM version 2.16.01 compiled on May3 2024
```

Если NASM не скомпилирован, установите его вручную. Подробнее см. в документации NASM [en].

4. Скачайте архив исходного кода OpenSSL и распакуйте его в `C:\Zabbix\openssl-3.5.0`.

5. Перейдите в извлечённый каталог и настройте OpenSSL, например:

```
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:**
Если вы выбрали пользовательский каталог для OpenSSL при компиляции Zabbix агента в Windows (например, `C:\zabbix` или `C:\openssl-64bit`), обязательно отзовите права на запись в этот каталог для пользователей без прав администратора. В противном случае агент будет загружать настройки SSL из пути, который могут изменять непривилегированные пользователи, что создаст потенциальную уязвимость безопасности.

- Параметр `no-shared` делает статические библиотеки OpenSSL libcrypto.lib и libssl.lib самодостаточными, поэтому исполняемые файлы Zabbix включают OpenSSL без необходимости использования внешних DLL-библиотек. Это означает, что исполняемые файлы Zabbix можно копировать на другие компьютеры Windows без библиотек OpenSSL; однако при выпуске новой версии с исправлением ошибок OpenSSL потребуется перекомпилировать агент Zabbix.
- Без параметра `no-shared` Zabbix использует DLL-библиотеки OpenSSL во время работы. Это означает, что обновления OpenSSL могут не требовать перекомпиляции агента Zabbix; однако при копировании на другие компьютеры необходимо также скопировать DLL-библиотеки OpenSSL.

Для получения дополнительной информации о других параметрах конфигурации OpenSSL обратитесь к документации OpenSSL [en].

6. Скомпилируйте OpenSSL и запустите тесты (обратите внимание, что это может занять некоторое время):

**Attention:**
Запускайте тесты без прав администратора; в противном случае это может привести к непредвиденным результатам или угрозам безопасности. Если некоторые тесты не пройдут, обратитесь к документации OpenSSL [en] для устранения неполадок.

```
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. Установите OpenSSL:

```
nmake install
```

Чтобы установить только программные компоненты (библиотеки, заголовочные файлы, но без документации), можно использовать `nmake install_sw`.

Установка PCRE2

1. Скачайте и установите [CMake] (https://cmake.org/download/) (доступен в виде MSI-установщика).

Во время установки обязательно укажите `C:\Zabbix\CMake` в качестве папки установки и выберите опцию *Add CMake to the PATH environment variable (Добавить CMake в переменную среды PATH)*.

2. Скачайте [архив исходного кода PCRE2] (https://github.com/PCRE2Project/pcre2/releases/latest) и распакуйте его в `C:\Zabbix\pcre2-10.45`.

3. Создайте каталог `build` в распакованном каталоге PCRE2 и перейдите в него:

```
mkdir C:\Zabbix\pcre2-10.45\build
cd C:\Zabbix\pcre2-10.45\build
```

4. Настройте PCRE2:

```
cmake -G "NMake Makefiles" -DPCRE_SUPPORT_UNICODE_PROPERTIES=ON -DCMAKE_BUILD_TYPE=Release -DCMAKE_INSTALL
```

**Note:**
При возникновении ошибок рекомендуется удалить кэш CMake перед повторной попыткой сборки CMake. Кэш (`CMakeCachecache.txt`) находится в каталоге build извлечённого каталога PCRE2.

5. Соберите PCRE2 с помощью NMake:

```
nmake
```

6. Установите PCRE2:

```
cmake --install .
```

Сборка агента Zabbix

1. Скачайте исходный архив Zabbix и распакуйте его в `C:\Zabbix\zabbix-7.4.0`.

Если вам нужно создать исходный архив из исходного репозитория (например, чтобы применить собственные исправления или собрать из последней версии исходного кода), выполните следующие команды на машине с **Linux**:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure --enable-agent --enable-ipv6 --prefix=`pwd`
make dist
```

Это создаст исходный архив, который затем можно скопировать на машину с Windows.

2. Перейдите в каталог сборки Zabbix и скомпилируйте агент Zabbix (или другие компоненты); убедитесь, что правильно указаны каталоги, в которых установлены OpenSSL и PCRE2:

```
cd C:\Zabbix\zabbix-7.4.0\build\win32\project

#### 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

####         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
```

После компиляции исполняемые файлы компонентов Zabbix будут находиться в `C:\Zabbix\zabbix-7.4.0\bin\win64`. Файл конфигурации агента Zabbix находится в `C:\Zabbix\zabbix-7.4.0\conf`.

Чтобы запустить агент, скопируйте `zabbix_agent.exe` и его файл конфигурации в отдельную папку (например, `C:\Zabbix\agent`), а затем запустите агент:

```
mkdir C:\Zabbix\agent
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agentd.exe C:\Zabbix\agent\
copy C:\Zabbix\zabbix-7.4.0\conf\zabbix_agentd.win.conf C:\Zabbix\agent\

C:\Zabbix\agent\zabbix_agentd.exe -c C:\Zabbix\agent\zabbix_agentd.win.conf -f
```

#### 2. Сборка Zabbix агента 2 на Windows

Обзор

На этой странице показано, как собрать Zabbix агент 2 из исходного кода в Windows 10 (64- или 32-разрядной версии).

Как 32-, так и 64-разрядные версии могут быть собраны на 64-разрядной платформе, но на 32-разрядной платформе может быть собрана только 32-разрядная версия.

Для сборки агента Zabbix 2 требуется:

- Инструменты сборки MinGW
- Язык программирования Go
- OpenSSL (для функций шифрования в Zabbix)
- PCRE2 (Perl Compatible Regular Expressions — Perl-совместимые регулярные выражения; для функций сопоставления шаблонов регулярных выражений в Zabbix)

Вы можете собрать агента Zabbix 2 одним из следующих способов:

- С использованием vcpkg — автоматизированный подход, упрощающий управление зависимостями с помощью менеджера пакетов C++.
- Ручная сборка — ручной подход, требующий установки всех зависимостей перед компиляцией агента.

**Attention:**
Перед началом сборки, пожалуйста, учтите следующее:<br><br>
- Для выполнения команд используйте командную строку, запущенную пользователем с правами на запись в защищённые папки. Однако при установке OpenSSL и PCRE2 используйте терминал MSYS2.
- Рекомендуется создать рабочий каталог `C:\Zabbix` для всех исходных файлов и папок сборки. Однако скомпилированные компоненты следует устанавливать в `C:\Zabbix\x64` (или `C:\Zabbix\x86` для 32-битных сборок).

Сборка Zabbix agent 2 с vcpkg

В этом разделе приведены инструкции по сборке Zabbix agent с помощью vcpkg — менеджера пакетов, который упрощает управление зависимостями и интеграцию с проектами на C++.

1. Скачайте и установите Build Tools for Visual Studio 2022. Во время установки обязательно выберите компонент *Desktop development with C++*, который включает менеджер пакетов vcpkg.

2. Скачайте и установите Go (доступен в виде MSI-установщика). Во время установки обязательно укажите `C:\Zabbix\Go` в качестве папки установки.

3. Скачайте дистрибутив MinGW, использующий библиотеку времени выполнения Microsoft Visual C; например:

- Для 64-разрядных сборок: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Для 32-разрядных сборок: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

**Attention:**
Версия MinGW должна соответствовать минимальным требованиям.

Затем распакуйте его в C:\Zabbix\mingw64 (или в C:\Zabbix\mingw32 для 32-разрядных сборок).

4. Инициализируйте vcpkg и установите зависимости, необходимые для сборки Zabbix agent 2 (учтите, что это может занять некоторое время):

```
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

#### For 64-bit builds:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
vcpkg install --triplet x64-mingw-static --x-install-root=x64

#### For 32-bit builds:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
vcpkg install --triplet x86-mingw-static --x-install-root=x86
```

5. Скачайте исходный архив Zabbix и распакуйте его в C:\Zabbix\zabbix-7.4.0.

6. Перейдите в каталог сборки Zabbix (C:\Zabbix\zabbix-7.4.0\build\mingw) и создайте следующий сценарий build.bat:

- Для 64-разрядных сборок:

```
:: 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
```

- Для 32-разрядных сборок:

```
:: 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. Скомпилируйте Zabbix agent 2, выполнив сценарий:

```
build.bat
```

После компиляции двоичный файл Zabbix agent 2 будет находиться в C:\Zabbix\zabbix-7.4.0\bin\win64 (для 64-разрядных сборок) или в C:\Zabbix\zabbix-7.4.0\bin\win32 (для 32-разрядных сборок). Файлы конфигурации Zabbix agent 2 находятся в C:\Zabbix\zabbix-7.4.0\src\go\conf.

Чтобы запустить агент, скопируйте zabbix_agent2.exe и его файлы конфигурации в отдельную папку (например, C:\Zabbix\agent2), а затем запустите агент:

```
mkdir C:\Zabbix\agent2

#### For 64-bit builds:
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\

#### For 32-bit builds:
copy C:\Zabbix\zabbix-7.4.0\bin\win32\zabbix_agent2.exe C:\Zabbix\agent2\

copy C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.win.conf C:\Zabbix\agent2\
xcopy /E /I C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.d C:\Zabbix\agent2\zabbix_agent2.d\

C:\Zabbix\agent2\zabbix_agent2.exe -c C:\Zabbix\agent2\zabbix_agent2.win.conf
```

При необходимости продолжите сборку Zabbix agent 2 с помощью подключаемых модулей.

Компиляция загружаемых плагинов Zabbix agent 2

1. Скачайте исходный код плагина Zabbix, соответствующий вашей версии Zabbix agent 2 (например, zabbix-agent2-plugin-postg
и распакуйте его в C:\Zabbix.

> **Attention:**
> Перед установкой плагина обязательно ознакомьтесь с файлом README. Он может содержать особые требования и инструкции по установке.

2. Перейдите в распакованный каталог плагина и скомпилируйте плагин:

```
cd C:\Zabbix\zabbix-agent2-plugin-ember-plus-7.4.0

####    64-          :
mingw32-make ARCH=AMD64

####    32-          :
mingw32-make ARCH=x86
```

После компиляции исполняемый файл плагина Zabbix agent 2 и его файл конфигурации будут находиться в том же каталоге плагина.

Исполняемый файл плагина можно разместить в любом месте, если он может быть загружен Zabbix agent 2. Укажите путь к двоичному файлу плагина в файле конфигурации плагина, например в postgresql.conf для плагина PostgreSQL:

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

Путь к файлу конфигурации плагина необходимо указать в параметре Include файла конфигурации Zabbix agent 2:

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

Перейдите к настройке для получения дополнительных сведений о настройке плагинов.

Сборка Zabbix агента 2 вручную

> **Attention:**
> Этот метод сборки агента Zabbix 2 подходит пользователям, которым требуется полный контроль над средой сборки или которые находятся в ограниченной среде, где использование vcpkg невозможно.

В этом разделе содержатся инструкции по ручной сборке агента Zabbix 2, включая установку необходимых инструментов сборки и зависимостей, а также компиляцию агента.

Настройка инструментов сборки

1. Скачайте и установите MSYS2 (доступен в виде MSI-установщика). Во время установки обязательно укажите `C:\Zabbix\msys64` в качестве папки установки.

2. Скачайте и установите Go (доступен в виде MSI-установщика; см. текущие поддерживаемые версии Go). Во время установки обязательно укажите `C:\Zabbix\Go` в качестве папки установки.

3. Скачайте дистрибутив MinGW, который использует библиотеку времени выполнения Microsoft Visual C; например:

- Для 64-разрядных сборок: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Для 32-разрядных сборок: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

**Attention:**
Версия MinGW должна соответствовать минимальным требованиям.

Затем распакуйте его в `C:\Zabbix\mingw64` (или `C:\Zabbix\mingw32` для 32-разрядных сборок).

Установка OpenSSL

**Note:**
Чтобы скомпилировать агент Zabbix без поддержки TLS, перейдите к разделу Установка PCRE2.

1. Откройте терминал MSYS2 с правами администратора и выполните следующие команды:

```
pacman -S perl-Locale-Maketext-Simple
pacman -S nasm
pacman -S make
pacman -S cmake
```

2. Скачайте архив исходного кода OpenSSL [en] и распакуйте его в папку `C:\Zabbix\openssl-3.5.0`.

3. Перейдите в распакованный каталог OpenSSL и создайте следующий скрипт `build.sh`:

- Для 64-битных сборок:

```
####!/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
```

- Для 32-битных сборок:

```
####!/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:**
Обязательно отмените права на запись для пользователей без прав администратора в каталоге `C:\Zabbix\x64\OpenSSL-Win64-350-static`. В противном случае агент будет загружать настройки SSL из пути, который могут изменять непривилегированные пользователи, что приводит к потенциальной уязвимости безопасности.

- Параметр `no-shared` делает статические библиотеки OpenSSL libcrypto.lib и libssl.lib самодостаточными, поэтому исполняемые файлы Zabbix включают OpenSSL без необходимости использования внешних DLL-библиотек. Это означает, что исполняемые файлы Zabbix можно копировать на другие компьютеры Windows без библиотек OpenSSL; однако при выпуске новой версии с исправлением ошибок OpenSSL потребуется перекомпилировать агент Zabbix.

- Без параметра `no-shared` Zabbix использует DLL-библиотеки OpenSSL во время выполнения. Это означает, что обновления OpenSSL могут не требовать перекомпиляции агента Zabbix; однако при копировании на другие компьютеры необходимо также скопировать DLL-библиотеки OpenSSL.

Подробнее о других параметрах конфигурации OpenSSL смотрите в документации OpenSSL [en].

4. Настройте и установите OpenSSL, выполнив скрипт (обратите внимание, что это может занять некоторое время):

```
cd /c/Zabbix/openssl-3.5.0
./build.sh
```

Установка PCRE2

1. Скачайте архив исходного кода PCRE2 [en] и распакуйте его в `C:\Zabbix\pcre2-10.45`.

2. Откройте терминал MSYS2 с правами администратора. Затем создайте каталог `build` в распакованном каталоге PCRE2 и перейдите в него:

```
mkdir /c/Zabbix/pcre2-10.45/build
cd /c/Zabbix/pcre2-10.45/build
```

3. Настройте PCRE2:

```
####    64-       :
export PATH="/c/Zabbix/mingw64/bin:$PATH"
cmake -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-O2 -g" -DCMAKE_INSTALL_PREFIX="/c/Zabbix/x64/PCRE2" ..

####    32-       :
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:**
При возникновении каких-либо ошибок рекомендуется удалить кэш CMake перед повторной попыткой сборки CMake. Кэш (`CMakeCachecache.txt`) находится в каталоге сборки распакованного каталога PCRE2.

4. Установите PCRE2:

```
make install
```

Сборка Zabbix agent 2

1. Скачайте исходный архив Zabbix и распакуйте его в `C:\Zabbix\zabbix-7.4.0`.

Если вам нужно создать исходный архив из исходного репозитория (например, чтобы применить собственные исправления или собрать из последней версии исходного кода), выполните следующие команды на машине с **Linux** и установленным Go (требуется для настройки Zabbix agent 2):

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure --enable-agent2 --enable-ipv6 --prefix=`pwd`
make dist
```

Это создаст исходный архив, который затем можно скопировать на машину с Windows.

2. Откройте командную строку с правами администратора. Затем перейдите в каталог сборки Zabbix и соберите Zabbix agent; убедитесь, что правильно указали каталоги, в которых установлены OpenSSL и PCRE2:

- Для 64-разрядной сборки:

```
cd C:\Zabbix\zabbix-7.4.0\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

####         TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2" OPENSSL="C:\Zabbix\x64\OpenSSL-Win64-350-static"

####          TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2"
```

- Для 32-разрядной сборки:

```
cd C:\Zabbix\zabbix-7.4.0\build\mingw
set PATH=C:\Zabbix\mingw32\bin;%PATH%

####         TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2" OPENSSL="C:\Zabbix\x86\OpenSSL-Win64-350-static"

####          TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2"
```

После компиляции двоичный файл Zabbix agent 2 будет находиться в C:\Zabbix\zabbix-7.4.0\bin\win64 (или в C:\Zabbix\zabbix-7.4.0\bin\win32 для 32-разрядной сборки). Файлы конфигурации Zabbix agent 2 находятся в C:\Zabbix\zabbix-7.4.0\src\go\conf.

Чтобы запустить агент, скопируйте двоичный файл zabbix_agent2.exe и его файлы конфигурации в отдельную папку (например, C:\Zabbix\agent2), а затем запустите агент:

```
mkdir C:\Zabbix\agent2
copy C:\Zabbix\zabbix-7.4.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\
copy C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.win.conf C:\Zabbix\agent2\
xcopy /E /I C:\Zabbix\zabbix-7.4.0\src\go\conf\zabbix_agent2.d C:\Zabbix\agent2\zabbix_agent2.d\

C:\Zabbix\agent2\zabbix_agent2.exe -c C:\Zabbix\agent2\zabbix_agent2.win.conf
```

При необходимости продолжите с сборкой загружаемых плагинов Zabbix agent 2.

#### 3. Сборка Zabbix агента на macOS

Обзор

Этот раздел демонстрирует, как можно собрать бинарные файлы Zabbix агента для macOS из исходного кода с/без поддержки TLS.

Предварительные требования

Вам потребуются средства разработки для командной строки (Xcode не требуется), Automake, pkg-config, а также PCRE (v8.x) или PCRE2 (v10.x). Если Вы хотите собрать бинарные файлы агента с поддержкой TLS, вам также понадобится OpenSSL или GnuTLS.

Чтобы установить Automake и pkg-config, вам потребуется менеджер пакетов Homebrew с https://brew.sh/. Чтобы его установить, откройте терминал и выполните следующую команду:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Затем установите Automake и pkg-config:

```
brew install automake
brew install pkg-config
```

Подготовка библиотек PCRE, OpenSSL и GnuTLS зависит от того, каким образом будет выполняться линковка к агенту.

Если вы намерены запускать бинарные файлы агента на машине macOS, на которой уже имеются данные библиотеки, то можно использовать предварительно скомпилированные библиотеки, поставляемые Homebrew. Обычно это macOS машины, которые используют Homebrew для сборки бинарных файлов Zabbix агента или для других целей.

Если бинарные файлы агента будут использоваться на macOS машинах, на которых отсутствуют разделяемые версии библиотек, то вам потребуется скомпилировать статические версии этих библиотек из исходного кода и выполнить линковку Zabbix агента к ним.

Сборка бинарных файлов агента с разделяемыми библиотеками

Установите PCRE2 (замените *pcre2* на *pcre* в команде ниже по необходимости):

```
brew install pcre2
```

При сборке с поддержкой TLS, установите OpenSSL и / или GnuTLS:

```
brew install openssl
brew install gnutls
```

Загрузите исходный код Zabbix:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

Сборка агента без поддержки TLS:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6
make
make install
```

Сборка агента с OpenSSL:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-openssl=/usr/local/opt/
make
make install
```

Сборка агента с 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
```

Сборка бинарных файлов агента со статическими библиотеками без TLS

Предположим, что статические библиотеки PCRE уже установлены в `$HOME/static-libs`. Мы будем использовать PCRE2 10.39.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
```

Загрузите и соберите PCRE с поддержкой свойств Юникод:

```
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
```

Загрузите исходный код Zabbix и соберите агента:

```
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
```

Сборка бинарных файлов агента со статическими библиотеками с OpenSSL

После успешной сборки OpenSSL рекомендуется выполнить `make test`. Иногда даже после успешной сборки тесты завершаются с ошибками. Если тесты завершаются с ошибками, проблемы необходимо рассмотреть и исправить перед тем как продолжить.

Предположим, что статические библиотеки PCRE и OpenSSL будут установлены в `$HOME/static-libs`. Мы будем использовать PCRE2 10.39 и OpenSSL 1.1.1a.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
OPENSSL_PREFIX="$HOME/static-libs/openssl-1.1.1a"
```

Давайте соберем статические библиотеки в `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Загрузите и соберите PCRE с поддержкой свойств Юникод:

```
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 ..
```

Загрузите и соберите 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 ..
```

Загрузите исходный код Zabbix и соберите агента:

```
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
```

Сборка бинарных файлов агента со статическими библиотеками с GnuTLS

GnuTLS зависит от низкоуровневой криптографической библиотеки Nettle и арифметической библиотеки GMP. Вместо использования полновесной библиотеки GMP, в этом руководстве будет использоваться mini-gmp, который включён в Nettle.

После успешной сборки GnuTLS и Nettle рекомендуется выполнить `make test`. Иногда даже после успешной сборки тесты завершаются с ошибками. Если тесты завершаются с ошибками, проблемы необходимо рассмотреть и исправить перед тем как продолжить.

Предположим, что статические библиотеки PCRE, Nettle и GnuTLS будут установлены в `$HOME/static-libs`. Мы будем использовать PCRE2 10.39, Nettle 3.4.1 и 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"
```

Давайте соберём статические библиотеки в `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Загрузите и соберите 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 ..
```

Загрузите и соберите 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 ..
```

Загрузите исходный код Zabbix и соберите агента:

```
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
```

#### 4. Установка из пакетов

Обзор

Официальные пакеты Zabbix 7.4 доступны на сайте Zabbix, где вы можете выбрать свою операционную систему и компонент Zabbix, чтобы сгенерировать необходимые инструкции по установке и команды. Обязательно ознакомьтесь с приведенными ниже примечаниями по установке пакетов и страницей Установка и настройка.

Пакеты доступны для следующих дистрибутивов Linux:

- Red Hat Enterprise Linux и его производные: 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:**
Некоторые дистрибутивы ОС (в частности, основанные на Debian) предоставляют собственные пакеты Zabbix. Эти пакеты **не** поддерживаются Zabbix и могут быть устаревшими или не содержать последних функций и исправлений ошибок. Рекомендуется использовать только официальные пакеты из официального репозитория Zabbix. Если вы ранее установили Zabbix из репозитория вашей операционной системы, см. шаги по обновлению пакетов Zabbix из репозиториев ОС.

Пакеты поддерживают базы данных MySQL/PostgreSQL и веб-сервер Apache/Nginx. Обратите внимание, что сервер Zabbix и прокси не могут использовать одну и ту же базу данных; используйте разные имена баз данных, если оба компонента установлены на одном узле сети.

При необходимости в официальном репозитории Zabbix доступны отдельные пакеты для агента Zabbix/агента 2, Zabbix get и Zabbix sender.

Zabbix также предоставляет предварительно скомпилированные двоичные файлы агента Zabbix для не-Linux операционных систем; см.:

- Установка агента Windows из MSI
- Установка агента macOS из PKG
- Устаревшие двоичные файлы (для более старых/менее распространенных систем, таких как HP-UX, NetBSD, Tru64 и более старых версий SLES)

Примечания по установке пакетов

Следующие примечания относятся ко всем системам:

- Если используется PostgreSQL, `DBHost=localhost` (или IP-адрес) в конфигурации Zabbix сервер/прокси заставляет PostgreSQL использовать сетевой сокет вместо локального UNIX-сокета; см. конфигурация SELinux для связанных инструкций по настройке.
- Если используется TimescaleDB, см. дополнительные сведения по настройке TimescaleDB.
- Если устанавливается Zabbix Java gateway (для мониторинга приложений JMX), см. дополнительные инструкции по настройке для систем на базе RHEL и систем на базе Debian.
- Для запуска Zabbix агент от имени root см. Запуск агента от имени root.

Следующие примечания относятся к RHEL и его производным:

- Если вы включили репозиторий EPEL для EL9, который также предоставляет пакеты Zabbix, его необходимо исключить из разрешения зависимостей перед установкой официальных пакетов Zabbix; см. Случайная установка пакетов EPEL Zabbix.

- Для установки пакетов Zabbix в средах Red Hat UBI см. Пакеты Zabbix для RHEL в средах Red Hat UBI.
- Для использования элементов данных ICMP ping пакеты для `fping` также доступны в официальном репозитории Zabbix.

Настройка SELinux

Zabbix использует межпроцессное взаимодействие на основе сокетов. На системах, где включен Security-Enhanced Linux (SELinux), может потребоваться добавить правила SELinux, чтобы разрешить Zabbix создавать и использовать UNIX domain sockets в каталоге SocketDir. Файлы сокетов используются сервером Zabbix (alerter, preprocessing, IPMI) и прокси Zabbix (IPMI), и они присутствуют, пока процесс выполняется.

При включенном SELinux в режиме enforcing выполните следующие команды, чтобы разрешить взаимодействие между веб-интерфейсом Zabbix и сервером:

Для RHEL 7 (и новее), AlmaLinux, CentOS Stream, Oracle Linux, Rocky Linux 8 (и новее):

```
setsebool -P httpd_can_connect_zabbix on
```

Если доступ к базе данных осуществляется по сети (включая `localhost` для PostgreSQL), также разрешите веб-интерфейсу Zabbix подключаться к базе данных:

```
setsebool -P httpd_can_network_connect_db on
```

Для RHEL до 7:

```
setsebool -P httpd_can_network_connect on
setsebool -P zabbix_can_network on
```

После применения настроек SELinux перезапустите Apache:

```
systemctl restart httpd
```

При необходимости вы можете установить предопределенный пакет `zabbix-selinux-policy` из The Zabbix Official Repository. Этот пакет предоставляется для всех поддерживаемых версий ОС, чтобы упростить развертывание Zabbix и предотвратить отключение SELinux пользователями из-за сложности настройки.

> **Attention:**
> Для максимальной безопасности рекомендуется задать собственные настройки SELinux.

Пакет `zabbix-selinux-policy` содержит базовую политику SELinux, которая позволяет Zabbix создавать и использовать сокеты, а также разрешает HTTPd подключение к PostgreSQL (используется веб-интерфейсом).

Исходный файл `zabbix_policy.te` содержит следующие правила:

```
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;
```

Пакеты Debuginfo

Пакеты Debuginfo содержат отладочные символы для бинарных файлов Zabbix. Они не требуются для обычной установки или работы, но полезны для расширенного устранения неполадок.

Обратите внимание, что пакеты Debuginfo доступны для следующих версий: Red Hat Enterprise Linux 7 и более ранних (включая производные), а также SUSE Linux Enterprise Server 15 и более ранних (включая производные).

Чтобы включить репозиторий zabbix-debuginfo:

- В RHEL 7 отредактируйте /etc/yum.repos.d/zabbix.repo и установите enabled=1 для секции zabbix-debuginfo:

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo - $basearch
baseurl=http://repo.zabbix.com/zabbix/7.4/stable/rhel/7/$basearch/debuginfo/
enabled=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
gpgcheck=1
```

- В SUSE отредактируйте /etc/zypp/repos.d/zabbix.repo и установите enabled=1 для секции zabbix-debuginfo:

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/7.4/stable/sles/15/$basearch/debuginfo/
gpgcheck=1
gpgkey=https://repo.zabbix.com/zabbix/7.4/stable/sles/15/$basearch/debuginfo/repodata/repomd.xml.key
enabled=0
update=1
```

После включения установите пакеты:

- В RHEL установите один пакет с отладочной информацией для всех компонентов Zabbix:

```
dnf install zabbix-debuginfo
```

- В SUSE установите пакеты debuginfo для отдельных компонентов:

```
zypper install zabbix-<component>-debuginfo
```

#### 1. Установка агента для Windows из MSI

Обзор

Zabbix агента для Windows можно установить из пакетов установщика Windows MSI (32-бит или 64-бит), доступных для загрузки.

Минимальные требования ОС для MSI установки:

- **для Zabbix агента:** Windows XP (64-бит) или Windows Сервер 2003 ;
- **для Zabbix агента 2:** Windows 10 (32-бит) или Windows Сервер 2016.

32-битный пакет нельзя установить на 64-битную Windows.

Пакеты включают:

- Поддержку TLS (настройка TLS необязательна)
- Утилиты Zabbix get и Zabbix sender (могут быть установлены вместе с Zabbix агентом/агентом 2 или отдельно)

> **Attention:**
> Пакеты Zabbix агента 2 не включают подгружаемые плагины (MongoDB, PostgreSQL, MSSQL), которые нужно загрузить и установить отдельно.

Установку можно выполнить с помощью Мастера настройки (Setup Wizard) или командной строки.

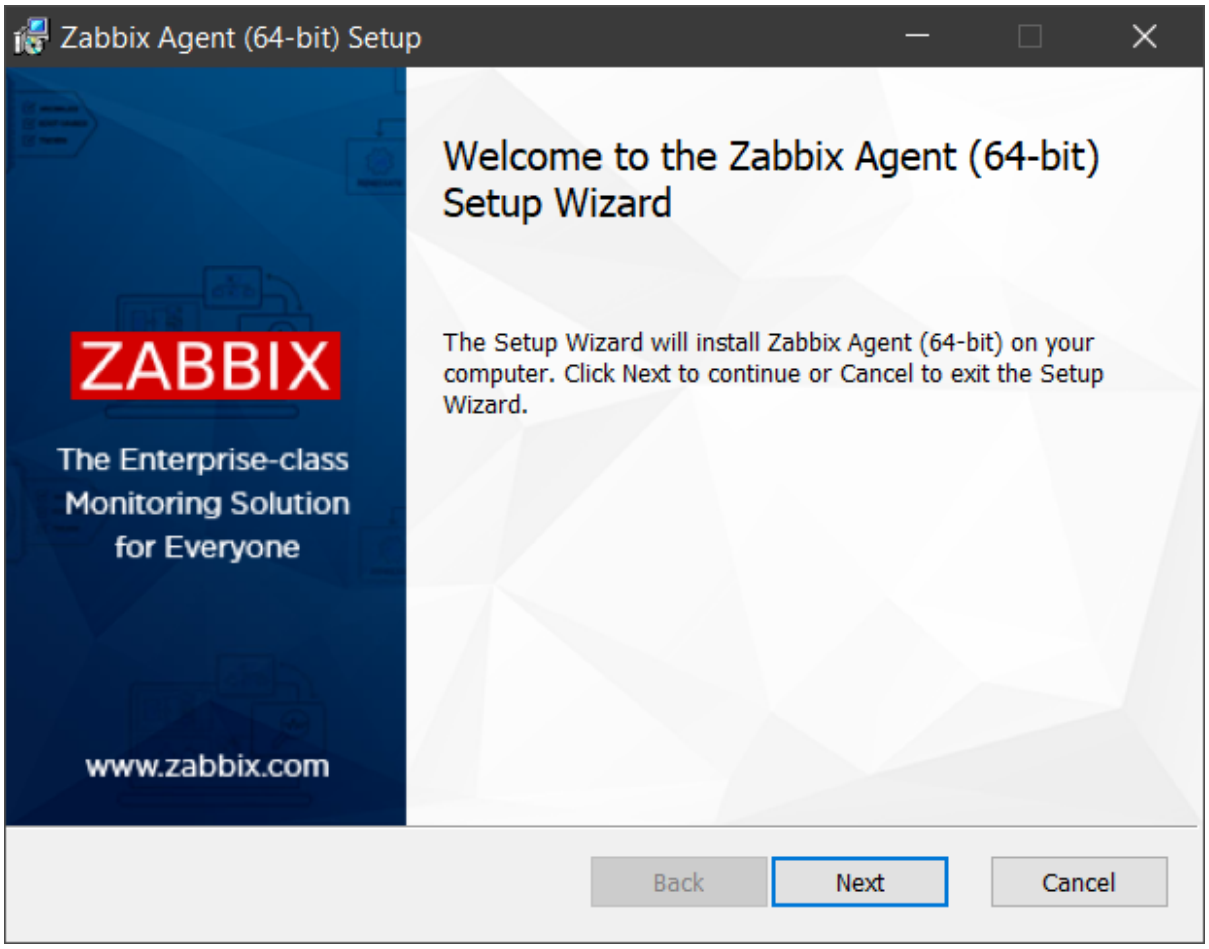
Хотя установка из пакетов установщика MSI полностью поддерживается, рекомендуется установить, как минимум, Microsoft .NET Framework 2 для правильной обработки ошибок.

> **Attention:**
> Рекомендуется использовать пути по умолчанию, предоставленные установщиком. Использование пользовательских путей без соответствующих разрешений может поставить под угрозу безопасность установки.

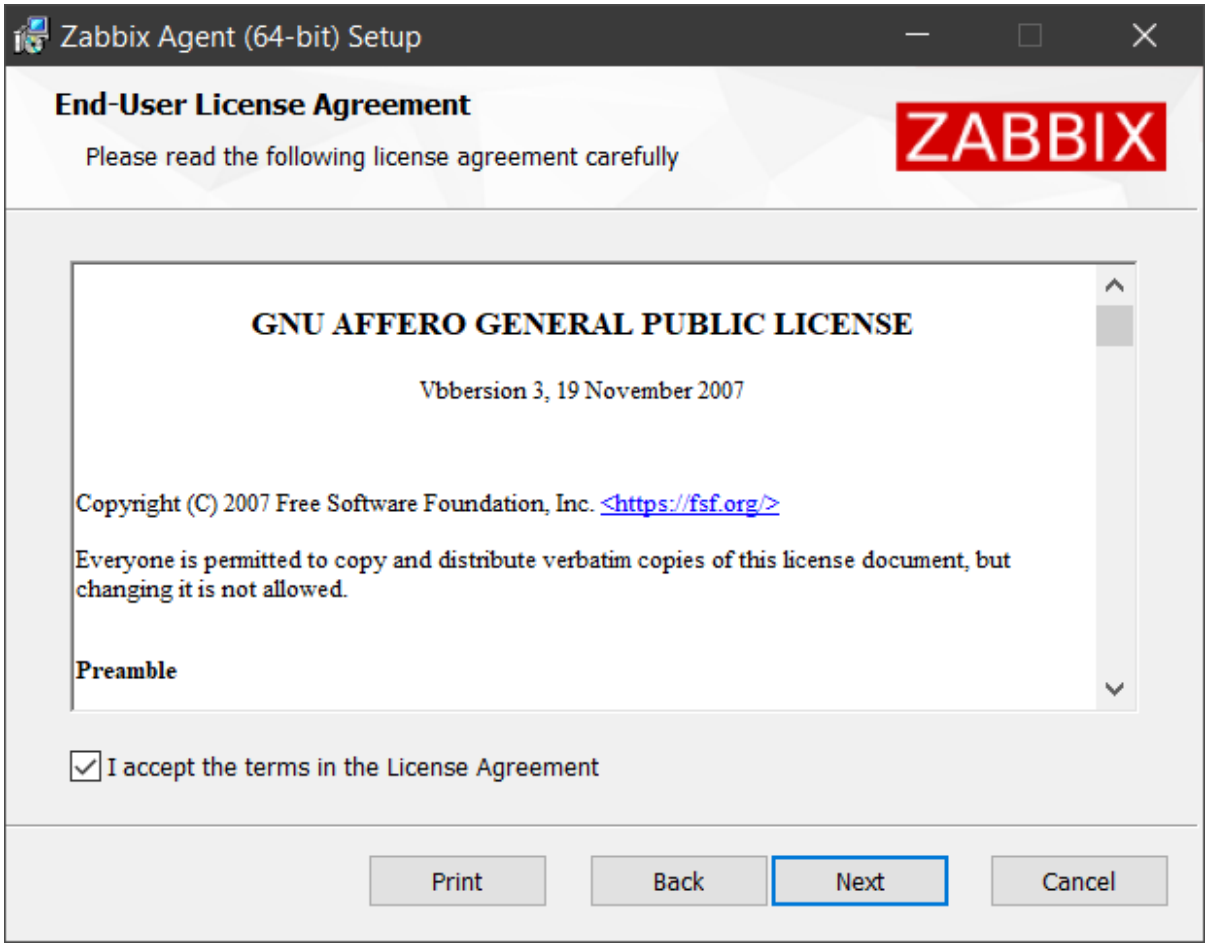
Установка с помощью мастера настройки

Следующие шаги установки применимы как для Zabbix агента, так и для Zabbix агента 2.

1. Чтобы начать установку, дважды щёлкните на загруженном MSI файле:

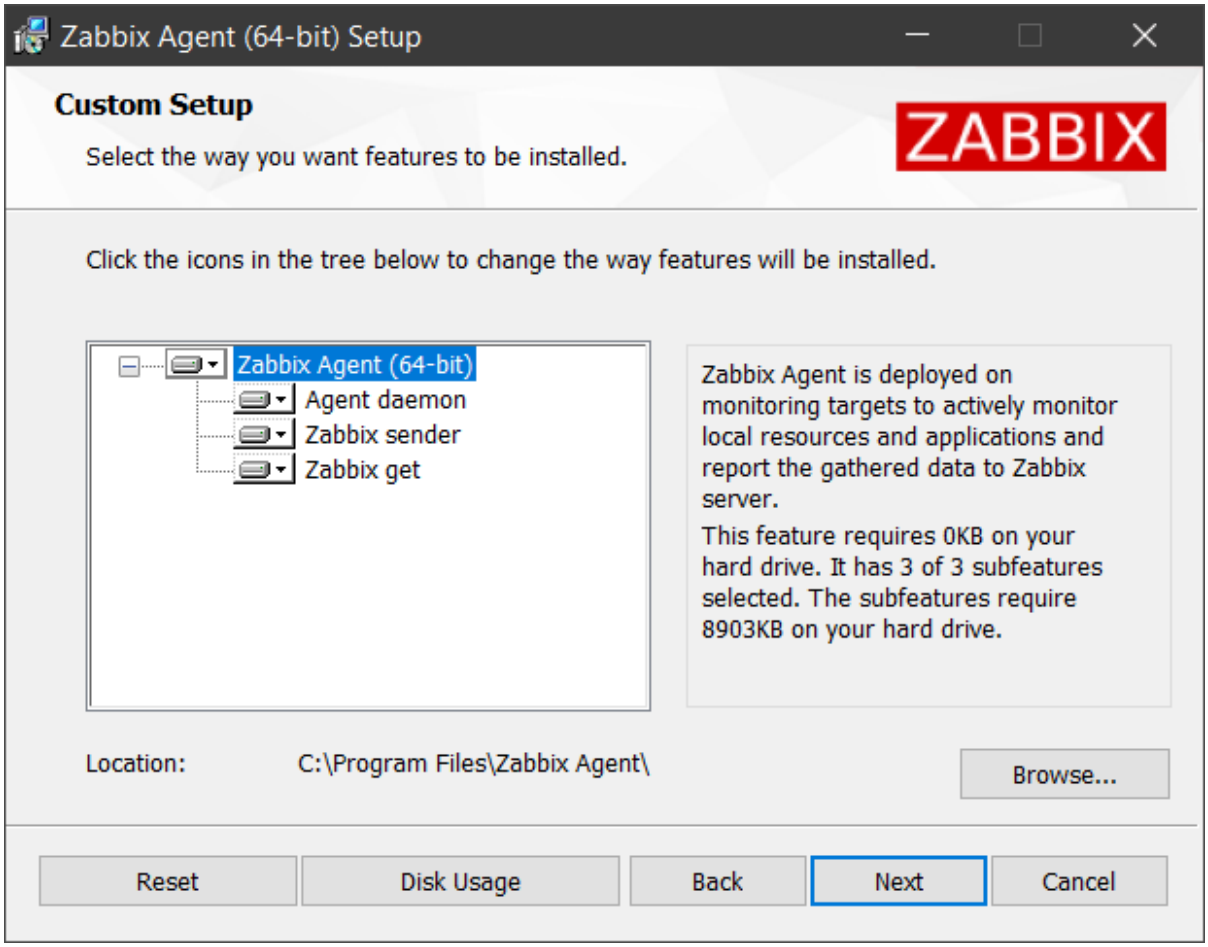



2. Примите лицензионное соглашение:




3. Выберите компоненты Zabbix для установки — демон Zabbix агента, Zabbix sender, Zabbix get:

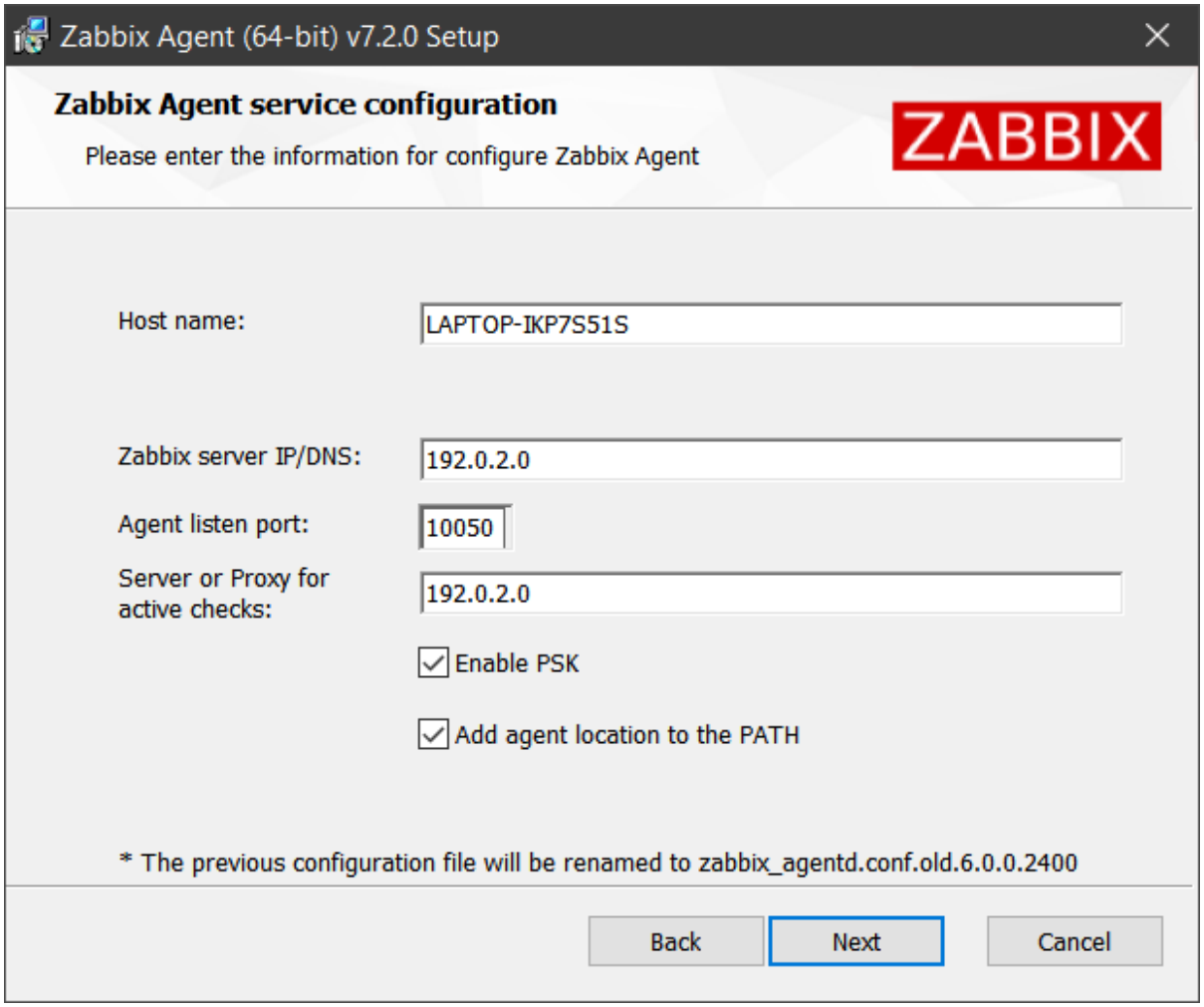
**Attention:**
Рекомендуется использовать пути по умолчанию, предоставленные установщиком. Использование пользовательских путей без соответствующих разрешений может поставить под угрозу безопасность установки.




4. Настройте следующие параметры. Их значения будут заданы в файле конфигурации Zabbix агента:

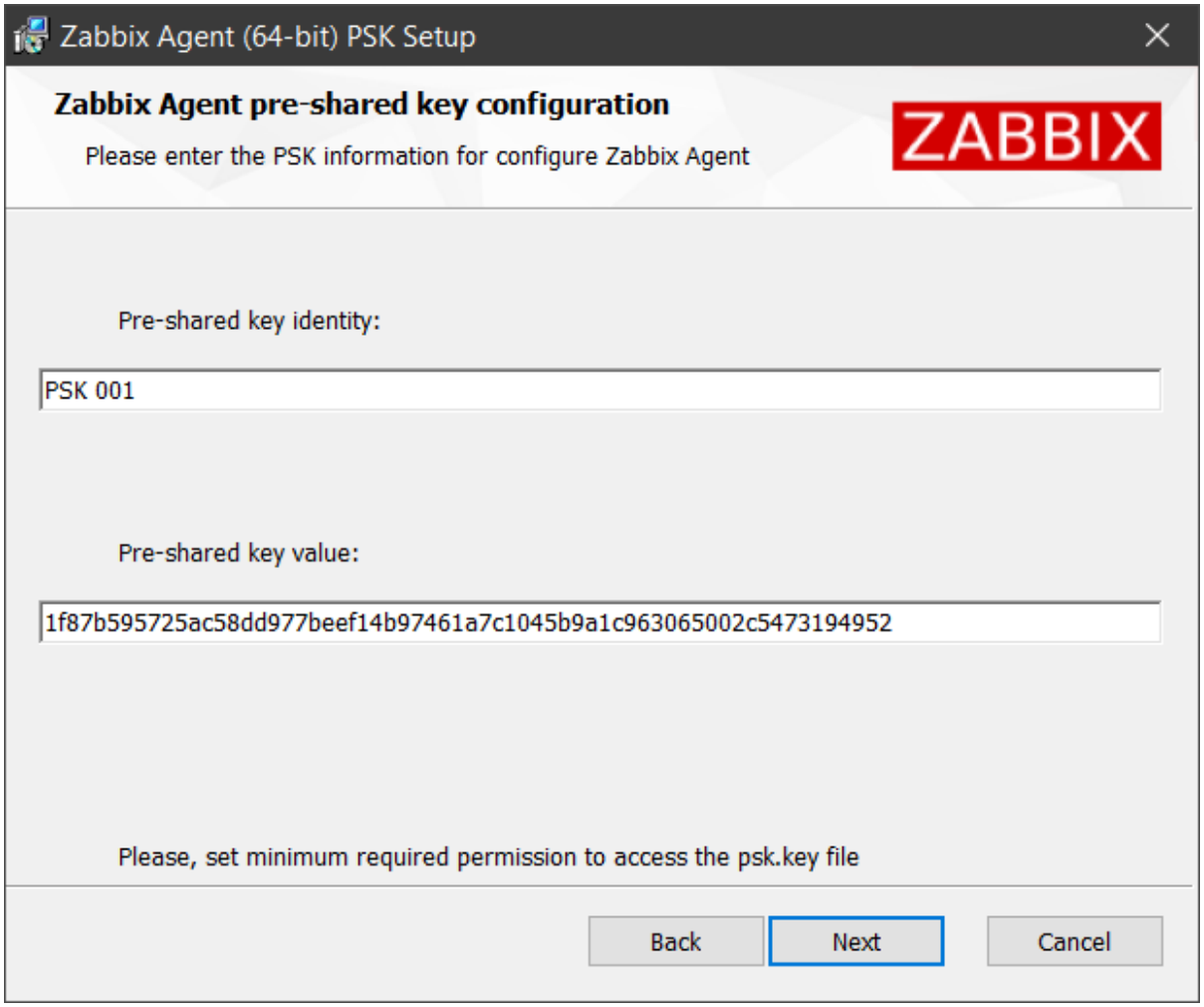
| Параметр | Описание |
|---|---|
| *Host name* | Имя узла сети машины, на которую устанавливается Zabbix агент. Задаёт параметр Hostname. |
| *Zabbix server IP/DNS* | Список (через запятую) IP-адресов, опционально в нотации CIDR, или DNS-имён Zabbix серверов или Zabbix прокси. Этот параметр **обязателен**. Устанавливает параметр Server. |
| *Agent listen port* | Порт, который будет слушать агент на предмет подключений со стороны сервера. Устанавливает параметр ListenPort. |
| *Server or Proxy for active checks* | Адрес сервера/прокси Zabbix либо конфигурация кластера для получения активных проверок. Адрес сервера/прокси — это IP-адрес или DNS-имя и необязательный порт через двоеточие. Устанавливает параметр ServerActive. |
| *Enable PSK* | Отметьте флажок, чтобы включить поддержку TLS с использованием общего ключа (pre-shared key). Устанавливает параметры TLSConnect и TLSAccept в значение `psk`. |
| *Add agent location to the PATH* | Отметьте флажок, чтобы добавить местоположение агента в системную переменную PATH. |

**Note:**
Если обнаружен существующий агент Zabbix, будут отображены значения параметров из его настроек. Кроме того, существующий файл настроек во время установки будет переименован, и будет создан новый файл настроек.




5. Настройте параметры общего ключа, если вы отметили флажок *Enable PSK* на предыдущем шаге. Эти параметры также будут установлены в файле настроек Zabbix агента:

| Параметр | Описание |
|---|---|
| *Pre-shared key identity* | Строка идентификатора общего ключа (PSK). Устанавливает параметр TLSPSKIdentity. |
| *Pre-shared key value* | Значение строки общего ключа. Создаёт файл psk.key, содержащий ключ, и устанавливает параметр TLSPSKFile в местоположение ключа (по умолчанию: `C:\Program Files\Zabbix Agent\psk.key`). **Рекомендуется** ограничить доступ к файлу общего ключа, изменив настройки безопасности файла таким образом, чтобы прочитать его мог только Zabbix агент (или пользователь, запускающий агента). |

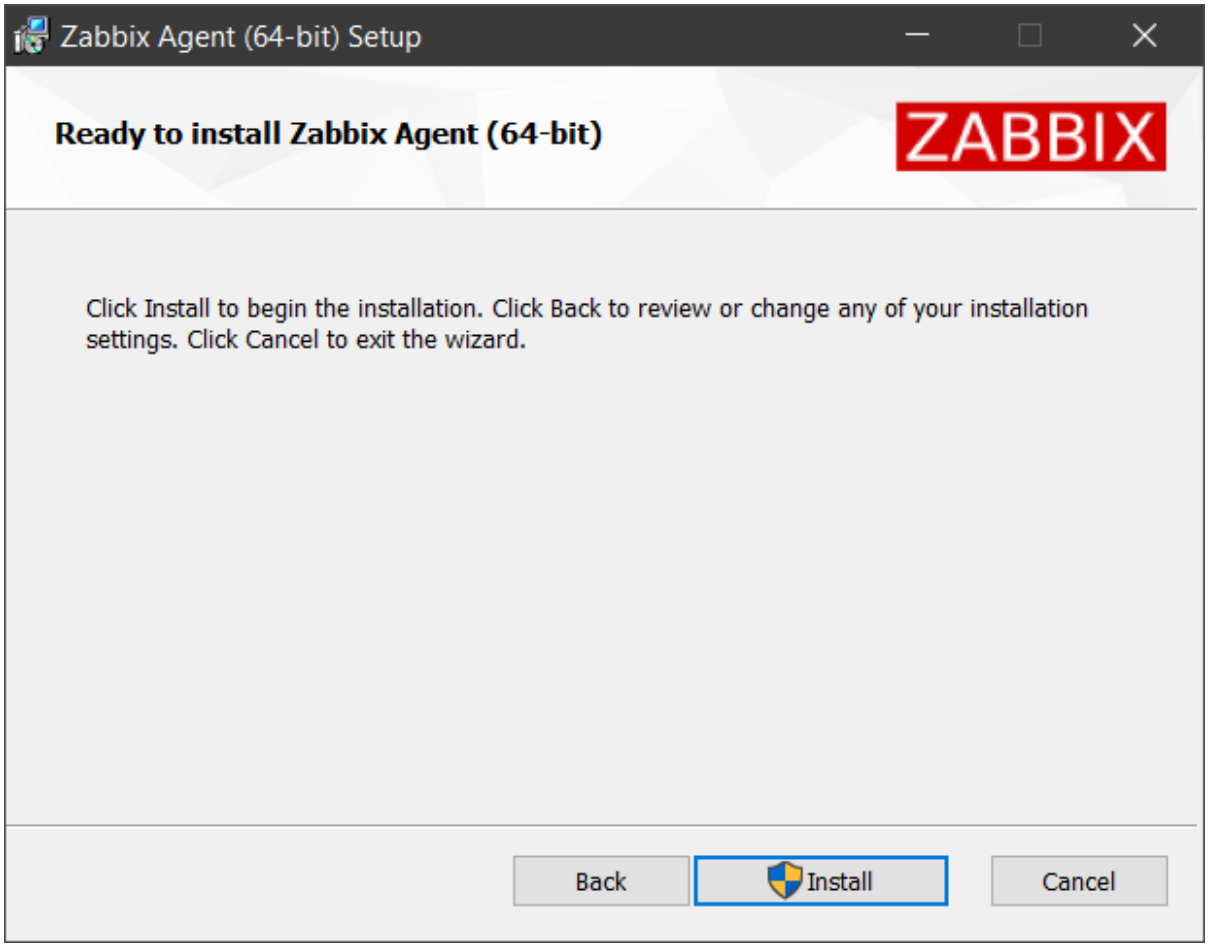



6. Нажмите *Install*, чтобы начать установку.

Все выбранные компоненты Zabbix и файлом настроек Zabbix агента будут установлены в указанную папку (по умочанию: `C:\Program Files\Zabbix Agent`). Тоо же самое относится к Zabbix агенты 2, за исключением того, что дополнительные конфигурационные файлы для его встроенных плагинов будут установлены в подпапку `zabbix_agent2.d\plugins.d`.

Кроме того, zabbix_agentd.exe (или zabbix_agent2.exe) будет настроен в качестве службы Windows с отложенным автоматическим запуском (или автоматическим запуском в версиях Windows до Windows Server 2008/Vista).

Если во время установки запущена другая версия Zabbix агента, вам будет предложено выбрать: либо закрыть приложение и попытаться перезапустить его, либо оставить его открытым (в этом случае потребуется перезагрузка).




Установка из командной строки

Агент Zabbix можно установить из командной строки, запустив MSI-установщик с помощью msiexec. Например:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.0-windows-amd64-openssl.msi" /qn+ SERVER=192.0.2
```

Этот способ позволяет выполнять установку без участия пользователя и использовать пользовательские настройки с помощью параметров.

Поддерживаемые параметры

Пакеты MSI-установщика Zabbix agent поддерживают следующие параметры как для Zabbix agent, так и для Zabbix agent 2.

**Note:**
Параметры Zabbix agent/agent2 задаются в файле конфигурации во время установки. Щелкните имя параметра, чтобы просмотреть его подробное описание и примеры настройки на странице Zabbix agent (Windows). Для Zabbix agent 2 см. страницу Zabbix agent 2 (Windows).

| Parameter | Description |
|---|---|
| ADDDEFAULT | Список компонентов, разделенных запятыми, которые нужно установить с конфигурацией по умолчанию. Дополнительные сведения см. в разделе свойство ADDDEFAULT.<br>Возможные значения: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL`<br>Пример: `ADDDEFAULT=AgentProgram,GetProgram` |
| ADDLOCAL | Список компонентов, разделенных запятыми, которые нужно установить локально. Дополнительные сведения см. в разделе свойство ADDLOCAL.<br>Возможные значения: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL`<br>Пример: `ADDLOCAL=AgentProgram,SenderProgram` |

| Parameter | Description |
|---|---|
| ALLOWDENYKEY | Список параметров AllowKey или DenyKey, разделенных точкой с запятой, для ограничения проверок Zabbix agent. При необходимости используйте обратную косую черту для экранирования разделителя (\;). Задает параметры AllowKey и DenyKey в файле конфигурации агента.<br>Пример: `ALLOWDENYKEY="AllowKey=system.run[more C:\Windows\System32\drivers\etc\hosts\\; echo 'File read complete'];DenyKey=system.run[*]"` |
| CONF | Полный путь к файлу конфигурации шаблона для Zabbix agent. Во время установки этот файл станет файлом конфигурации агента. Файл должен содержать как минимум параметры Server и LogFile.<br>Пример: `CONF="C:\full\path\to\example.conf"` |
| DONOTSTART | Используйте `DONOTSTART=1`, чтобы MSI-установщик не запускал службу Zabbix agent.<br>Поддерживается в пакетах MSI-установщика начиная с Zabbix 7.4.6. |
| ENABLEPATH | Используйте `ENABLEPATH=1`, чтобы добавить расположение Zabbix agent в системную переменную PATH. |
| ENABLEPERSISTENTBUFFER | Только для Zabbix agent 2. Включает использование локального постоянного хранилища для активных элементов данных. |
| HOSTINTERFACE | Необязательный параметр, определяющий интерфейс узла сети. |
| HOSTMETADATA | Необязательный параметр, определяющий метаданные узла сети. |
| HOSTMETADATAITEM | Необязательный параметр, определяющий элемент данных, используемый для получения метаданных узла сети. |
| HOSTNAME | Необязательный параметр, определяющий имя узла сети. |
| HOSTNAMEITEM | Необязательный параметр, определяющий элемент данных, используемый для получения имени узла сети.<br>Поддерживается в пакетах MSI-установщика начиная с Zabbix 7.4.6. |
| INCLUDE | Список отдельных файлов или всех файлов в каталоге, разделенных точкой с запятой, которые нужно включить в файл конфигурации Zabbix agent. |
| INSTALLFOLDER | Полный путь к папке, в которую будут установлены компоненты Zabbix и файл конфигурации Zabbix agent. Для Zabbix agent 2 дополнительные файлы конфигурации для встроенных плагинов будут установлены в подпапку `zabbix_agent2.d\plugins.d`.<br>Пример: `INSTALLFOLDER="C:\Program Files\Zabbix Agent"` |
| LISTENIP | Список IP-адресов, разделенных запятыми, на которых должен прослушивать агент. |
| LISTENPORT | Агент будет прослушивать этот порт для подключений от сервера. |
| LOGFILE | Имя файла журнала Zabbix agent. |
| LOGTYPE | Тип вывода журнала. |
| NONMSICONFNAME | Полный путь к пользовательскому файлу конфигурации для Zabbix agent. Во время установки любые допустимые параметры конфигурации агента, присутствующие в этом файле (ограниченные перечисленными в этой таблице), будут записаны в новый файл конфигурации агента. Файл должен содержать как минимум параметр Server.<br>Пример: `NONMSICONFNAME="C:\full\path\to\example.conf"` |
| PERSISTENTBUFFERFILE | Только для Zabbix agent 2. Файл, в котором Zabbix agent 2 должен хранить базу данных SQLite. |
| PERSISTENTBUFFERPERIOD | Только для Zabbix agent 2. Период времени, в течение которого данные должны храниться при отсутствии соединения с сервером или прокси. |
| SERVER | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен серверов Zabbix или прокси Zabbix. Этот параметр **обязателен**, кроме случая, когда STARTAGENTS установлен в 0. |
| SERVERACTIVE | Адрес сервера Zabbix или прокси либо конфигурация кластера, откуда получать активные проверки. |
| SKIP | Используйте `SKIP=fw`, чтобы MSI-установщик не добавлял правило исключения в брандмауэр Windows для Zabbix agent. |
| SOURCEIP | Исходный IP-адрес для исходящих подключений к серверу Zabbix или прокси Zabbix, а также для установления соединений при выполнении некоторых элементов данных (web.page.get, net.tcp.port и т. д.).<br>Поддерживается в пакетах MSI-установщика начиная с Zabbix 7.4.6. |
| STARTAGENTS | Количество предварительно созданных экземпляров zabbix_agentd, обрабатывающих пассивные проверки.<br>Если установлено значение 0, пассивные проверки отключаются, и агент не будет прослушивать ни один TCP-порт.<br>Поддерживается в пакетах MSI-установщика начиная с Zabbix 7.4.6. |

| Parameter | Description |
|---|---|
| STARTUPTYPE | Тип запуска службы Zabbix agent. Возможные значения:<br>**automatic** - запускать службу автоматически при старте Windows;<br>**delayed** - *(по умолчанию)* отложить запуск службы после завершения запуска служб, запускаемых автоматически (доступно в Windows Vista/Server 2008 и более поздних версиях);<br>**manual** - запускать службу вручную (пользователем или приложением);<br>**disabled** - отключить службу, чтобы ее не мог запустить пользователь или приложение.<br>Пример: `STARTUPTYPE=disabled` |
| STATUSPORT | Только для Zabbix agent 2. Если задано, агент будет прослушивать этот порт для HTTP-запросов состояния (http://localhost:<port>/status). |
| TIMEOUT | Указывает, сколько времени ожидать (в секундах) установления соединения и обмена данными с прокси или сервером Zabbix. |
| TLSACCEPT | Входящие подключения, которые следует принимать (используется для пассивных проверок). Если установлено значение `psk`, то для TLSCONNECT также будет установлено значение `psk` (если не указано иное). |
| TLSCAFILE | Полный путь к файлу, содержащему сертификаты корневого центра сертификации для проверки сертификата узла. |
| TLSCERTFILE | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов. |
| TLSCONNECT | Как агент должен подключаться к серверу Zabbix или прокси (используется для активных проверок). Если установлено значение `psk`, то для TLSACCEPT также будет установлено значение `psk` (если не указано иное). |
| TLSCRLFILE | Полный путь к файлу, содержащему отозванные сертификаты. |
| TLSKEYFILE | Полный путь к файлу, содержащему закрытый ключ Zabbix agent. |
| TLSPSKFILE | Полный путь к файлу, содержащему предварительно общий ключ Zabbix agent. Если заданы и TLSPSKFILE, и TLSPSKVALUE, значение TLSPSKVALUE будет записано в файл, указанный в TLSPSKFILE. **Рекомендуется** ограничить доступ к файлу предварительно общего ключа, настроив параметры безопасности файла так, чтобы его мог читать только Zabbix agent (или пользователь, от имени которого запущен агент). |
| TLSPSKIDENTITY | Строка идентификатора предварительно общего ключа. |
| TLSPSKVALUE | Строковое значение предварительно общего ключа. Если заданы и TLSPSKFILE, и TLSPSKVALUE, значение TLSPSKVALUE будет записано в файл, указанный в TLSPSKFILE.<br>Пример:<br>`TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952` |
| TLSSERVERCERTISSUER | Допустимый издатель сертификата сервера (прокси). |
| TLSSERVERCERTSUBJECT | Допустимый субъект сертификата сервера (прокси). |
| UNSAFEUSERPARAMETERS | Разрешить передачу всех символов в аргументах пользовательских параметров. Поддерживается в пакетах MSI-установщика начиная с Zabbix 7.4.6. |

Примеры

Следующий пример устанавливает агент Zabbix с пользовательской конфигурацией. Он также включает поддержку TLS с использованием предварительно общих ключей.

```
mkdir "C:\Program Files\Zabbix Agent" 2>nul
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.0-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[*]"
```

Следующий пример устанавливает более новую версию агента Zabbix и использует **файл конфигурации шаблона** (`CONF="C:\agent-template.conf"`). Во время установки этот файл станет файлом конфигурации агента. Чтобы унаследовать параметры из старого файла конфигурации, используйте заполнители параметров (например, `[AllowDenyKey]`).

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME
```

```
####     agent-template.conf:
LogFile=[LogFile]
[AllowDenyKey]
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
[Include]
[TLSConnect]
[TLSAccept]
[TLSPSKIdentity]
[TLSPSKFile]
```

В качестве альтернативы можно использовать **пользовательский файл конфигурации** (NONMSICONFNAME="C:\agent-custom.co
Во время установки любые допустимые параметры конфигурации агента, присутствующие в этом файле (ограниченные перечисленными в таблице выше), будут записаны в newly created файл конфигурации агента. Чтобы сохранить существующую конфигурацию агента, определите параметры, которые нужно сохранить.

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-7.4.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

####     agent-custom.conf:
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
```

Загружаемые плагины Zabbix agent 2

Загружаемые плагины Zabbix agent 2 можно установить в Windows с помощью 64-разрядных MSI-пакетов установщика, доступных для скачивания.

Минимальные требования к ОС для установки MSI: Windows 10 (64-bit) или Windows Server 2016.

Как и в случае с Zabbix agent/agent2, загружаемые плагины можно установить с помощью мастера установки или из командной строки.

> **Attention:**
> Перед установкой плагина, пожалуйста, ознакомьтесь с его файлом README. Он может содержать особые требования и инструкции по установке.

Установка с помощью мастера настройки

1. Дважды щёлкните загруженный файл MSI, чтобы начать установку.

2. Примите лицензионное соглашение конечного пользователя.

3. Выберите для установки подгружаемый плагин Zabbix агента 2.

> **Attention:**
> Рекомендуется использовать пути по умолчанию, предоставленные установщиком. Использование пользовательских путей без соответствующих разрешений может поставить под угрозу безопасность установки

4. Нажмите *Install*, чтобы начать установку.

Все выбранные подгружаемые модули Zabbix агента 2 будут установлены в указанное место (по умолчанию: `C:\Program Files\Zabbix Agent 2`), а из конфигурационные файлы будут установлены в подпапку `zabbix_agent2.d`.

5. Нажмите кнопку *Finish*, чтобы выйти из мастера установки.

Установка из командной строки

Подключаемые плагины Zabbix agent 2 можно установить из командной строки, запустив MSI-установщик с помощью msiexec. Например:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent2_plugins-7.4.0-windows-amd64.msi" /qn+
```

Пакеты MSI-установщика подключаемых плагинов Zabbix agent 2 поддерживают следующие параметры.

| Parameter | Description |
|---|---|
| ADDDEFAULT | Список компонентов, разделенных запятыми, для установки в конфигурации по умолчанию. Дополнительные сведения см. в свойстве ADDDEFAULT.<br>Возможные значения: ALL, EmberplusPlugin, MongodbPlugin, MssqlPlugin, NvidiagpuPlugin, PostgresqlPlugin<br>Пример: ADDDEFAULT=MongodbPlugin,PostgresqlPlugin |
| ADDLOCAL | Список компонентов, разделенных запятыми, для локальной установки. Дополнительные сведения см. в свойстве ADDLOCAL.<br>Возможные значения: ALL, EmberplusPlugin, MongodbPlugin, MssqlPlugin, NvidiagpuPlugin, PostgresqlPlugin<br>Пример: ADDLOCAL=MongodbPlugin,MssqlPlugin |
| INSTALLFOLDER | Полный путь к папке, в которую будут установлены компоненты Zabbix, а их файлы конфигурации будут установлены в подпапку zabbix_agent2.d.<br>Пример: INSTALLFOLDER="C:\Program Files\Zabbix Agent 2" |

**2. Установка агента для macOS из PKG**

Обзор

Zabbix агент можно установить на macOS с помощью пакетов установщика PKG, которые доступны для загрузки.

Пакеты агентов Zabbix доступны с шифрованием и без него.

Установка агента

Агент можно установить с помощью графического пользовательского интерфейса или из командной строки, например:

```
sudo installer -pkg zabbix_agent-7.4.0-macos-arm64-openssl.pkg -target /
```

Убедитесь, что в команде используется правильная версия пакета Zabbix. Она должна совпадать с именем загруженного пакета.

Запуск агента

Агент автоматически запустится после установки или перезагрузки.

При необходимости вы можете отредактировать файл конфигурации в /usr/local/etc/zabbix/zabbix_agentd.conf.

Чтобы запустить агента вручную, вы можете выполнить:

```
sudo launchctl start com.zabbix.zabbix_agentd
```

Для остановки агента вручную:

```
sudo launchctl stop com.zabbix.zabbix_agentd
```

В процессе обновления существующий файл конфигурации не будет перезаписан. Вместо этого будет создан новый файл zabbix_agentd.conf.NEW, который можно просмотреть и затем обновить существующий файл конфигурации, если необходимо. Не забудьте перезапустить агента после ручных изменений в файле конфигурации.

Устранение неполадок и удаление агента

В этом разделе перечислены некоторые полезные команды, которые можно использовать для поиска неполадок и удаления установленного Zabbix агента.

Просмотр, запущен ли Zabbix агент:

```
ps aux | grep zabbix_agentd
```

Просмотр, установлен ли Zabbix агент из пакетов:

```
pkgutil --pkgs | grep zabbix
com.zabbix.pkg.ZabbixAgent
```

Просмотр списка файлов, которые установлены пакетом установщика (обратите внимание, начальный символ «/» не отображается в выводе ниже):

```
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
```

Останов Zabbix агента, если он запущен при помощи `launchctl`:

```
sudo launchctl unload /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
```

Удаление файлов (включая конфигурацию и журналы), которые были установлены пакетом установщика:

```
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
```

Забыть, что Zabbix агент был установлен:

```
sudo pkgutil --forget com.zabbix.pkg.ZabbixAgent
```

#### 3. Нестабильные релизы

Обзор

Ниже приведены инструкции для подключения репозиториев нестабильных релизов Zabbix (по умолчанию отключённых), используемых для минорных версий релиз-кандидатов Zabbix.

Первым делом, установите либо обновите до последней версии пакет zabbix-release. Чтобы подключить пакеты релиз-кандидатов на вашей системе, выполните следующие действия:

Red Hat Enterprise Linux

Откройте файл `/etc/yum.repos.d/zabbix.repo` и установите `enabled=1` для репозитория `zabbix-unstable`.

```
[zabbix-unstable]
name=              Zabbix (unstable) - $basearch
baseurl=https://repo.zabbix.com/zabbix/7.4/unstable/rhel/8/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
```

Debian/Ubuntu

Откройте файл `/etc/apt/sources.list.d/zabbix.list` и раскомментируйте строку "Zabbix unstable repository".

```
#### Zabbix unstable repository
deb https://repo.zabbix.com/zabbix/7.4/unstable/debian bullseye main
deb-src https://repo.zabbix.com/zabbix/7.4/unstable/debian bullseye main
```

SUSE

Откройте файл `/etc/zypp/repos.d/zabbix.repo` и установите `enable=1` для репозитория `zabbix-unstable`.

```
[zabbix-unstable]
name=Zabbix Official Repository
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/7.4/unstable/sles/15/x86_64/
gpgcheck=1
gpgkey=https://repo.zabbix.com/zabbix/7.4/unstable/sles/15/x86_64/repodata/repomd.xml.key
enabled=1
update=1
```

#### 5. Установка из контейнеров

**Обзор** На этой странице описано, как развернуть Zabbix с помощью Docker или Docker Compose.

Docker Compose — самый быстрый способ развернуть Zabbix. Он читает файл конфигурации и автоматически запускает все контейнеры, из которых состоит полная установка Zabbix, в правильном порядке.

Docker (ручное развертывание) позволяет добиться того же результата, развертывая каждый компонент шаг за шагом.

**Предварительные требования** Прежде чем начать, убедитесь, что в вашей системе установлен Docker (версия 1.12.0 или более поздняя). Если это не так, следуйте руководству по установке Docker [en]. Для развёртывания с помощью Docker Compose также требуется Docker Compose (версия 2.24.0 или более поздняя).

Для работы некоторых компонентов Zabbix на хосте, где запущен Docker, необходимо открыть определённые порты (например, 10051/TCP для сервера Zabbix, 162/UDP для SNMP-трапов). Полный список портов, используемых компонентами Zabbix, смотрите в разделе Требования. Для сервера и агента Zabbix порт по умолчанию можно изменить, установив в соответствующем контейнере переменную окружения `ZBX_LISTENPORT`.

**Доступные образы Docker** Zabbix предоставляет образ Docker для каждого компонента Zabbix, все они опубликованы на Docker Hub. Каждый образ используется для создания контейнера, выполняющего этот компонент.

| Компонент Zabbix | Образ Docker |
|---|---|
| Агент | zabbix/zabbix-agent |
| Агент 2 | zabbix/zabbix-agent2 |
| Сервер (MySQL) | zabbix/zabbix-server-mysql |
| Сервер (PostgreSQL) | zabbix/zabbix-server-pgsql |
| Веб-интерфейс (Apache + MySQL) | zabbix/zabbix-web-apache-mysql |
| Веб-интерфейс (Apache + PostgreSQL) | zabbix/zabbix-web-apache-pgsql |
| Веб-интерфейс (Nginx + MySQL) | zabbix/zabbix-web-nginx-mysql |
| Веб-интерфейс (Nginx + PostgreSQL) | zabbix/zabbix-web-nginx-pgsql |
| Прокси (SQLite3) | zabbix/zabbix-proxy-sqlite3 |
| Прокси (MySQL) | zabbix/zabbix-proxy-mysql |
| Java gateway | zabbix/zabbix-java-gateway |
| Веб-сервис | zabbix/zabbix-web-service |
| SNMP трапы | zabbix/zabbix-snmptraps |

**Note:**
Для использования SNMP трапов контейнер SNMP трапов должен использовать общий том с сервером Zabbix или контейнером прокси (см. как использовать этот образ [en] на Docker Hub и пример ниже).

Все образы компонентов Zabbix на Docker Hub основаны на последних мажорных версиях поддерживаемых операционных систем. Эти образы автоматически пересобираются при обновлении базовых образов ОС.

Теги образов

Каждый образ поддерживает теги для выбора базовой операционной системы и версии Zabbix:

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

Поддерживаемые значения `<os>`:

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

Поддерживаемые значения `<version>`:

- `latest` - Последняя стабильная версия Zabbix в Alpine Linux
- `<os>-latest` - Последняя стабильная версия Zabbix для выбранной ОС
- `<os>-trunk` - Последняя сборка разработки (nightly) для выбранной ОС
- `<os>-X.X-latest` - Последний минорный выпуск Zabbix для указанной мажорной версии Zabbix на выбранной ОС
- `<os>-X.X.*` - Конкретный минорный выпуск Zabbix на выбранной ОС

Примеры:

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

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

###                 (nightly)      Zabbix (MySQL)  Ubuntu:
docker pull zabbix/zabbix-server-mysql:ubuntu-trunk

###                 7.4      Zabbix (MySQL)  Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-7.4-latest

###     7.4.1      Zabbix (MySQL)  Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-7.4.1
```

**Docker Compose** Docker Compose — это самый быстрый способ развернуть Zabbix. Он читает файл конфигурации (Compose file) и выполняет всю настройку: загружает образы Docker, создает внутреннюю сеть между контейнерами, настраивает хранилище, инициализирует базу данных и запускает все в правильном порядке.

В репозитории официальных Dockerfiles Zabbix представлены готовые к использованию файлы Docker Compose и система конфигурации на основе `.env`, поддерживающая разные операционные системы, серверные части баз данных и конфигурацию компонентов Zabbix.

1. Клонируйте репозиторий, перейдите в него и переключитесь на версию 7.4:

```
git clone https://github.com/zabbix/zabbix-docker.git
cd zabbix-docker
git checkout 7.4
```

2. (Необязательно) Настройте развертывание с помощью переменных среды. Этот шаг можно пропустить для конфигурации по умолчанию.

3. Разверните конфигурацию по умолчанию — сервер Zabbix, веб-интерфейс (работающий на Nginx) и серверную часть базы данных (MySQL или PostgreSQL), каждый в своем контейнере на Alpine Linux.

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

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

После того как контейнеры будут запущены и начнут работать (обычно в течение 1-3 минут), сервер Zabbix будет запущен, а веб-интерфейс будет доступен по адресу `http://localhost`.

> **Note:**
> Используйте `docker compose ps`, чтобы проверить состояние контейнеров. Все контейнеры (кроме контейнера `zabbix-docker-server-db-init-1`) должны иметь статус `Up`. Если у какого-либо контейнера статус `Exited`, проверьте его журналы на наличие ошибок с помощью `docker logs -f <container-name>`.

Переменные окружения

Поведение Docker Compose и конфигурацию компонентов Zabbix можно настраивать с помощью переменных окружения.

**Переменные уровня Compose** (определены в файле `.env`) управляют тем, какие образы Docker, порты и диапазоны IP сети используются. Вы можете использовать эти переменные inline перед командой `docker compose` или отредактировать файл `.env`.

Например, следующая команда разворачивает полную многоконтейнерную конфигурацию Zabbix с использованием образа на базе Ubuntu (вместо стандартного Alpine Linux) и предоставляет доступ к веб-интерфейсу через Nginx на пользовательских портах HTTP (8282) и HTTPS (8443):

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

**Переменные уровня компонентов** (определены в файлах env_vars/.env_<component>) управляют конфигурацией компонентов Zabbix. Отредактируйте соответствующий файл .env_<component> перед выполнением команды docker compose.

Например, вы можете изменить количество пассивных poller'ов сервера Zabbix и параметры кэша, а также настроить часовой пояс веб-интерфейса, отредактировав следующие переменные:

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

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

> **Note:**
> Переменные окружения уровня компонентов соответствуют параметрам конфигурации компонентов Zabbix, но используют другой стиль именования (например, ZBX_STARTPOLLERS соответствует StartPollers). Некоторые переменные зависят от Docker, а некоторые параметры конфигурации изменить нельзя (например, PIDFile и LogType). При использовании переменных окружения см. раздел *Environment variables* в описании образа Docker каждого компонента на Docker Hub.

Тома

Docker Compose сохраняет постоянные данные в каталоге zbx_env/, созданном вместе с файлом Compose. Этот каталог сохраняет данные компонентов при перезапуске и обновлении контейнера.

Содержимое zbx_env/ предопределено для образа каждого компонента. Например:

- zbx_env/etc/ — используется контейнером веб-интерфейса Zabbix [en]
- zbx_env/usr/ and zbx_env/var/ — используется контейнером Zabbix сервера [en]

Для получения более подробной информации о каждом томе смотрите раздел *Allowed volumes [en]* в описании каждого компонента образа Docker на Docker Hub.

Makefile

Репозиторий Zabbix Dockerfiles также предоставляет Makefile как сокращение для типовых задач Docker Compose. Вместо полного команды docker compose вы можете использовать более короткие команды make (выполните make help, чтобы увидеть доступные параметры):

```
###                 (    Zabbix,   -     , MySQL,      Alpine Linux)
make up

###                   (    Zabbix,   -                    Nginx, PostgreSQL,      Ubuntu)
make up \
  OS=ubuntu \
  DB=pgsql \
  ZABBIX_WEB_NGINX_HTTP_PORT=8282 \
  ZABBIX_WEB_NGINX_HTTPS_PORT=8443
```

> **Attention:**
> При остановке или удалении контейнеров всегда указывайте тот же тип базы данных, который был развернут (например, make down DB=pgsql).

По умолчанию make up запускает только минимальный набор служб (сервер Zabbix, веб-интерфейс и база данных). Это делает настройку более легкой и позволяет не запускать компоненты, которые могут не понадобиться. Чтобы включить дополнительные компоненты, можно использовать следующие профили Compose:

```
###                 +     Zabbix, Java gateway,   -      SNMP traps:
make up COMPOSE_PROFILES=full

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

**Docker (ручное развертывание)** Используйте ручное развертывание, если хотите развернуть Zabbix поэтапно, запускать отдельные компоненты, интегрировать их с существующей средой или использовать альтернативную среду

выполнения контейнеров, например Podman.

Например, чтобы развернуть прокси Zabbix, выполните следующую команду:

```
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-7.4-latest
```

Эта команда:

- Загружает образ Docker `zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest`.
- Создает и запускает контейнер `zabbix-proxy-sqlite3` на основе образа Docker (с флагами `--init` и `-d`).
- Настраивает параметр Zabbix proxy `Server` через переменную среды `ZBX_SERVER_HOST`. Этот параметр задает IP-адрес сервера Zabbix, с которого прокси будет получать данные конфигурации и на который будет отправлять собранные данные. Обратите внимание, что другие переменные, необходимые для работы активного прокси, используют значения по умолчанию и могут быть опущены.

**Note:**
Переменные среды соответствуют параметрам конфигурации компонентов Zabbix, но используют другой стиль именования (например, `ZBX_SERVER_HOST` соответствует `Server`). Некоторые переменные зависят от Docker, а некоторые параметры конфигурации изменить нельзя (например, `PIDFile` и `LogType`). При использовании переменных среды см. раздел *Environment variables* в описании образа Docker каждого компонента на Docker Hub.

- Подключает хранилище, управляемое Docker (например, `/var/lib/docker/volumes/zabbix-proxy-data/`), к каталогу `/var/lib/zabbix` контейнера, чтобы данные прокси Zabbix сохранялись постоянно, даже если контейнер будет удален.

**Note:**
Дополнительные сведения о каждом томе см. в разделе *Allowed volumes* в описании образа Docker каждого компонента на Docker Hub.

После развертывания контейнера прокси Zabbix вы можете перейти к добавлению прокси в веб-интерфейсе Zabbix и настройке узлов сети для мониторинга этим прокси.

Приведенные ниже примеры охватывают еще три сценария развертывания:

- сервер Zabbix (MySQL) с Java gateway
- сервер Zabbix (PostgreSQL) с SNMP traps
- сервер Zabbix (MySQL) с Java gateway на RHEL 8-10

Дополнительные примеры см. в описании образа Docker каждого компонента на Docker Hub.

Развертывание сервера Zabbix (MySQL) с Java gateway

В этом примере показано, как развернуть сервер Zabbix с бэкендом MySQL, Java gateway для мониторинга JMX и веб-интерфейсом на базе Nginx.

1. Создайте выделенную Docker-сеть, чтобы все контейнеры компонентов Zabbix могли обращаться друг к другу по имени контейнера:

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

2. Запустите пустой контейнер сервера MySQL (замените `zabbix_pwd` и `root_pwd` на надежные пароли и используйте эти значения в следующих шагах):

```
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:8.0-oracle \
  --character-set-server=utf8mb4 --collation-server=utf8mb4_bin
```

3. Просмотрите журналы MySQL и дождитесь, пока MySQL будет готов к подключениям (нажмите Ctrl+C, чтобы выйти из просмотра журналов):

```
docker logs -f mysql-server

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

4. Запустите контейнер Zabbix Java gateway:

```
docker run --name zabbix-java-gateway -t \
  --network=zabbix-net \
  --restart unless-stopped \
  -d zabbix/zabbix-java-gateway:alpine-7.4-latest
```

5. Включите log_bin_trust_function_creators в контейнере сервера MySQL. Это временно ослабляет ограничения безопасности для хранимых функций; в последующих шагах параметр будет отключен. Это позволяет пользователю базы данных Zabbix создавать хранимые функции без привилегии SUPER, которая требуется MySQL при включенном двоичном журналировании (по умолчанию с MySQL 8.0). Выполните следующую команду:

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

6. Запустите контейнер сервера Zabbix и подключите его к контейнерам Java Gateway и сервера 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-7.4-latest
```

7. Просмотрите журналы сервера Zabbix и дождитесь завершения инициализации схемы базы данных сервером Zabbix (нажмите Ctrl+C, чтобы выйти из просмотра журналов):

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

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

8. Отключите log_bin_trust_function_creators в контейнере сервера MySQL. Это снова включает ограничение безопасности, которое запрещает пользователям без привилегии SUPER создавать хранимые функции. Выполните следующую команду:

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

9. Запустите контейнер веб-интерфейса Zabbix и подключите его к контейнерам сервера Zabbix и сервера 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-7.4-latest
```

После запуска контейнеров сервер Zabbix будет запущен, а веб-интерфейс будет доступен по адресу `http://localhost`.

> **Note:**
> Используйте `docker ps`, чтобы проверить состояние контейнеров. Все контейнеры должны иметь статус `Up`. Если какой-либо контейнер имеет статус `Exited`, проверьте его журналы на наличие ошибок с помощью `docker logs -f <container-name>`.

Развертывание сервера Zabbix (PostgreSQL) с SNMP traps

В этом примере показано, как развернуть сервер Zabbix с бэкендом PostgreSQL, SNMP traps и веб-интерфейсом на базе Nginx.

1. Создайте выделенную Docker-сеть, чтобы все контейнеры компонентов Zabbix могли обращаться друг к другу по имени контейнера:

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

2. Запустите пустой контейнер сервера PostgreSQL (замените `zabbix_pwd` на надежный пароль и используйте это значение в следующих шагах):

```
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. Запустите контейнер Zabbix SNMP traps:

```
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-7.4-latest
```

4. Запустите контейнер сервера Zabbix и подключите его к контейнерам сервера PostgreSQL и SNMP traps:

```
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-7.4-latest
```

5. Просмотрите журналы сервера Zabbix и дождитесь завершения инициализации схемы базы данных сервером Zabbix (нажмите `Ctrl+C`, чтобы выйти из просмотра журналов):

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

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

6. Запустите контейнер веб-интерфейса Zabbix и подключите его к контейнерам сервера Zabbix и сервера 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-7.4-latest
```

После запуска контейнеров сервер Zabbix будет запущен, а веб-интерфейс будет доступен по адресу `http://localhost`.

> **Note:**
> Используйте `docker ps`, чтобы проверить состояние контейнеров. Все контейнеры должны иметь статус `Up`. Если какой-либо контейнер имеет статус `Exited`, проверьте его журналы на наличие ошибок с помощью `docker logs -f <container-name>`.

Развертывание сервера Zabbix (MySQL) с Java gateway на RHEL 8-10

В этом примере показано, как развернуть сервер Zabbix с бэкендом MySQL, Java gateway для мониторинга JMX и веб-интерфейсом на базе Nginx, работающими в Red Hat Enterprise Linux 8, 9 или 10.

В Red Hat Enterprise Linux рекомендуемая среда выполнения контейнеров — Podman, а не Docker. Podman работает аналогично Docker, но не требует фоновой службы, запущенной от root, что делает его более подходящим для сред Red Hat.

1. Создайте новый pod с именем `zabbix` и открытыми портами для веб-интерфейса Zabbix и trapper сервера Zabbix:

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

2. (Необязательно) Запустите контейнер агента Zabbix в pod `zabbix`:

```
podman run --name zabbix-agent \
  -e ZBX_SERVER_HOST="127.0.0.1,localhost" \
  --restart=always \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-agent-74:latest
```

3. Создайте каталог `./mysql/` на хосте RHEL:

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

4. Запустите пустой контейнер сервера MySQL (замените `zabbix_pwd` и `root_pwd` на надежные пароли и используйте эти значения в следующих шагах):

```
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:8.0 \
  --character-set-server=utf8mb4 --collation-server=utf8mb4_bin
```

5. Просмотрите журналы MySQL и дождитесь, пока MySQL будет готов к подключениям (нажмите `Ctrl+C`, чтобы выйти из просмотра журналов):

```
podman logs -f mysql-server

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

6. Запустите контейнер Zabbix Java gateway:

```
podman run --name zabbix-java-gateway -t \
  --restart=always \
  --pod=zabbix \
  -d registry.connect.redhat.com/zabbix/zabbix-java-gateway-74
```

7. Включите log_bin_trust_function_creators в контейнере сервера MySQL. Это временно ослабляет ограничения безопасности для хранимых функций; в последующих шагах параметр будет отключен. Это позволяет пользователю базы данных Zabbix создавать хранимые функции без привилегии SUPER, которая требуется MySQL при включенном двоичном журналировании (по умолчанию с MySQL 8.0). Выполните следующую команду:

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

8. Запустите контейнер сервера Zabbix и подключите его к контейнерам Java Gateway и сервера 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-74
```

9. Просмотрите журналы сервера Zabbix и дождитесь завершения инициализации схемы базы данных сервером Zabbix (нажмите Ctrl+C, чтобы выйти из просмотра журналов):

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

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

10. Отключите log_bin_trust_function_creators в контейнере сервера MySQL. Это снова включает ограничение безопасности, которое запрещает пользователям без привилегии SUPER создавать хранимые функции. Выполните следующую команду:

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

11. Запустите контейнер веб-интерфейса Zabbix и подключите его к контейнерам сервера Zabbix и сервера 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-74
```

> **Note:**
> Pod zabbix открывает порт 80/TCP для хоста из 8080/TCP контейнера zabbix-web-mysql.

После запуска контейнеров сервер Zabbix будет запущен, а веб-интерфейс будет доступен по адресу http://localhost.

> **Note:**
> Используйте docker ps, чтобы проверить состояние контейнеров. Все контейнеры должны иметь статус Up. Если какой-либо контейнер имеет статус Exited, проверьте его журналы на наличие ошибок с помощью docker logs -f <container-name>.

### 6. Установка веб-интерфейса

Данный раздел приводит пошаговую инструкцию по установке веб-интерфейса Zabbix. Веб-интерфейс Zabbix написан на PHP, в связи с чем для его запуска необходим веб-сервер с поддержкой PHP.

**Note:**
Вы можете найти более подробную информацию о настройке SSL для веб-интерфейса Zabbix, обращаясь к этим наилучшим практикам.

Экран приветствия

Откройте в браузере URL веб-интерфейса Zabbix. Если вы установили Zabbix из пакетов, то URL:

- для Apache: *http://<ip_или_имя_сервера>/zabbix*
- для Nginx: *http://<ip_или_имя_сервера>*

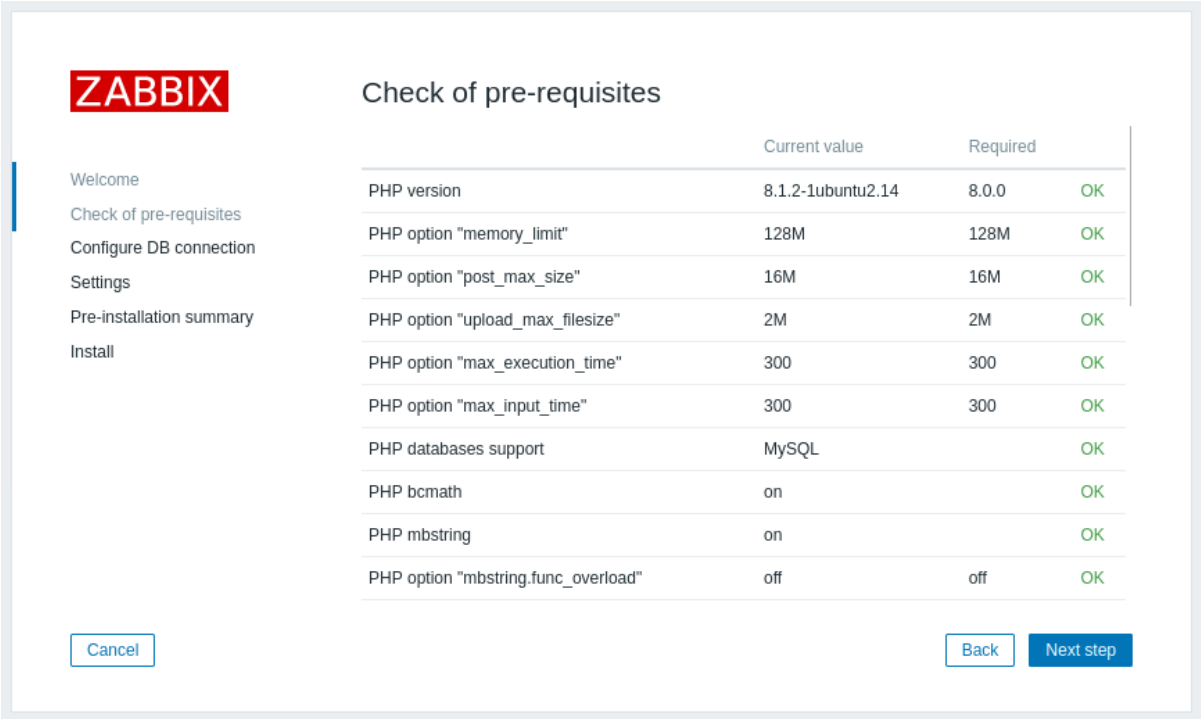
Вы должны увидеть первый экран мастера установки веб-интерфейса.

Используйте выпадающее меню *Default language*, чтобы изменить язык системы по умолчанию, и продолжите установку на выбранном языке (опционально). Для получения более подробной информации обратитесь к разделу Установка дополнительных языков для веб-интерфейса.

Обратите внимание, что установка языка в *English (en_US)* также включит в веб-интерфейсе формат даты/времени США.

Проверка предварительных требований

Убедитесь, что все обязательные предварительные требования веб-интерфейса Zabbix выполнены.




| Предварительное требование | Минимальное значение | Описание |
|---|---|---|
| *Версия PHP* | 8.0.0 | |
| *Параметр PHP "memory_limit"* | 128MB | В php.ini: memory_limit = 128M |
| *Параметр PHP "post_max_size"* | 16MB | В php.ini: post_max_size = 16M |
| *Параметр PHP "upload_max_filesize"* | 2MB | В php.ini: upload_max_filesize = 2M |
| *Параметр PHP "max_execution_time"* | 300 seconds | В php.ini: max_execution_time = 300 (значения 0 и -1 также допускаются) |
| *Параметр PHP "max_input_time"* | 300 seconds | В php.ini: max_input_time = 300 (значения 0 и -1 также допускаются) |
| *Поддержка баз данных PHP* | Одна из: MySQL, PostgreSQL | См. Requirements для списка всех обязательных и необязательных расширений PHP. Обратите внимание, что невыполненные необязательные предварительные требования отображаются со статусом *Warning* красного цвета, и процесс установки может продолжиться, даже если они не выполнены. |
| *PHP bcmath* | должен быть включен | |
| *PHP mbstring* | должен быть включен | |
| *Параметр PHP "mbstring.func_overload"* | должен быть отключен | В php.ini: mbstring.func_overload = 0 |
| *Параметр PHP "session.auto_start"* | должен быть отключен | В php.ini: session.auto_start = 0 |
| *Параметр PHP "arg_separator.output"* | & | В php.ini: arg_separator.output = "&" (значение "&" также допускается) |

**Attention:**
Если требуется изменить пользователя или группу пользователей Apache, проверьте права доступа к папке сеансов; в противном случае установка Zabbix может не продолжиться.

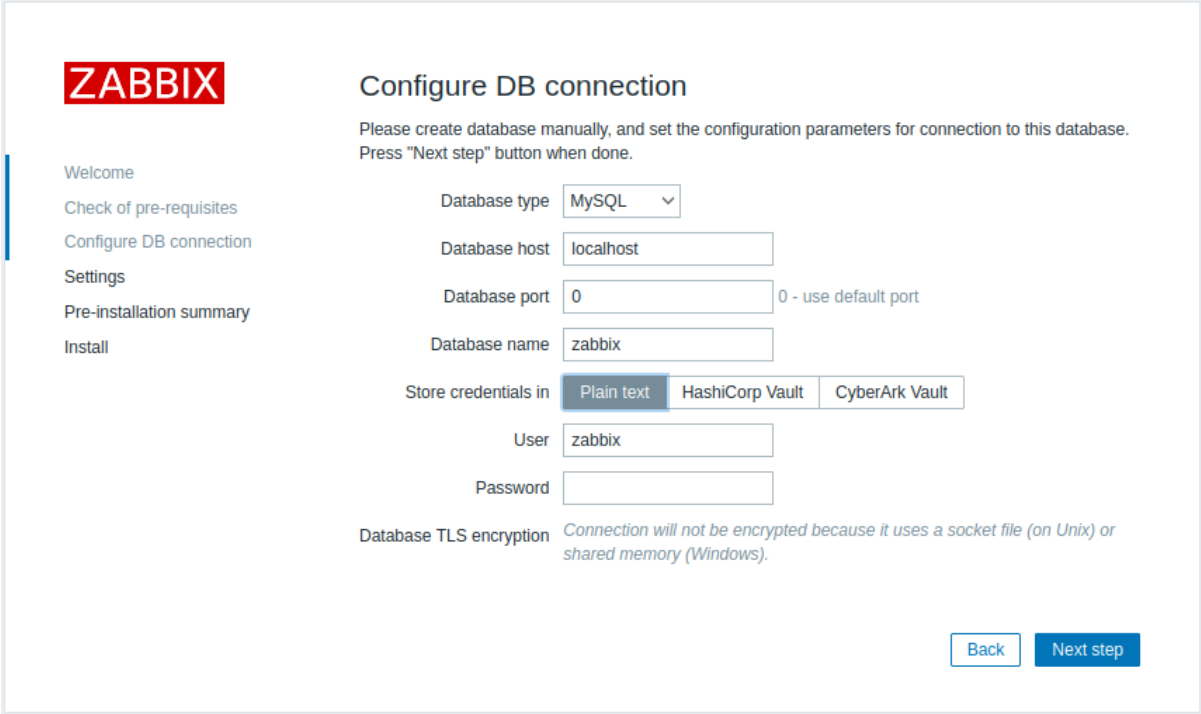
Настройка соединения с базой данных

Укажите информацию для подключения к базе данных. База данных Zabbix должна быть уже создана.

В случае MySQL, если ввести `localhost` или оставить поле *Database host* пустым, то это приведёт к установлению соединения через Unix сокет по умолчанию. Диалог настройки не предоставляет отдельного поля *Database socket*, поэтому для использования кастомизируемого сокета сконфигурируйте настройки Zabbix сервера (например, указывая

параметр DBSocket= в файле zabbix_server.conf). Это сохранит веб-интерфейс синхронизированным с настройками соединения сервера к базе данных.

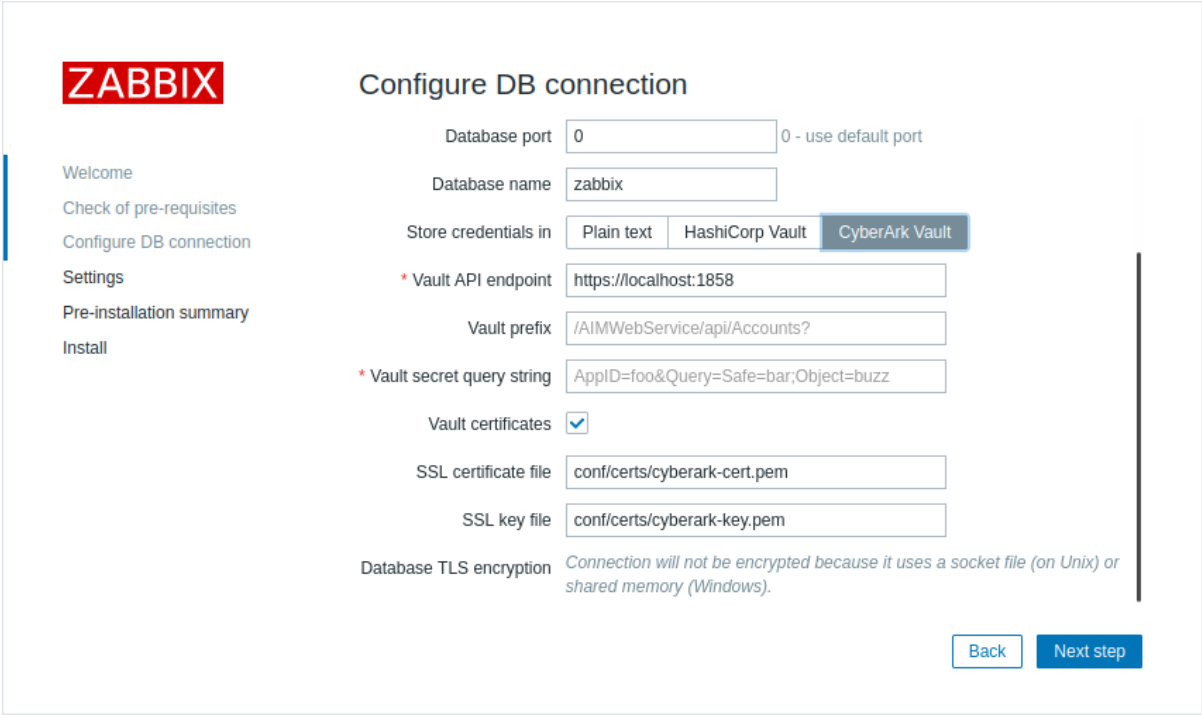
В случае PostgreSQL, если поле *Database host* оставить пустым, используется Unix-domain сокет по умолчанию. Если ввести путь сокета (например, /var/run/pgbouncer), то этот Unix-domain сокет и будет использоваться.




Если выбрана опция *TLS шифрование базы данных (Database TLS encryption)*, в диалоге появятся дополнительные поля для настройки TLS соединения к базе данных (только для MySQL или PostgreSQL).

Если опция *Хранить учетные данные в (Store credentials in)* выставлена в HashiCorp Vault или CyberArk Vault, станут доступны дополнительные поля:

- для HashiCorp Vault: API endpoint Хранилища, префикс Хранилища, путь к секрету и токен аутентификации;
- для CyberArk Vault: API endpoint Хранилища, префикс Хранилища, строка запроса секрета и сертификаты. При установке флажка *Сертификаты Хранилища (Vault certificates)* появятся два новых поля для указания путей к файлу сертификата SSL и файлу ключа SSL.

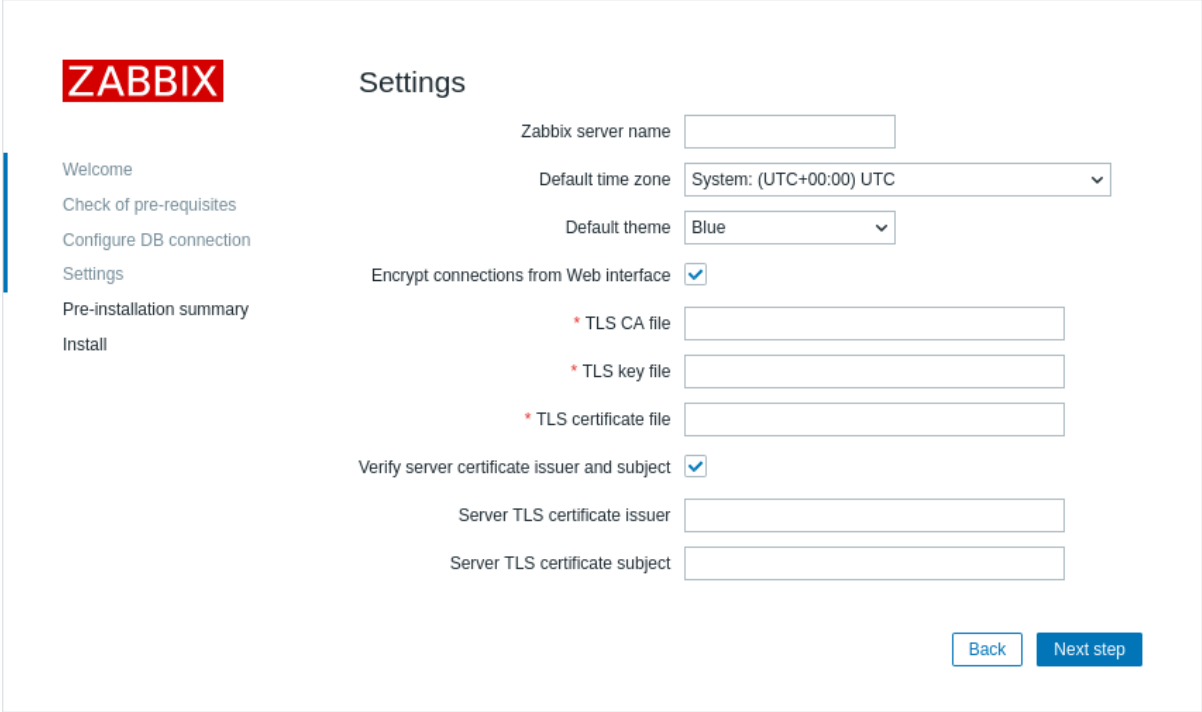



Настройки

Указывать имя для Zabbix server необязательно, однако если оно задано, оно будет отображаться в строке меню и заголовках страниц.

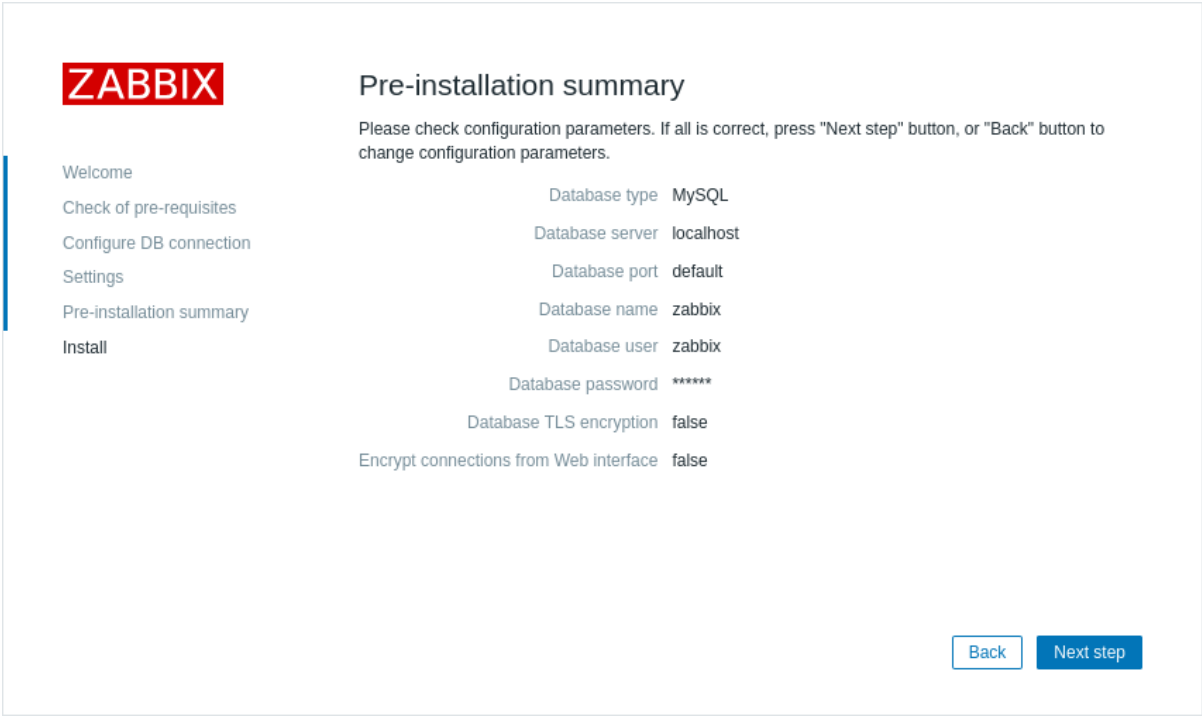
Задайте часовой пояс по умолчанию time zone и тему для веб-интерфейса.

Если установлен флажок *Encrypt connections from Web interface*, в форме появятся дополнительные поля для настройки TLS-соединения между Zabbix server и веб-интерфейсом.

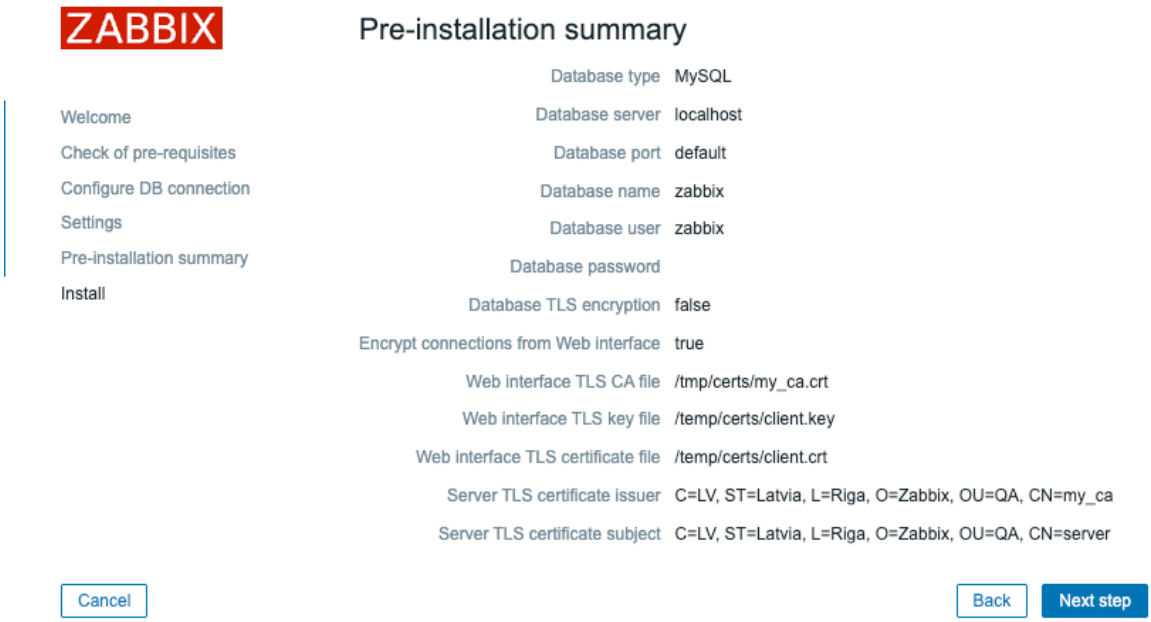



Краткое описание перед установкой

Просмотрите сводку настроек.

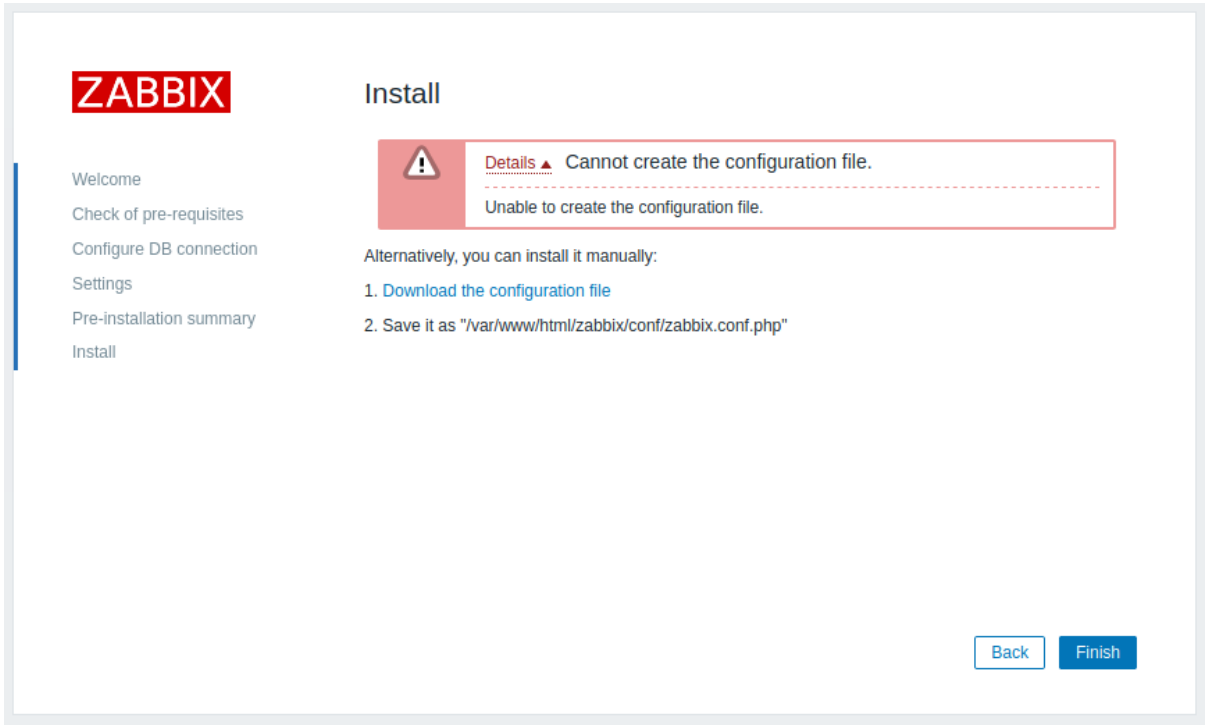


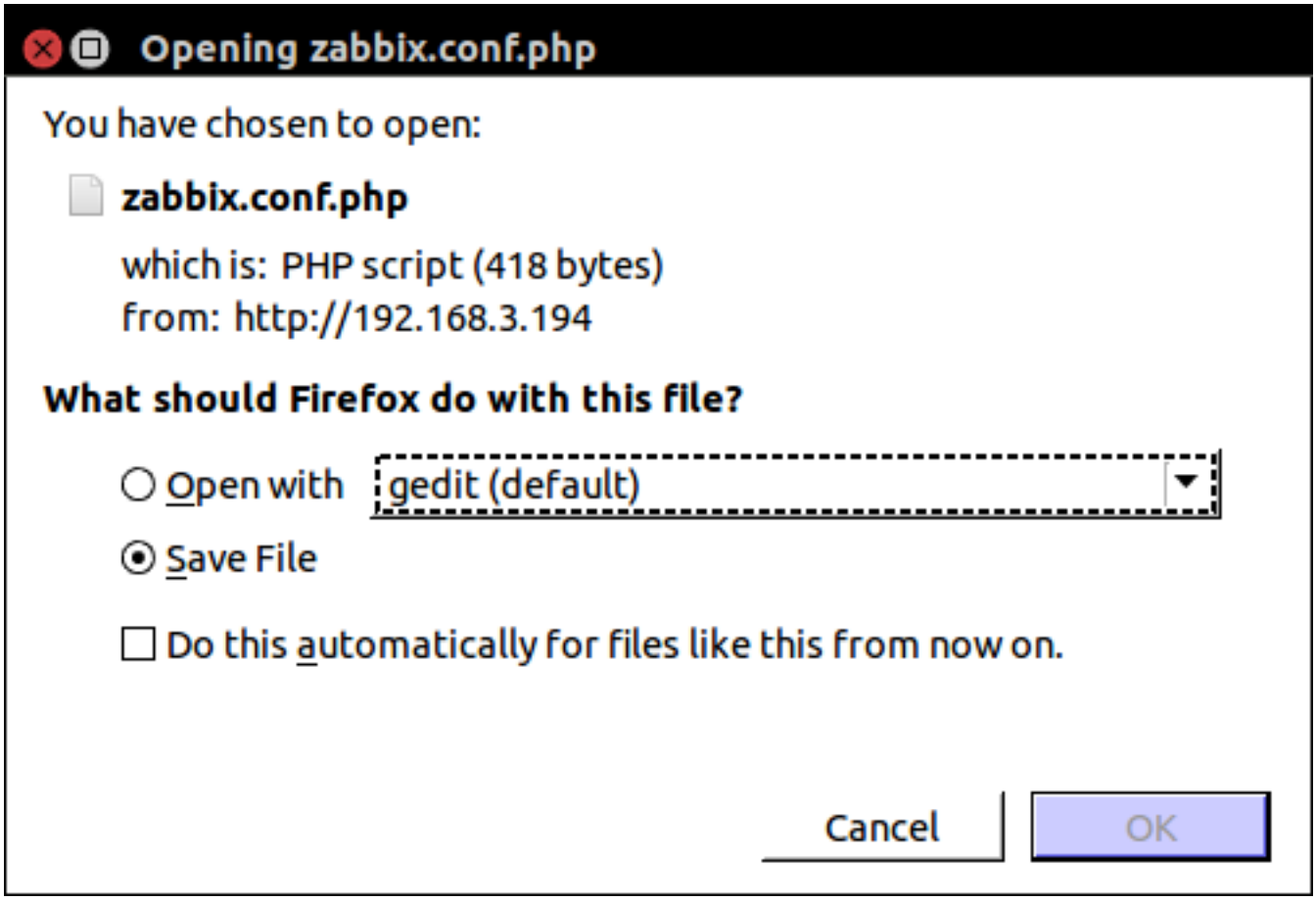

На подстранице будут отображаться данные, если была добавлена конфигурация TLS.




Установка

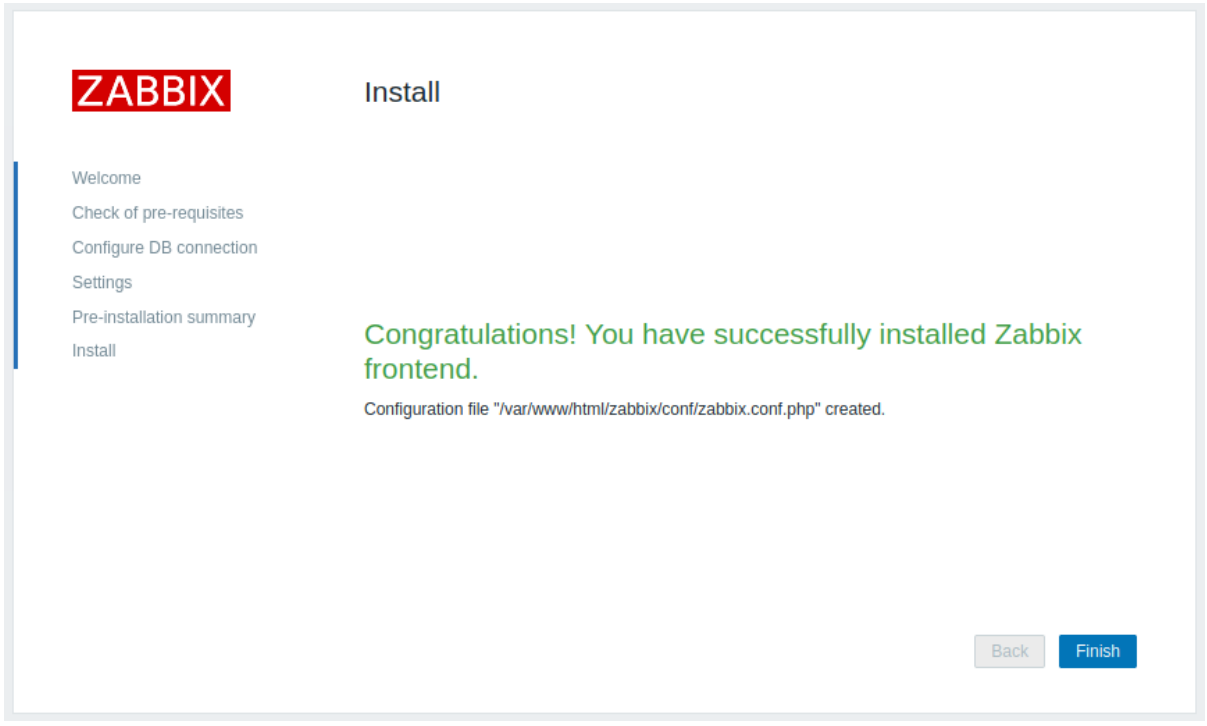
Если вы устанавливаете Zabbix из исходных кодов, скачайте файл конфигурации и поместите его в каталог conf/ в подкаталоге HTML-документов веб-сервера, куда вы скопировали PHP-файлы Zabbix.





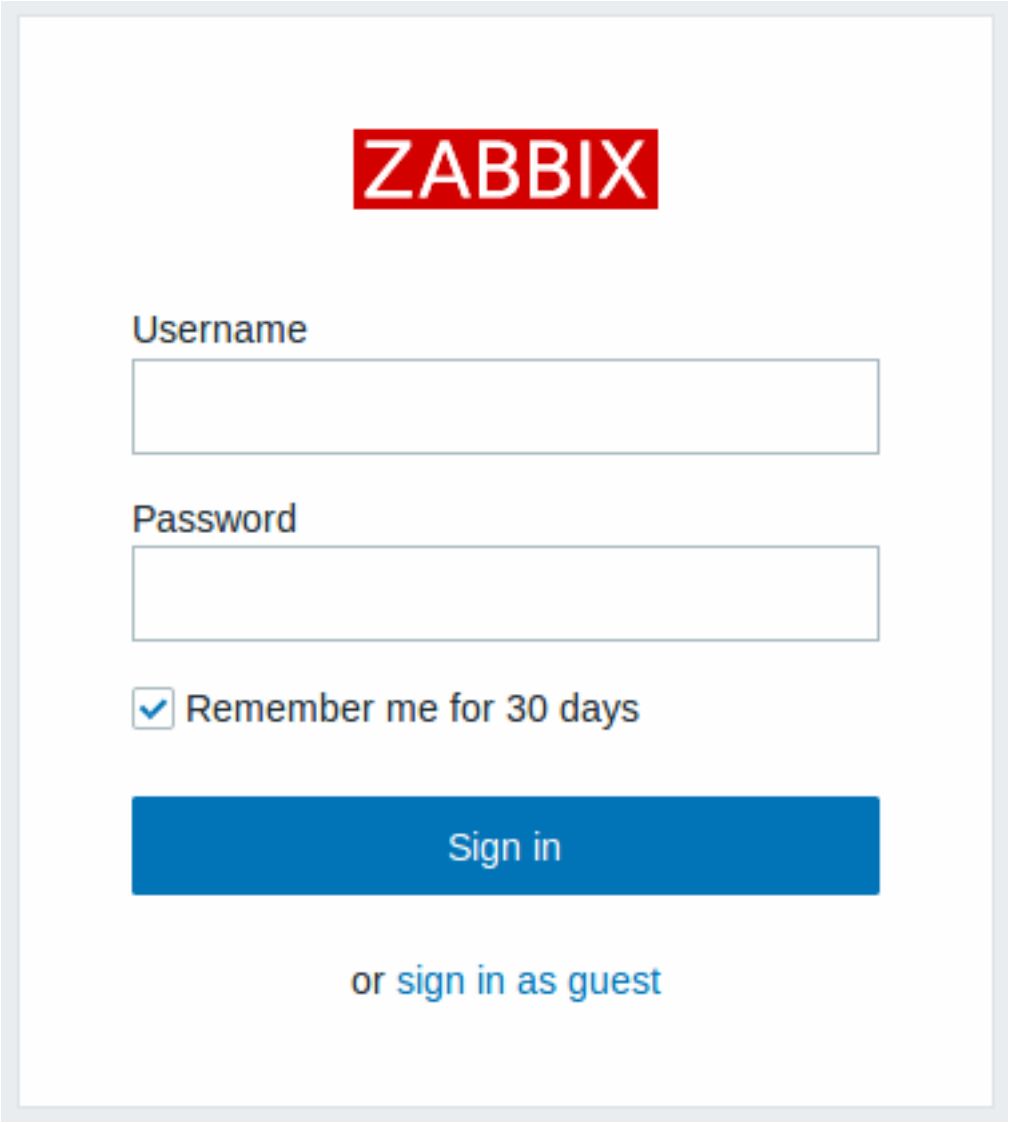

**Note:**
Если у пользователя веб-сервера есть права на запись в каталог conf/, файл конфигурации будет сохранен автоматически, и можно будет сразу перейти к следующему шагу.

Завершите установку.




Вход в систему

Веб-интерфейс Zabbix готов! Имя пользователя по умолчанию — **Admin**, пароль — **zabbix**.




Перейдите к началу работы с Zabbix.

**7. Процедура обновления**

Обзор

В этом разделе приведена информация об обновлении Zabbix **7.4**:

- Использование пакетов для Red Hat Enterprise Linux или Debian/Ubuntu
- Использование исходных кодов
- Использование контейнеров

Связанные инструкции:

- Для серверов в кластере высокой доступности (HA) см. Обновление HA-кластера
- Для базы данных TimescaleDB см. Обновление схемы TimescaleDB

Обновление агентов Zabbix рекомендуется, но не является обязательным.

Настоятельно рекомендуется обновить Zabbix прокси. Сервер Zabbix полностью поддерживает прокси такой же основной версии, как и сервер. Сервер Zabbix также поддерживает прокси, которые **не старше** предыдущей LTS-версии сервера Zabbix, но с ограниченной функциональностью (сбор данных, выполнение удалённых команд, незамедлительная проверка значения элемента данных). Обновление конфигурации также отключено, и не обновлённые прокси будут работать только со старой конфигурацией.

**Attention:**
Прокси старше предыдущей LTS-версии сервера Zabbix или новее основной версии сервера Zabbix не поддерживаются. Сервер Zabbix будет игнорировать данные от неподдерживаемых прокси-серверов, и вся связь с сервером Zabbix прервётся с предупреждением. Для получения дополнительной информации смотрите Совместимость версий.

Чтобы свести к минимуму время простоя и потерю данных в процессе обновления, рекомендуется останавливать, обновлять и запускать сервер Zabbix и затем останавливать, обновлять и запускать Zabbix прокси один за другим. Во время простоя сервера работающие прокси продолжают собирать данные. Как только сервер будет запущен и заработает, не обновлённые прокси отправят данные новому серверу (хотя конфигурация прокси не будет обновлена) и частично останутся функциональными. Любые уведомления о проблемах во время простоя сервера Zabbix будут сгенерированы только после запуска обновлённого сервера.

Если прокси Zabbix запускается впервые и файл базы данных SQLite отсутствует, прокси автоматически создаст его.

**Обратите внимание**, что если прокси Zabbix использует SQLite3 и при запуске обнаруживает, что версия существующего файла базы данных старее требуемой, он **автоматически удалит файл базы данных** и создаст новый. Поэтому исторические данные, хранящиеся в файле базы данных SQLite, будут потеряны. Если версия прокси Zabbix старее версии файла базы данных, Zabbix запишет ошибку в журнал и завершит работу.

В зависимости от размера базы данных обновление базы данных до версии 7.4 может занять много времени.

Прямое обновление до Zabbix 7.4.x возможно с Zabbix **7.2**.x, **7.0**.x, **6.4**.x, **6.2**.x, **6.0**.x, **5.4**.x, **5.2**.x, **5.0**.x, **4.4**.x, **4.2**.x, **4.0**.x, **3.4**.x, **3.2**.x, **3.0**.x, **2.4**.x, **2.2**.x и **2.0**.x. Для обновления с более ранних версий обратитесь к документации Zabbix для 2.0 и более ранних версий.

**Note:**
Обратите внимание, что после обновления некоторые интеграции Zabbix со сторонним программным обеспечением могут быть затронуты, если внешнее программное обеспечение несовместимо с обновленной версией Zabbix.

Доступны следующие примечания к обновлению:

| Обновление с | Полные примечания к обновлению | Наиболее важные изменения между версиями |
|---|---|---|
| 7.2.x | Для:<br>Zabbix 7.4 | Поддержка библиотеки PCRE прекращена. |
| 7.0.x LTS | Для:<br>Zabbix 7.2<br>Zabbix 7.4 | Поддержка Oracle DB прекращена. |
| 6.4.x | Для:<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемая версия PHP повышена с 7.4.0 до 8.0.0.<br>Асинхронные опросчики для проверок agent, HTTP agent, SNMP walk[OID] и SNMP get[OID].<br>Отдельная таблица базы данных для прокси.<br>Изменено расположение файла конфигурации агента для Windows.<br>Oracle DB объявлена устаревшей.<br>Старый числовой тип значения (float) удален. |
| 6.2.x | Для:<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемая версия MySQL повышена с 8.0.0 до 8.0.30.<br>Для сервера/прокси Zabbix требуется библиотека 'libevent_pthreads'.<br>При первом запуске после обновления прокси Zabbix с SQLite3 автоматически удаляет старую версию базы данных (со всей историей) и создает новую. |
| 6.0.x LTS | Для:<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемая версия PHP повышена с 7.2.5 до 7.4.0.<br>Мониторинг сервисов существенно переработан.<br>Во время обновления необходимо создать детерминированные триггеры. Если для MySQL/MariaDB включено двоичное журналирование, для этого требуются привилегии суперпользователя или установка переменной/параметра конфигурации *log_bin_trust_function_creators = 1*. Инструкции по установке переменной см. в разделе Скрипты создания базы данных. |

| Обновление с | Полные примечания к обновлению | Наиболее важные изменения между версиями |
|---|---|---|
| 5.4.x | Для:<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемые версии баз данных повышены.<br>Сервер/прокси не запустится при устаревшей базе данных.<br>Записи журнала аудита потеряны из-за изменения структуры базы данных. |
| 5.2.x | Для:<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемые версии баз данных повышены.<br>Агрегированные элементы данных удалены как отдельный тип. |
| 5.0.x LTS | Для:<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемая версия PHP повышена с 7.2.0 до 7.2.5.<br>Алгоритм хеширования паролей изменен с MD5 на bcrypt. |
| 4.4.x | Для:<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Поддержка IBM DB2 прекращена.<br>Минимально требуемая версия PHP повышена с 5.4.0 до 7.2.0.<br>Минимально требуемые версии баз данных повышены.<br>Изменен каталог файлов PHP Zabbix. |
| 4.2.x | Для:<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Типы медиа Jabber и Ez Texting удалены. |
| 4.0.x LTS | Для:<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Старые прокси больше не могут передавать данные на обновленный сервер.<br>Новые агенты больше не смогут работать со старым сервером Zabbix. |

| Обновление с | Полные примечания к обновлению | Наиболее важные изменения между версиями |
|---|---|---|
| 3.4.x | Для:<br>Zabbix 4.0<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Теперь библиотеки ’libpthread’ и ’zlib’ обязательны.<br>Поддержка простого текстового протокола прекращена, и заголовок обязателен.<br>Агенты Zabbix версии до 1.4 больше не поддерживаются.<br>Параметр Server в пассивной конфигурации прокси теперь обязателен. |
| 3.2.x | Для:<br>Zabbix 3.4<br>Zabbix 4.0<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Поддержка SQLite в качестве внутренней базы данных для сервера Zabbix/веб-интерфейса прекращена.<br>Поддерживаются регулярные выражения, совместимые с Perl (PCRE), вместо расширенных POSIX.<br>Для сервера Zabbix обязательны библиотеки ’libpcre’ и ’libevent’.<br>Добавлены проверки кода завершения для пользовательских параметров, удаленных команд и элементов данных system.run[] без флага ’nowait’, а также для скриптов, выполняемых сервером Zabbix.<br>Для поддержки новой функциональности необходимо обновить Java gateway Zabbix. |
| 3.0.x LTS | Для:<br>Zabbix 3.2<br>Zabbix 3.4<br>Zabbix 4.0<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Обновление базы данных может выполняться медленно в зависимости от размера таблицы истории. |
| 2.4.x | Для:<br>Zabbix 3.0<br>Zabbix 3.2<br>Zabbix 3.4<br>Zabbix 4.0<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемая версия PHP повышена с 5.3.0 до 5.4.0.<br>Необходимо указать параметр агента LogFile. |

| Обновление с | Полные примечания к обновлению | Наиболее важные изменения между версиями |
|---|---|---|
| 2.2.x LTS | Для:<br>Zabbix 2.4<br>Zabbix 3.0<br>Zabbix 3.2<br>Zabbix 3.4<br>Zabbix 4.0<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Удален распределенный мониторинг на основе узлов. |
| 2.0.x | Для:<br>Zabbix 2.2<br>Zabbix 2.4<br>Zabbix 3.0<br>Zabbix 3.2<br>Zabbix 3.4<br>Zabbix 4.0<br>Zabbix 4.2<br>Zabbix 4.4<br>Zabbix 5.0<br>Zabbix 5.2<br>Zabbix 5.4<br>Zabbix 6.0<br>Zabbix 6.2<br>Zabbix 6.4<br>Zabbix 7.0<br>Zabbix 7.2<br>Zabbix 7.4 | Минимально требуемая версия PHP повышена с 5.1.6 до 5.3.0.<br>Для корректной работы сервера требуется база данных MySQL с учетом регистра; для корректной работы сервера Zabbix с базой данных MySQL требуется набор символов utf8 и сопоставление utf8_bin. См. скрипты создания базы данных.<br>Требуется расширение PHP 'mysqli' вместо 'mysql'. |

#### 1. Обновление из исходных кодов

Обзор

В этом разделе приведены шаги, необходимые для успешного обновления с Zabbix **7.2**.x до Zabbix **7.4**.x с использованием официальных исходных кодов Zabbix.

**Warning:**
Перед обновлением обязательно ознакомьтесь с соответствующими **примечаниями к обновлению**!

Также рекомендуется ознакомиться с требованиями для 7.4.

**Note:**
Во время обновления может быть удобно использовать две параллельные SSH-сессии: в одной выполнять шаги обновления, а в другой отслеживать журналы сервера/прокси. Например, во второй SSH-сессии можно запустить `tail -f zabbix_server.log` или `tail -f zabbix_proxy.log`, чтобы в реальном времени видеть последние записи журнала и возможные ошибки. Для рабочих экземпляров это может быть критически важно.

Процесс обновления сервера

1. Останов сервера

Остановите Zabbix сервер, чтобы быть уверенным, что в базу данных не будет происходить запись новых данных.

2. Резервное копирование существующей базы данных Zabbix

Этот шаг очень важен. Убедитесь, что у вас имеется резервная копия вашей базы данных. Копия поможет, если процедура обновления закончится неудачно (отсутствие свободного места на диске, отключение питания, любая непредвиденная проблема).

3. Резервное копирование файлов конфигурации, PHP файлов и бинарных файлов Zabbix

Выполните резервное копирование бинарных файлов Zabbix, файлов конфигурации и папки с PHP файлами.

4 Установка новых бинарных файлов сервера

Используйте эти instructions для компиляции сервера Zabbix из исходных кодов.

5 Проверьте параметры конфигурации сервера

Обязательно ознакомьтесь с Примечаниями к обновлению, чтобы проверить, требуются ли изменения в параметрах конфигурации.

С новыми необязательными параметрами можно ознакомиться на странице Что нового.

6. Запуск новых бинарных файлов Zabbix

Запустите новые бинарные файлы. Проверьте файлы журналов, чтобы удостовериться, что сервер запустился успешно.

Zabbix сервер автоматически обновит базу данных. При запуске Zabbix сервер сообщает текущую (обязательную и опциональную) и требуемую версии базы данных. Если текущая обязательная версия старее требуемой версии, Zabbix сервер автоматически выполнит требуемые патчи обновления базы данных. Начало и прогресс (в процентах) обновления базы данных записываются в файл журнала Zabbix сервера. Когда обновление завершится, в файл журнала запишется сообщение «database upgrade fully completed». В случае сбоя какого-либо из патчей обновления Zabbix сервер не запустится. Zabbix сервер также не запустится, если текущая обязательная версия более новая, чем требуемая. Zabbix сервер запустится, только если текущая обязательная версия базы данных соответствует требуемой обязательной версии.

```
8673:20161117:104750.259 current database version (mandatory/optional): 03040000/03040000
8673:20161117:104750.259 required mandatory version: 03040000
```

До запуска сервера:

- Убедитесь, что пользователь базы данных имеет достаточно прав (create table, drop table, create index, drop index).
- Убедитесь что у вас достаточно свободного дискового пространства.

7. Установка нового веб-интерфейса Zabbix

Минимально требуемая версия PHP — 8.0.0. Обновите, если потребуется, и следуйте инструкции по установке.

8. Очистка cookies и кэша в веб-браузере

После обновления вам, возможно, потребуется очистить cookies веб-браузера и кэш веб-браузера, чтобы веб-интерфейс Zabbix работал должным образом.

Процесс обновления прокси

1. Останов прокси

Остановите Zabbix прокси.

2. Резервное копирование файлов конфигурации и бинарного файла Zabbix прокси

Выполните резервное копирование бинарного файла и файла конфигурации Zabbix прокси.

3. Установка новых бинарных файлов прокси

Воспользуйтесь этой инструкцией для компиляции Zabbix прокси из исходных кодов.

4 Проверьте параметры конфигурации прокси

Обязательно ознакомьтесь с Примечаниями к обновлению, чтобы проверить, требуются ли изменения в параметрах конфигурации.

О новых необязательных параметрах см. страницу Что нового.

6. Запуск нового Zabbix прокси

Запустите новый Zabbix прокси. Проверьте файлы журналов, чтобы удостовериться, что прокси запустился успешно.

Zabbix прокси автоматически обновит базу данных. Обновление базы данных выполняется аналогично запуску Zabbix сервера.

Процесс обновления агента

**Attention:**
Обновление агентов не является обязательным. Вы можете обновить агенты, только если это требуется для доступа к новому функционалу.

Процедурой обновления, описанной в этом разделе, можно воспользоваться для обновления как Zabbix агента, так и Zabbix агента 2.

1. Останов агента

Остановите Zabbix агент.

2. Резервное копирование файлов конфигурации и бинарных файлов Zabbix агента

Выполните резервное копирование бинарного файла и файла конфигурации Zabbix агента.

3 Установка новых бинарных файлов агента

Используйте эти instructions для компиляции агента Zabbix из исходных кодов.

В качестве альтернативы вы можете загрузить предварительно скомпилированные агенты Zabbix со страницы загрузки Zabbix.

4 Просмотрите параметры конфигурации агента

Обязательно ознакомьтесь с Примечаниями к обновлению, чтобы проверить, требуются ли какие-либо изменения в параметрах конфигурации.

О новых необязательных параметрах см. страницу Что нового.

5. Запуск нового Zabbix агента

Запустите новый Zabbix агент. Проверьте файлы журналов, чтобы удостовериться, что агент запустился успешно.

Обновление между минорными версиями

При обновлении между минорными версиями 7.4.x (например, с 7.4.1 до 7.4.3) требуется выполнить те же действия для сервер/прокси/агент, что и при обновлении между мажорными версиями. Единственное отличие заключается в том, что при обновлении между минорными версиями изменения в базу данных не вносятся.

### 2. Обновление из пакетов

Обзор

В этом разделе описаны шаги, необходимые для успешного обновления с использованием официальных пакетов RPM и DEB, предоставляемых Zabbix, для:

- Red Hat Enterprise Linux
- Debian/Ubuntu

Пакеты Zabbix из репозиториев ОС

Некоторые дистрибутивы ОС (в частности, дистрибутивы на основе Debian) поставляют свои собственные пакеты Zabbix. Эти пакеты **не поддерживаются компанией Zabbix** и могут быть устаревшими, или в них могут отсутствовать последние функции и исправления ошибок. Рекомендуется использовать только официальные пакеты из официального репозитория Zabbix.

Если вы обновляетесь с пакетов, предоставленных дистрибутивами ОС (или они были установлены в какой-то момент), выполните эту процедуру, чтобы перейти на официальные пакеты Zabbix:

1. Удалите старые пакеты.
2. Проверьте на наличие и удалите все оставшиеся файлов из старой инсталляции.
3. Установите официальные пакеты, следуя инструкциям по установке, предоставленным компанией Zabbix.

Не выполняйте прямое обновление, так как такой подход может привести к неработоспособной инсталляции.

#### 1. Red Hat Enterprise Linux

Обзор

В этом разделе приведены инструкции по обновлению с Zabbix **7.2.x** до последней версии Zabbix **7.4.x** с использованием официальных пакетов Zabbix для Red Hat Enterprise Linux или его производных - AlmaLinux, CentOS Stream, Oracle Linux и Rocky Linux.

**Warning:**
Перед обновлением ознакомьтесь с соответствующими примечаниями по обновлению и убедитесь, что ваша система соответствует требованиям для Zabbix 7.4.

**Note:**
Рекомендуется использовать два параллельных SSH-сеанса во время обновления: один для выполнения шагов обновления, а другой для мониторинга журналов сервера/прокси. Например, во втором сеансе выполните `tail -f zabbix_server.log` или `tail -f zabbix_proxy.log`, чтобы в реальном времени просматривать последние записи журнала и возможные ошибки. Это может быть критически важно для рабочих экземпляров.

Инструкции по обновлению между минорными версиями Zabbix 7.4.x (например, с 7.4.1 до 7.4.3) см. в разделе Обновление между минорными версиями.

Процедура обновления

1. Остановите процессы Zabbix

Остановите Zabbix сервер, чтобы быть уверенным, что в базу данных не будет происходить запись новых данных.

```
systemctl stop zabbix-server
```

Если обновляете Zabbix прокси, агента или агента 2, то остановите и эти компоненты тоже.

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2. Сделайте резервную копию базы данных Zabbix

Сделайте резервную копию вашей существующей базы данных Zabbix, чтобы подстраховаться от возможных сбоев при обновлении (например, проблем с дисковым пространством, отключения электричества или непредвиденных проблем).

3. Сделайте резервную копию файлов конфигурации, PHP и бинарных файлов Zabbix

Сделайте резервную копию существующих файлов конфигурации, файлов PHP и бинарных файлов Zabbix.

Для файлов конфигурации выполните:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
cp /etc/httpd/conf.d/zabbix.conf  /opt/zabbix-backup/
```

Для файлов PHP и двоичных файлов Zabbix выполните:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Обновление пакета конфигурации репозитория

Перед продолжением обновления обновите текущий пакет репозитория до последней версии, чтобы обеспечить совместимость с новейшими пакетами и включить все недавние исправления безопасности или ошибки.

На **RHEL 10** выполните:

```
rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/rhel/10/noarch/zabbix-release-latest.el10.noarch.rpm
```

На **RHEL 9** выполните:

```
rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/rhel/9/noarch/zabbix-release-latest.el9.noarch.rpm
```

**Note:**
Для более старых версий RHEL или их производных замените ссылку выше на правильную из репозитория Zabbix. Однако обратите внимание, что пакеты для этих версий могут не включать все компоненты Zabbix, и для обновления этих компонентов из пакетов следует рассмотреть возможность обновления ОС. Список включенных компонентов см. в разделе Пакеты Zabbix.

Затем очистите кэш менеджера пакетов `dnf` (включая заголовки, метаданные и файлы пакетов, загруженные во время предыдущих установок или обновлений):

```
dnf clean all
```

При следующем выполнении dnf он загрузит свежие метаданные из репозиториев, поскольку старые метаданные будут удалены.

См. также: Известные проблемы для обновления пакета конфигурации репозитория в RHEL.

5 Обновление компонентов Zabbix

Чтобы обновить компоненты Zabbix, выполните:

```
dnf install zabbix-server-mysql zabbix-web-mysql zabbix-agent
```

- Если используется PostgreSQL, замените mysql на pgsql в команде.
- Если выполняется обновление прокси, замените server на proxy в команде.
- Если выполняется обновление агента 2, замените zabbix-agent на zabbix-agent2 zabbix-agent2-plugin-* в команде.

**Attention:**
Обновление агента 2 Zabbix с помощью команды dnf install zabbix-agent2 может привести к ошибке. Дополнительные сведения см. в разделе *Известные проблемы*.

Затем, чтобы обновить веб-интерфейс Zabbix с Apache и перезапустить Apache, выполните:

```
dnf install zabbix-apache-conf
systemctl restart httpd
```

6 Просмотрите параметры конфигурации компонента

Просмотрите соответствующие примечания к обновлению, чтобы проверить, нужны ли какие-либо изменения в параметрах конфигурации.

О новых необязательных параметрах см. страницу Что нового.

7. Запустите процессы Zabbix

Запустите обновлённые компоненты Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8. Очистите cookies и кэш в веб-браузере

После обновления вам, возможно, потребуется очистить cookies веб-браузера и кэш веб-браузера, чтобы веб-интерфейс Zabbix работал должным образом.

Обновление между минорными версиями

Возможно выполнить обновление между минорными версиями Zabbix 7.4.x (например, с 7.4.1 до 7.4.3).

Чтобы обновить все компоненты Zabbix, выполните:

```
dnf upgrade 'zabbix-*'
```

- Чтобы обновить только сервер Zabbix, замените 'zabbix-*' на 'zabbix-server-*' в команде.
- Чтобы обновить только прокси Zabbix, замените 'zabbix-*' на 'zabbix-proxy-*' в команде.
- Чтобы обновить только агент Zabbix, замените 'zabbix-*' на 'zabbix-agent-*' в команде.
- Чтобы обновить только агент Zabbix 2, замените 'zabbix-*' на 'zabbix-agent2-*' в команде.

### 2. Debian/Ubuntu

Обзор

В этом разделе приведены инструкции по обновлению Zabbix **7.2.x** до последней версии Zabbix **7.4.x** с использованием официальных пакетов Zabbix для Debian/Ubuntu.

**Warning:**
Перед обновлением ознакомьтесь с соответствующими примечаниями по обновлению и убедитесь, что ваша система соответствует требованиям для Zabbix 7.4.

**Note:**
Рекомендуется во время обновления использовать две параллельные SSH-сессии: одну для выполнения шагов обновления, а другую для мониторинга журналов сервера/прокси. Например, во второй сессии выполните `tail -f zabbix_server.log` или `tail -f zabbix_proxy.log`, чтобы в реальном времени просматривать последние записи журнала и возможные ошибки. Это может быть критически важно для рабочих экземпляров.

Инструкции по обновлению между минорными версиями Zabbix 7.4.x (например, с 7.4.1 до 7.4.3) см. в разделе Обновление между минорными версиями.

Процедура обновления

1. Остановите процессы Zabbix

Остановите Zabbix сервер, чтобы быть уверенным, что в базу данных не будет происходить запись новых данных.

```
systemctl stop zabbix-server
```

Если обновляете Zabbix прокси, агента или агента 2, то остановите и эти компоненты тоже.

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2. Сделайте резервную копию базы данных Zabbix

Сделайте резервную копию вашей существующей базы данных Zabbix, чтобы подстраховаться от возможных сбоев при обновлении (например, проблем с дисковым пространством, отключения электричества или непредвиденных проблем).

3. Сделайте резервную копию файлов конфигурации, PHP и бинарных файлов Zabbix

Сделайте резервную копию существующих файлов конфигурации, файлов PHP и бинарных файлов Zabbix.

Для файлов конфигурации выполните:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-backup/
```

Для файлов PHP и двоичных файлов Zabbix выполните:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Обновление пакета конфигурации репозитория

Перед продолжением обновления удалите текущий пакет репозитория Zabbix:

```
rm -Rf /etc/apt/sources.list.d/zabbix.list
```

Также может потребоваться вручную удалить все старые пакеты Zabbix из рабочего каталога (например, `rm zabbix-release_late` перед загрузкой нового, чтобы менеджер пакетов не использовал устаревшую версию в процессе обновления.

Затем установите последний пакет конфигурации репозитория, чтобы обеспечить совместимость с новейшими пакетами и включить все недавние исправления безопасности или ошибки.

На **Debian 12** выполните:

```
wget https://repo.zabbix.com/zabbix/7.4/release/debian/pool/main/z/zabbix-release/zabbix-release_latest+de
dpkg -i zabbix-release_latest+debian12_all.deb
```

**Note:**
Для более старых версий Debian замените ссылку выше на правильную из репозитория Zabbix. Однако учтите, что пакеты для этих версий могут не включать все компоненты Zabbix, и для обновления этих компонентов из пакетов стоит рассмотреть обновление ОС. Список включенных компонентов см. в разделе Пакеты Zabbix.

На **Ubuntu 24.04** выполните:

```
wget https://repo.zabbix.com/zabbix/7.4/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu24.04_all.deb
```

На **Ubuntu 22.04** выполните:

```
wget https://repo.zabbix.com/zabbix/7.4/release/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest+ub
dpkg -i zabbix-release_latest+ubuntu22.04_all.deb
```

**Note:**
Для более старых версий Ubuntu замените ссылку выше на правильную из репозитория Zabbix. Однако учтите, что пакеты для этих версий могут не включать все компоненты Zabbix, и для обновления этих компонентов из пакетов стоит рассмотреть обновление ОС. Список включенных компонентов см. в разделе Пакеты Zabbix.

Вы можете увидеть запрос, связанный с конфигурацией репозитория 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] ?
```

Введите Y (или I), чтобы установить версию конфигурации репозитория Zabbix от сопровождающего пакета.

Затем обновите сведения о репозитории:

```
apt update
```

5 Обновление компонентов Zabbix

Чтобы обновить компоненты Zabbix, выполните:

```
apt install --only-upgrade zabbix-server-mysql zabbix-frontend-php zabbix-agent
```

- Если используется PostgreSQL, замените `mysql` на `pgsql` в команде.
- Если обновляется прокси, замените `server` на `proxy` в команде.
- Если обновляется Zabbix agent 2, замените `zabbix-agent` на `zabbix-agent2 zabbix-agent2-plugin-*` в команде.

**Attention:**
Обновление Zabbix agent 2 с помощью команды `apt install zabbix-agent2` может привести к ошибке. Дополнительную информацию см. в разделе *Известные проблемы*.

Вы можете увидеть запрос, связанный с конфигурацией сервера Zabbix (или прокси):

```
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] ?
```

Введите вариант, который лучше всего подходит для вашей ситуации. Например, введите D, чтобы сравнить текущую и новую конфигурацию, а затем решите, нужно ли устанавливать версию, предоставленную сопровождающим пакета (Y или I).

Затем, чтобы обновить веб-интерфейс Zabbix с Apache и перезапустить Apache, выполните:

```
apt install zabbix-apache-conf
systemctl restart apache2
```

6 Просмотрите параметры конфигурации компонента

Ознакомьтесь с соответствующими примечаниями к обновлению, чтобы проверить, нужны ли какие-либо изменения в параметрах конфигурации.

О новых необязательных параметрах см. страницу Что нового.

7. Запустите процессы Zabbix

Запустите обновлённые компоненты Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8. Очистите cookies и кэш в веб-браузере

После обновления вам, возможно, потребуется очистить cookies веб-браузера и кэш веб-браузера, чтобы веб-интерфейс Zabbix работал должным образом.

Обновление между минорными версиями

Возможно обновление между минорными версиями Zabbix 7.4.x (например, с 7.4.1 до 7.4.3).

Сначала обновите информацию о репозитории:

```
apt update
```

Затем, чтобы обновить все компоненты Zabbix, выполните:

```
apt install --only-upgrade 'zabbix*'
```

- Чтобы обновить только сервер Zabbix, замените `'zabbix*'` на `'zabbix-server*'` в команде.
- Чтобы обновить только прокси Zabbix, замените `'zabbix*'` на `'zabbix-proxy*'` в команде.
- Чтобы обновить только агент Zabbix, замените `'zabbix*'` на `'zabbix-agent*'` в команде.
- Чтобы обновить только агент Zabbix 2, замените `'zabbix*'` на `'zabbix-agent2*'` в команде.

### 3. Обновление из контейнеров

Обзор

В этом разделе описаны шаги, необходимые для успешного обновления контейнеров Zabbix до версии **7.4**.x.

Отдельные наборы инструкций доступны для:

- обновления образа компонента Zabbix
- обновления файла Docker Compose

**Warning:**
Перед обновлением обязательно ознакомьтесь с соответствующими **примечаниями к обновлению**!

**Attention:**
Перед началом обновления убедитесь, что у пользователей есть необходимые права доступа к базе данных для выполнения обновления. <br><br> При обновлении с Zabbix 6.0 или более ранней версии во время обновления потребуется создать детерминированные триггеры. Если для MySQL/MariaDB включено бинарное журналирование, для этого требуются привилегии суперпользователя или установка переменной/параметра конфигурации `log_bin_trust_function_creators = 1`. См. Скрипты создания базы данных для инструкций по настройке этой переменной. <br><br> Обратите внимание, что при выполнении из консоли переменная будет установлена только временно и будет сброшена при перезапуске Docker. В этом случае оставьте службу SQL запущенной, остановите только службу `zabbix-server`, выполнив `docker compose down zabbix-server`, а затем `docker compose up -d zabbix-server`. <br><br> В качестве альтернативы вы можете задать эту переменную в файле конфигурации.

В зависимости от размера базы данных обновление базы данных до версии 7.4 может занять много времени.

Обновление образа компонента Zabbix

Эти инструкции позволяют обновить образы Docker, которые используются в качестве основы для контейнеров компонентов Zabbix.

В приведенных ниже шагах в качестве примера используется прокси Zabbix.
Замените `zabbix-proxy-sqlite3` на имя контейнера вашего компонента.

1. Проверьте все контейнеры, чтобы определить их имена и текущие версии образов:

```
docker ps -a

#### CONTAINER ID   IMAGE                                              (...) NAMES
#### 5a0c2e3d02b4   zabbix/zabbix-proxy-sqlite3:alpine-7.2-latest      (...) zabbix-proxy-sqlite3
```

2. Загрузите новую версию образа:

```
docker pull zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest
```

3. Остановите контейнер компонента Zabbix:

```
docker stop zabbix-proxy-sqlite3
```

4. Удалите контейнер компонента Zabbix:

**Attention:**
Перед продолжением убедитесь, что вы записали все флаги конфигурации и точки монтирования томов, использованные в исходной команде `docker run`, так как они понадобятся для повторного создания контейнера.

```
docker rm zabbix-proxy-sqlite3
```

5. Запустите новый контейнер, используя обновленный образ:

```
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-7.4-latest
```

**Note:**
Дополнительные сведения о запуске контейнеров Zabbix см. в разделе Docker (ручное развертывание).

6. Убедитесь, что обновление прошло успешно, проверив журналы контейнера:

```
docker logs -f zabbix-proxy-sqlite3
```

Обновление файла Docker Compose

Эти инструкции позволяют обновить файлы Docker Compose, используемые для развертывания компонентов Zabbix. Следуйте этим инструкциям, если вы установили Zabbix с помощью Docker Compose.

1. Проверьте все контейнеры, чтобы определить их имена и текущие версии образов:

```
docker ps -a

#### CONTAINER ID   IMAGE                                              (...) NAMES
#### d095f486deef   zabbix/zabbix-server-mysql:alpine-7.2-latest       (...) zabbix-docker-zabbix-server-1
#### 70627d833da0   zabbix/zabbix-web-nginx-mysql:alpine-7.2-latest    (...) zabbix-docker-zabbix-web-ngin
#### aaaa664cc185   zabbix/zabbix-server-mysql:alpine-7.2-latest       (...) zabbix-docker-server-db-init-1
#### 9ab3e77492ff   mysql:8.0-oracle                                   (...) zabbix-docker-mysql-server-1
```

2. Перейдите в ранее клонированный репозиторий Zabbix Docker, получите последние обновления и переключитесь на ветку 7.4:

**Attention:**
Если вы изменяли `.env`, `compose.yaml` или другие файлы в локальном репозитории, команда `git pull` может перезаписать их. Сделайте резервную копию всех локальных изменений перед выполнением этого шага.

```
cd zabbix-docker
git pull
git checkout 7.4
```

3. Разверните компоненты Zabbix, используя обновленный файл Docker Compose:

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

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

> **Note:**
> После того как все контейнеры будут пересозданы и запущены, возобновление сбора данных может занять 1—3 минуты. Дополнительные сведения о запуске контейнеров Zabbix см. в разделе Docker Compose.

4. Убедитесь, что обновление прошло успешно, проверив журналы контейнера (замените `zabbix-docker-zabbix-server-1` на имя контейнера вашего компонента):

```
docker logs -f zabbix-docker-zabbix-server-1
```

### 8. Известные проблемы

Смотрите также: Проблемы при компиляции.

Известные проблемы в 7.4.4

Не рекомендуется обновляться до этой версии из-за:

- внезапного скачка загрузки CPU, если вы используете плагин MySQL для Zabbix agent 2 (см. ZBX-27156)
- графика для элементов данных активного агента Zabbix, на котором отображается предупреждение ”Undefined array key” из-за ошибки неопределенного индекса (см. ZBX-27153)

Обновление

Настройки режима SQL для успешного обновления

Параметр `sql_mode` в MySQL/MariaDB должен быть выставлен в режим «STRICT_TRANS_TABLES». Если он отсутствует, обновление базы данных Zabbix завершится неудачей (смотрите также ZBX-19435 [en]).

Обновление с MariaDB 10.2.1 и старше

Обновление Zabbix может завершится с ошибкой, если таблицы базы данных были созданы с MariaDB версии 10.2.1 и старше, так как в этих версиях по умолчанию используется компактный формат строк. Такая проблема исправляется изменением формата строк на динамический формат (смотрите также ZBX-17690 [en]).

Шаблоны

Совместимость шаблонов в средах с двойным стеком (IPv4/IPv6)

В средах с двойным стеком (системы, настроенные на поддержку как IPv4, так и IPv6) имя хоста `localhost` обычно разрешается как в IPv4-, так и в IPv6-адреса. Из-за того, что многие операционные системы и DNS-резолверы по умолчанию отдают приоритет IPv6 перед IPv4, шаблоны Zabbix могут работать некорректно, если служба, за которой ведется мониторинг, настроена на прослушивание только IPv4.

Службы, не настроенные на прослушивание IPv6-адресов, могут стать недоступными, что приведет к сбоям мониторинга. Пользователи могут корректно настроить доступ для IPv4, но при этом все равно столкнуться с проблемами подключения из-за поведения по умолчанию, при котором приоритет отдается IPv6.

Обходной путь заключается в том, чтобы убедиться, что службы (Nginx, Apache, PostgreSQL и т. д.) настроены на прослушивание как IPv4-, так и IPv6-адресов, а сервер/агент Zabbix имеет доступ через IPv6. Кроме того, в шаблонах и конфигурациях Zabbix явно используйте `localhost` вместо `127.0.0.1`, чтобы обеспечить совместимость как с IPv4, так и с IPv6.

**Например**, при мониторинге PostgreSQL с помощью шаблона PostgreSQL by Zabbix agent 2 может потребоваться изменить файл `pg_hba.conf`, чтобы разрешить подключения для пользователя `zbx_monitor`. Если в среде с двойным стеком приоритет отдается IPv6 (система разрешает `localhost` в `::1`) и вы настраиваете `localhost`, но добавляете только

запись для IPv4 (127.0.0.1/32), подключение завершится неудачей, поскольку отсутствует соответствующая запись для IPv6.

Следующий пример файла pg_hba.conf обеспечивает возможность подключения пользователя zbx_monitor к любой базе данных с локальной машины с использованием как IPv4-, так и IPv6-адресов и разных методов аутентификации:

```
### 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
```

При необходимости вы также можете напрямую использовать IPv4-адрес (127.0.0.1) при настройке макроса шаблона PostgreSQL by Zabbix agent 2 для строки подключения.

Случайная установка пакетов Zabbix из EPEL

Если репозиторий EPEL установлен и активирован, установка пакетов Zabbix может привести к загрузке версий EPEL вместо официальных пакетов Zabbix. Чтобы решить эту проблему:

1. Удалите все пакеты Zabbix, установленные из EPEL:

```
dnf remove zabbix-server-mysql
```

2. Исключите пакеты Zabbix из EPEL, добавив следующую строку в файл /etc/yum.repos.d/epel.repo:

```
[epel]
...
excludepkgs=zabbix*
```

3. Переустановите официальный пакет сервера Zabbix:

```
dnf install zabbix-server-mysql
```

Во время установки официальные пакеты Zabbix включают слово release в строку версии (например, 7.0.0-release1.el8), что отличает их от пакетов EPEL.

Пакеты Zabbix для RHEL в средах Red Hat UBI

При установке Zabbix из пакетов Red Hat Enterprise Linux в средах Red Hat Universal Base Image убедитесь, что есть доступ к необходимым репозиториям и зависимостям. Пакеты Zabbix зависят от библиотек libOpenIPMI.so и libOpenIPMIposix.so, которые не предоставляются ни одним пакетом в репозиториях менеджера пакетов по умолчанию, включенных в системах UBI, и это приведет к ошибкам установки.

Библиотеки libOpenIPMI.so и libOpenIPMIposix.so доступны в пакете OpenIPMI-libs, который предоставляется репозиторием redhat-#-for-<arch>-appstream-rpms. Доступ к этому репозиторию определяется подписками, которые в случае сред UBI передаются путем подключения каталогов конфигурации репозиториев и секретов хоста RHEL в пространство имен файловой системы контейнера.

Дополнительные сведения см. в ZBX-24291.

Истекший ключ подписи для пакетов RHEL

При обновлении Zabbix на Red Hat Enterprise Linux или его производных вы можете столкнуться с проблемой истекшего ключа подписи для пакетов в репозитории Zabbix. Когда срок действия ключа подписи истекает, попытки проверить подписи пакетов приводят к ошибке, указывающей, что сертификат или ключ больше недействителен. Например:

```
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
```

Чтобы устранить такую проблему, вручную переустановите последний пакет zabbix-release для вашей конкретной версии RHEL (замените ссылку ниже на правильную из репозитория Zabbix).

Например, для **RHEL 9** выполните:

```
rpm -Uvh https://repo.zabbix.com/zabbix/7.4/release/rhel/9/noarch/zabbix-release-latest.el9.noarch.rpm
```

Затем обновите сведения о репозитории:

```
dnf update
```

Дополнительную информацию см. в ZBX-24761.

Timescale DB: высокое потребление памяти при большом количестве разделов

Версии PostgreSQL 9.6-12 используют слишком много памяти при обновлении таблиц с большим количеством разделов. Эта проблема проявляется, когда Zabbix обновляет тренды в системах с TimescaleDB, если тренды разбиты на относительно небольшие фрагменты (например, по 1 дню). Это приводит к тому, что в таблицах трендов при настройках обслуживания по умолчанию присутствуют сотни фрагментов — в такой ситуации PostgreSQL, скорее всего, исчерпает память.

Проблема была устранена начиная с Zabbix 5.0.1 для новых установок с TimescaleDB, но если TimescaleDB был настроен вместе с Zabbix ранее, см. ZBX-16347 для заметок по миграции.

Timescale DB 2.5.0: политика сжатия может завершаться сбоем для таблиц, содержащих целые числа

Эта проблема проявляется при использовании TimescaleDB 2.5.0/2.5.1. Она была устранена начиная с TimescaleDB 2.5.2.

Для получения дополнительной информации см. TimescaleDB Issue #3773.

TLS-подключение к базе данных с MariaDB

TLS-подключение к базе данных не поддерживается с опцией 'verify_ca' для параметра DBTLSConnect, если используется MariaDB.

Возможные взаимные блокировки с MySQL/MariaDB

При работе под высокой нагрузкой и использовании более одного процесса LLD worker возможно возникновение взаимных блокировок (deadlocks), вызванных ошибкой InnoDB, относящейся к стратегии блокировок строк (см. upstream bug [en]). Ошибка была исправлена в MySQL, начиная с версии 8.0.29, но не в MariaDB. Для более подробной информации, обратитесь к ZBX-21506 [en].

Глобальная корреляция событий

События могут не коррелироваться корректно, если временной интервал между первым и вторым событием очень мал, то есть составляет полсекунды и меньше.

Диапазон типа данных числовой (с плавающей точкой) в PostgreSQL 11 и более старых версиях

PostgreSQL версии 11 и старше поддерживают значения с плавающей точкой только в диапазоне приблизительно от -1.34E-154 до 1.34E+154.

NetBSD 8.0 и новее

Различные процессы Zabbix могут случайным образом аварийно завершать работу при запуске на NetBSD версий 8.X и 9.X. Такое поведение связано со слишком малым размером стека по умолчанию (4МБ), который необходимо увеличить, выполнив:

```
ulimit -s 10240
```

Для получения более подробной информации, пожалуйста, обратитесь к соответствующему отчёту о проблеме: ZBX-18275 [en].

Ограничения регулярных выражений в Zabbix агенте 2

Агент Zabbix 2 не поддерживает просмотр вперед и назад в регулярных выражениях из-за ограничений стандартной библиотеки регулярных выражений Go.

IPMI проверки

IPMI проверки не будут работать со стандартным пакетом библиотеки OpenIPMI на Debian до версии 9 (stretch) и Ubuntu до 16.04 (xenial). Чтобы исправить проблему, пересоберите OpenIPMI библиотеку с включённым OpenSSL, как обсуждалось в ZBX-6139 [en].

IPMI — недоверенные узлы сети могут вызвать сбой OpenIPMI

В библиотеке OpenIPMI, используемой Zabbix для опроса данных IPMI, обнаружена ошибка, которую можно вызвать специально сформированными ответами от недоверенного устройства.
Недоверенное устройство IPMI может отправить специально сформированные данные, из-за которых библиотека OpenIPMI завершится сбоем, что, в свою очередь, может привести к завершению процесса сервера Zabbix, выполняющего опрос IPMI.

SSH проверки

- Некоторые дистрибутивы Linux, такие как Debian и Ubuntu, не поддерживают зашифрованные закрытые (private) ключи (с парольной фразой), если библиотека libssh2 установлена из пакетов. Пожалуйста, обратитесь к ZBX-4850 [en] для получения более подробных сведений.
- При использовании libssh 0.9.x на некоторых дистрибутивах Linux с OpenSSH 8 проверки SSH могут иногда сообщать: «Cannot read data from SSH server». Такое поведение вызвано проблемой [en](более подробный отчёт [en]) libssh. Ожидается, что эта ошибка исправлена в стабильном выпуске libssh 0.9.5. Смотрите также ZBX-17756 [en] для получения подробной информации.
- Использование вертикальной черты "|" в SSH скрипте может привести к ошибке «Cannot read data from SSH server». В этом случае рекомендуется обновить версию библиотеки libssh. Смотрите также ZBX-21337 [en] для получения подробной информации.

Проверки ODBC

- Драйвер MySQL unixODBC не следует использовать с сервером Zabbix или прокси Zabbix, скомпилированными с библиотекой MariaDB connector, и наоборот; если возможно, также лучше избегать использования того же connector, что и драйвер, из-за ошибки upstream. Рекомендуемая настройка:

  PostgreSQL, SQLite или Oracle connector → драйвер MariaDB или MySQL unixODBC<br>MariaDB connector → драйвер MariaDB unixODBC<br>MySQL connector → драйвер MySQL unixODBC

  См. ZBX-7665 для получения дополнительной информации и доступных обходных решений.

- XML-данные, запрашиваемые из Microsoft SQL Server, могут обрезаться различными способами в системах Linux и UNIX.

- Было замечено, что использование проверок ODBC для мониторинга баз данных Oracle с применением различных версий Oracle Instant Client для Linux приводит к аварийному завершению работы сервера Zabbix.<br> См. также: ZBX-18402, ZBX-20803.

- При использовании драйвера FreeTDS UnixODBC необходимо добавить оператор 'SET NOCOUNT ON' в начало SQL-запроса (например, `SET NOCOUNT ON DECLARE @strsql NVARCHAR(max) SET @strsql = ....`). В противном случае элемент данных типа database monitor в Zabbix не сможет получить информацию и вернет ошибку "SQL query returned empty result".<br> См. ZBX-19917 для получения дополнительной информации.

Некорректный параметр метода запроса в элементах данных

Параметр метода запроса, используемый только в HTTP проверках, может быть некорректным образом задан значением '1', значением не по умолчанию для всех элементов данных, в результате обновления с версии Zabbix pre-4.0. Для получения более подробных сведений по устранению данной ситуации смотрите ZBX-19308 [en].

Веб-мониторинг и HTTP агент

Zabbix сервер имеет утечку памяти на некоторых дистрибутивах Linux по причине ошибки в библиотеке [en] при включённой опции «Проверка SSL узла» в веб-сценариях или HTTP агенте. Пожалуйста, обратитесь к ZBX-10486 [en] для получения более подробной информации и возможных вариантов решения.

Простые проверки

Имеется проблема в утилите **fping** версий до v3.10, которые некорректно обрабатывают дубликаты ответных echo пакетов. Такое поведение может вызывать непредвиденные результаты в элементах данных `icmpping`, `icmppingloss`, `icmppingsec`. Рекомендуется использовать последнюю версию **fping**. Пожалуйста, обратитесь к ZBX-11726 [en] для получения более подробной информации.

Ошибки при выполнении fping в контейнерах без root-доступа

Когда контейнеры работают в режиме без root-доступа или в среде со специальными ограничениями, вы можете столкнуться с ошибками, связанными с работой fping при выполнении проверок ICMP, такими как: «`fping: Operation not permitted`», — или потерей всех пакетов ко всем ресурсам.

Для исправления этой проблемы добавьте `--cap-add=net_raw` к командам «docker run» или «podman run».

В дополнение к этому, выполнение fping в средах без root-доступа может потребовать модификации sysctl, то есть:

```
sudo sysctl -w "net.ipv4.ping_group_range=0 1995"
```

где «1995» — это GID zabbix. Для дополнительной информации обратитесь к ZBX-22833 [en].

SNMP проверки

При использовании операционной системы OpenBSD, проблема «использование памяти после её освобождения» в библиотеке Net-SNMP вплоть до версии 5.7.3 может привести к аварийной остановке сервера Zabbix, если в файле конфигурации сервера Zabbix указан параметр SourceIP. Как временное решение, пожалуйста, не задавайте параметр

SourceIP. Эта проблема также применима и к Linux, но она не приводит к остановке работы сервера Zabbix. К пакету net-snmp применен локальный патч на OpenBSD и будет выпущен с версией OpenBSD 6.3.

Резкие скачки в SNMP данных

Наблюдались резкие скачки в SNMP данных, которые могут быть связаны с определёнными физическими факторами, такими как скачки напряжения в сети. Обратитесь к ZBX-14318 [en] для получения более подробной информации.

SNMP-трапы

Пакет ”net-snmp-perl”, необходимый для SNMP-трапов, был удален в RHEL 8.0-8.2; снова добавлен в RHEL 8.3.

Поэтому, если вы используете RHEL 8.0-8.2, лучшее решение — обновиться до RHEL 8.3.

Дополнительную информацию также см. в ZBX-17192.

Сбой в процессе alerter в RHEL 7

В RHEL 7 обнаружены случаи сбоя в процессах alerter сервера Zabbix. Пожалуйста, обратитесь к ZBX-10461 [en] для получения более подробной информации.

Обновление Zabbix агента 2 (6.0.5 или более старого)

При обновлении Zabbix агента 2 (версии 6.0.5 или старше) из пакетов может возникнуть ошибка конфликта файлов, связанных с плагином. Чтобы исправить ошибку, сделайте резервную копию конфигурации агента 2 (если необходимо), деинсталлируйте агент 2 и установите заново.

На системах на базе RHEL выполните:

```
dnf remove zabbix-agent2
dnf install zabbix-agent2
```

На системах на базе Debian выполните:

```
apt remove zabbix-agent2
apt install zabbix-agent2
```

Для более подробной информации обратитесь к ZBX-23250 [en].

Переключение локалей в веб-интерфейсе

Было замечено, что локали в веб-интерфейсе могут переключаться без какой-либо явной логики, то есть некоторые страницы (или части страниц) отображаются на одном языке, тогда как другие страницы (или части страниц) — на другом. Обычно такая проблема может возникнуть, когда имеется несколько пользователей, некоторые из которых используют одну локаль, в то время как другие используют другую локаль.

Известный вариант решения такой проблемы — отключение многопоточности в PHP и Apache.

Эта проблема связана с тем, как локали работают в PHP [en]: информация о локали сохраняется по каждому процессу, а не по потоку. Поэтому в многопоточной среде, где несколько проектов выполняются одним и тем же процессом Apache, имеется вероятность, что локаль меняется в другом потоке, что меняет способ обработки данных в потоке Zabbix.

Для получения более подробной информации смотрите соответствующие отчёты о проблеме:

- ZBX-10911 [en] (Проблема с переключением локалей в веб-интерфейсе)
- ZBX-16297 [en] (Проблема с обработкой чисел в графиках с использованием функции `bcdiv` из функций BC Math)

Графики

Проблемы с графиками (классическими)

Если у вас возникли проблемы с классическими графиками, рекомендуется обновить библиотеку GD (libgd) до версии 2.3.3-13 или выше, а PHP — до версии 8.0.19, 8.1.33, 8.2.29, 8.3.25, 8.4.12 или выше.

Летнее время

Переходы на летнее время (DST) приводят к нарушению отображения подписей к оси X (дублирование даты, отсутствие даты и т.п.).

Агрегирование в виде суммы

При использовании агрегирования в виде суммы в графике за период менее одного часа, графики отображают некорректные (размноженные) значения, когда данные поступают из динамики изменений.

Наложение текста

Для некоторых языков веб-интерфейса (например, японского), местные шрифты могут приводить к наложению текста в легенде графика. Чтобы избежать этого, используйте версию 2.3.0 (или более позднюю) расширения PHP GD.

Мониторинг файлов журналов

Элементы данных `log[]` и `logrt[]` многократно перечитывают файл журнала с самого начала, если файловая система заполнена на 100% и файл журнала продолжает заполняться (обратитесь к ZBX-10884 [en] для получения более подробной информации).

Медленные запросы MySQL

Zabbix сервер генерирует медленные запросы `SELECT` в случае несуществующих значений у элементов данных. Это обусловлено известной проблемой [en] в MySQL версий 5.6/5.7 (для её обсуждения обратитесь к ZBX-10652 [en]), а также, в некоторых случаях, может происходить и в более поздних версиях MySQL. Чтобы обойти эту проблему, отключите параметр `index_condition_pushdown` [en] или `prefer_ordering_index` [en] оптимизатора в MySQL. Обратите внимание, однако, что такой обходной путь может исправить не все проблемы, относящиеся к медленным запросам.

Постоянные настройки фильтра из ссылок

При открытии ссылки на страницу веб-интерфейса Zabbix, содержащую настройки фильтра, включая выбор времени, фильтр автоматически сохраняется в базе данных для пользователя, заменяя ранее сохраненные настройки фильтра и/или выбора времени для этой страницы. Эти настройки остаются активными, пока пользователь не обновит или не сбросит их вручную.

Проблема с IPv6 адресами в SNMPv3 трапах

По причине проблемы в net-snmp, в SNMP трапах IPv6 адреса могут некорректно отображаться при использовании SNMPv3. Для получения более подробной информации и возможных вариантах решения смотрите ZBX-14541 [en].

Урезание длинных IP адресов IPv6 в информации о неуспешном входе в систему

Сообщение об ошибочном входе в систему отобразит только первые 39 символов сохранённого IP адреса, так как это ограничение длины поля базы данных. Это означает, что IP адреса IPv6 длиннее 39 символов будут отображаться не полностью.

Проверки Zabbix агент на Windows

Несуществующие записи DNS в параметре `Server` в файле конфигурации Zabbix агента (zabbix_agentd.conf) могут повысить время ответа Zabbix агента на Windows. Такое случается по причине того, что демон Windows DNS кэширования не кэширует отрицательные ответы по IPv4 адресам. Однако, для IPv6 адресов отрицательные ответы кэшируются; таким образом, возможным вариантом решения будет отключение IPv4 на хосте.

Экспорт/импорт YAML

Существуют некоторые известные проблемы с YAML экспортом/импортом:

- Сообщения об ошибках не подлежат переводу;
- Допустимый JSON с расширением файла .yaml иногда не удается импортировать;
- Не заключенные в кавычки человекочитаемые даты автоматически преобразуются в Unix-метки времени.

Мастер установки в SUSE с NGINX и php-fpm

Мастер установки веб-интерфейса не в состоянии сохранить файл конфигурации в SUSE с NGINX + php-fpm. Эта проблема вызвана настройкой в /usr/lib/systemd/system/php-fpm.service юните, который не позволяет Zabbix выполнять запись в /etc. (введено в PHP 7.4 [en]).

Доступно два варианта решения:

- Задать опции ProtectSystem [en] значение «true» вместо «full» в php-fpm юните systemd.
- Вручную сохранить файл /etc/zabbix/web/zabbix.conf.php.

Пересылка заголовка Authorization

В некоторых случаях Apache или NGINX могут препятствовать тому, чтобы заголовок Authorization в API-запросах доходил до Zabbix. Это может вызывать проблемы с аутентификацией при использовании Zabbix API или служб единого входа (SSO), таких как SAML с Okta.

Чтобы устранить это, обновите конфигурацию вашего веб-сервера.

Для **Apache**, если вы используете его как обратный прокси (схема без CGI), добавьте следующую директиву в `/etc/httpd/conf/httpd.conf` (в системах на базе RHEL) или `/etc/apache2/apache2.conf` (в Debian/Ubuntu):

```
SetEnvIfNoCase ^Authorization$ "(.+)" HTTP_AUTHORIZATION=$1
```

Если Apache напрямую выполняет сценарии для обработки запросов (например, с использованием mod_cgi), вместо этого добавьте следующую директиву:

```
CGIPassAuth On
```

В отличие от этого, **NGINX** обрабатывает заголовок Authorization автоматически. Однако, если NGINX работает как обратный прокси, вы можете явно пересылать заголовок Authorization, добавив следующие директивы в `/etc/nginx/nginx.conf` (для расположения вашего веб-интерфейса Zabbix):

```
...
location / {
...
    proxy_set_header Authorization $http_authorization;
    proxy_pass http://backend_server;
...
}
```

После обновления конфигурации перезапустите ваш веб-сервер.

Подробнее см.:

- ZBX-22952
- Apache 2.4 + PHP-FPM and Authorization headers
- директивы SetEnvIfNoCase и CGIPassAuth
- NGINX Reverse Proxy

Chromium для веб-сервиса Zabbix в Ubuntu 20

Хотя в большинстве случаев веб-сервис Zabbix может работать с Chromium, в Ubuntu 20.04 использование Chromium приводит к следующей ошибке:

```
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
```

Эта ошибка возникает из-за того, что в качестве домашней директории пользователя «zabbix» используется `/var/lib/zabbix`.

Специальные коды ошибок MySQL

Когда Zabbix обнаруживает, что внутренняя база данных недоступна, он отправляет уведомление и продолжает попытки подключения. Для определённых движков баз данных распознаются определённые коды ошибок. В MySQL эти распознаваемые коды ошибок включают:

- 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

Кроме того, если Zabbix используется с MySQL инсталляцией в Azure, в журналах Zabbix может появиться расплывчатое сообщение об ошибке *[9002] Some errors occurred*. Это сообщение отправляется Zabbix серверу или прокси со стороны базы данных. Чтобы определить причину ошибки, обратитесь к журналам Azure.

Ошибочные регулярные выражения после переключения на PCRE2

В Zabbix 6.0 была добавлена поддержка PCRE2. Хотя PCRE по-прежнему поддерживается, установочные пакеты Zabbix для RHEL 7 и новее, SLES (всех версий), Debian 9 и новее, Ubuntu 16.04 и новее были обновлены и теперь используют PCRE2. Несмотря на множество преимуществ, переход на PCRE2 может привести к тому, что существующие шаблоны регулярных выражений PCRE станут ошибочными или изменят свое поведение. В частности, это изменение затрагивает шаблон *^[\w-\.]*. Чтобы сделать это регулярное выражение снова корректным, не затрагивая семантику, измените выражение на *^[-\w\.]* . Такая проблема происходит из-за того, что PCRE2 рассматривает знак минуса в качестве разделителя, создавая диапазон внутри класса символов.

Ошибка виджета Geomap

Карты в виджете Geomap могут загружаться некорректно, если вы обновились с более старой версии Zabbix с NGINX и не переключились на новый файл конфигурации NGINX во время обновления.

Чтобы исправить проблему, вы можете удалить старый файл конфигурации, использовать файл конфигурации из пакета текущей версии и перенастроить его, как описано в инструкции по загрузке в разделе *e. Configure PHP for Zabbix frontend*.

Либо вы можете вручную отредактировать существующий файл конфигурации NGINX (обычно */etc/zabbix/nginx.conf*). Для этого откройте файл и найдите следующий блок:

```
location ~ /(api\/|conf[^\.]|include|locale|vendor) {
        deny            all;
        return          404;
}
```

Затем замените этот блок на:

```
location ~ /(api\/|conf[^\.]|include|locale) {
        deny            all;
        return          404;
}

location /vendor {
        deny            all;
        return          404;
}
```

Препроцессинг — глобальные переменные небезопасны

JavaScript в препроцессинге выполняется для каждого запроса, но присваивания не объявленным идентификаторам (например, `secret = value`) создают неявные глобальные переменные, которые могут сохраняться дольше текущего выполнения. Хранение конфиденциальных данных (токенов, паролей и т. д.) в неявных глобальных переменных повышает риск случайного раскрытия или повторного использования при последующих запусках препроцессинга либо другими интеграциями, выполняющимися в той же среде.

Не полагайтесь на неявные глобальные переменные. Всегда объявляйте переменные с помощью `var` или `const` и не прикрепляйте секреты к глобальным объектам (например, `globalThis` или `window`). Поддерживаемого способа переопределить встроенные глобальные объекты из препроцессинга не существует.

Безопасный пример:

```
var apiToken = payload.token;
var count = 1;
return JSON.stringify({ token: apiToken, calls: count });
```

Группы процессоров в Windows

В документации Microsoft указано, что системы с менее чем 64 логическими процессорами всегда имеют одну группу процессоров, Group 0. Однако пользователи Zabbix сообщали о редкой ошибке ZBX-20260, когда на системах с 64 или менее логическими процессорами было две группы процессоров. Это приводило к тому, что счетчики производительности "\Processor(n)" были доступны только для одной из двух групп процессоров. Фактическая первопричина этой ошибки неизвестна. Однако похожий случай был описан на stackoverflow.com, и там первопричина заключалась во взаимодействии между BIOS и Windows.

Ограничения фильтрации с utf8mb4-сопоставлениями

Фильтры (например, в *Сбор данных* > *Обслуживание*) могут работать некорректно при применении к сущностям, содержащим определенные символы Unicode (например, 𝓒, 𝔠). Эта проблема возникает из-за того, как сопоставление utf8mb4_bin по умолчанию для баз данных MySQL или MariaDB обрабатывает сортировку и сравнение символов Unicode.

Чтобы обойти это ограничение, пользователи могут изменить сопоставление столбцов базы данных на альтернативные варианты, такие как utf8mb4_0900_bin, utf8mb4_0900_ai_ci или utf8mb4_unicode_520_ci. Однако следует учитывать, что изменение сопоставления может привести к неожиданному поведению при обработке пустых пробелов, а также при сортировке и фильтрации других символов.

Дополнительную информацию об изменении сопоставлений см. в документации MySQL или документации MariaDB. Подробности о различиях между сопоставлениями см. в разделе Unicode Character Sets в документации MySQL.

Доступ к элементам интерфейса с MariaDB 10.5.1-10.5.9

При доступе к веб-интерфейсу Zabbix с ролью, отличной от Super Admin, может появиться сообщение: "System error occurred. Please contact Zabbix administrator.". Эта проблема затрагивает установки, использующие версии MariaDB с 10.5.1 по 10.5.9.

Чтобы избежать этой проблемы, обновите MariaDB до версии новее 10.5.9. Подробнее см. ZBX-25746.

Профилирование чрезмерного использования памяти с tcmalloc

Если вы подозреваете, что ваша установка Zabbix использует слишком много памяти, можно воспользоваться функцией профилирования памяти tcmalloc, чтобы исследовать потребление памяти сервером/прокси Zabbix.

1. При установке Zabbix из исходных кодов задайте дополнительные флаги:

```
export CFLAGS="-std=gnu99 -g -O0"
```

Флаг -std=gnu99 требуется для сборки сервера Zabbix, прокси Zabbix или агента Zabbix. Флаг -g добавляет дополнительную отладочную информацию, а -O0 отключает оптимизации, которые могут мешать профилированию tcmalloc.

2. Перед запуском сервера Zabbix задайте следующие переменные окружения. Эти переменные сообщают tcmalloc, как отслеживать и сообщать об использовании памяти:

```
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. Запустите дамп профиля, отправив сигнал 5 целевому процессу. Замените 1234 на фактический идентификатор процесса (PID):

```
kill -5 1234
```

4. Выведите созданный профиль:

```
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
```

В этом примере zbx_malloc2 отвечает почти за все выделения памяти.

См. также:

- ZBX-25050 и ZBX-25584 — связанные отчеты о проблеме.
- GCC Option Summary — о параметрах компиляции (-std=gnu99, -g, -O0 и т. д.).
- Gperftools Heap Profiler — документация по переменным окружения для профилирования tcmalloc.

MySQL 8.0 Group Replication в режиме multi-primary

При использовании MySQL 8.0 Group Replication в режиме multi-primary во время выполнения фиксации транзакции может возникать ошибка, похожая на следующую:

```
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
```

По-видимому, эта ошибка вызывается проблемами с операциями отката, связанными с ограничениями внешнего ключа.

См. также:

- ZBX-26060 — связанный отчет о проблеме.
- MySQL Bug #96758 "Rollbacks with Foreign Keys on single node" — соответствующая проблема в upstream.

### 1. Проблемы при компиляции

Это известные проблемы, связанные с компиляцией Zabbix из исходных кодов. Во всех остальных случаях смотрите страницу Известные проблемы.

Библиотека в нестандартном месте

Zabbix позволяет указать библиотеку, расположенную в нестандартном месте. В приведённом ниже примере Zabbix запустит `curl-config` из указанного нестандартного местоположения и использует его вывод для определения правильной используемой библиотеки libcurl.

```
$ ./configure --enable-server --with-mysql --with-libcurl=/usr/local/bin/curl-config
```

Это будет работать, если это единственная библиотека libcurl, установленная в системе, но может не сработать, если имеется другая библиотека libcurl, установленная в стандартном месте (например, с помощью менеджера пакетов). Так бывает, когда вам нужна более новая версия библиотеки для Zabbix и более старая для остальных приложений.

Поэтому указание компонента в нестандартном месте не всегда будет работать, если тот же компонент существует и в стандартном месте.

Например, если вы используете более новую версию библиотеки libcurl, установленную в `/usr/local`, но при этом установлен ещё пакет libcurl, Zabbix может взять неправильную библиотеку, и компиляция завершится неудачно:

```
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
```

Здесь функция `curl_easy_header()` недоступна в старой версии `/usr/lib/x86_64-linux-gnu/libcurl.so`, но доступна в более новой `/usr/local/lib/libcurl.so`.

Проблема заключается в порядке флагов линкера, и одно из решений — указать полный путь к библиотеке в переменной LDFLAGS:

```
$ LDFLAGS="-Wl,--no-as-needed /usr/local/lib/libcurl.so" ./configure --enable-server --with-mysql --with-l
```

Обратите внимание на параметр `-Wl,--no-as-needed`, который может потребоваться в некоторых системах (см. также: параметры связывания по умолчанию в системах на базе Debian [en]).

Слишком маленький размер стека на некоторых системах

Если Zabbix аварийно завершает работу или зависает из-за переполнения стека, вы можете увеличить размер стека для каждого потока с помощью параметра `--with-stacksize` при настройке исходных кодов. Эта проблема может возникать на системах с низкими значениями ограничения размера стека потоков по умолчанию, особенно во время предобработки, когда создается несколько потоков.

Следующий пример задает размер стека 512 КБ на поток:

```
./configure --enable-server --with-mysql --with-stacksize=512
```

Вы можете проверить ограничения размера стека потоков в системе во время выполнения с помощью команды `ulimit -s` в системах на базе Linux.

### 2. Проблемы обновления, связанные с экранированием

Превышено ограничение на количество символов для параметров функции после обновления

В Zabbix 7.0 добавлено правильное экранирование обратных косых черт в строковых параметрах функций истории.

Поскольку при обновлении с версий Zabbix до 7.0 добавляются дополнительные обратные косые черты, это приводит к увеличению длины параметров, что может привести к поломке функций триггера, если длина параметра превышает максимальный размер данных в 255 символов.

Чтобы проиллюстрировать, какие проблемы могут возникнуть при обновлении базы данных, каково их влияние, как их распознать и исправить, давайте рассмотрим пример настройки с тремя элементами данных и двумя триггерами.

Настройка

1. Версия Zabbix 6.0.
2. Файл `/tmp/ONE`, содержащий 114 символов обратной косой черты и символ «a» в конце:

```
\\\
```

3. Элемент данных `vfs.file.contents[/tmp/ONE]` для чтения /tmp/ONE.
4. Два вычисляемых элемента данных:

**Calc_228** — в его параметре содержится 228 обратных косых черт (в том числе экранирующих), так что они **соответствуют** 114 символам обратной косой черты из файла /tmp/ONE:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\
```

**Trig_228**:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\
```

**Изменения в 7.4.8** В этой минорной версии нет изменений шаблонов.

**Изменения в 7.4.7** Обновленные шаблоны

- Ciena 3906 by SNMP был обновлен и теперь включает элементы данных для файловой системы и загрузки CPU.
- Vyatta Virtual Router by SNMP был обновлен: добавлены новые элементы данных OID и улучшения панели мониторинга.

**Изменения в 7.4.6** Новые шаблоны

- Ribbon SBC Edge by HTTP, шаблон для мониторинга устройств Ribbon SBC Edge (ранее SWe Lite) по HTTP.
- Ribbon SBC SWe Core by HTTP, шаблон для мониторинга устройств SBC SWe Core по HTTP, и Ribbon SBC SWe CE by HTTP, шаблон для мониторинга экземпляров Ribbon SBC SWe Call Engine (CE) по HTTP.
- VeloCloud SD-WAN Edge by HTTP, шаблон для мониторинга устройств VeloCloud SD-WAN Edge по HTTP.

Обновленные шаблоны

- MySQL by Zabbix agent, MySQL by Zabbix agent 2, MySQL by Zabbix agent active, MySQL by Zabbix agent 2 active и MySQL by ODBC были обновлены для поддержки как `SHOW SLAVE STATUS` (старый синтаксис), так и `SHOW REPLICA STATUS` (новый синтаксис).
- VeloCloud SD-WAN by HTTP, ранее *VMWare SD-WAN VeloCloud by HTTP*, был переименован и обновлен в соответствии с последней платформой VeloCloud SD-WAN.

**Изменения в 7.4.4** Новые шаблоны

- Stormshield SNS by SNMP, шаблон для мониторинга устройств Stormshield Network Security (SNS) через SNMP.

**Изменения в 7.4.3** Новые шаблоны

- Aruba CX 8300s by SNMP, шаблон, обеспечивающий мониторинг серии коммутаторов Aruba CX 8300 на основе SNMP.
- Ciena 3906 by SNMP, шаблон для мониторинга устройств Ciena 3906.
- Vyatta Virtual Router by SNMP, шаблон для мониторинга виртуального маршрутизатора Vyatta 1908e.

Обновленные шаблоны

- Huawei OceanStor Dorado by SNMP был обновлен с добавлением сведений о покрытии устройств.
- Наборы шаблонов Zabbix server health и Zabbix proxy health были обновлены за счет расширенных возможностей мониторинга пропускной способности препроцессора.

**Изменения в 7.4.2** Новые шаблоны

- Набор шаблонов AWS by HTTP был дополнен шаблоном *AWS Backup Vault by HTTP*.

**Изменения в 7.4.1** Новые шаблоны

- Cisco Secure Firewall Threat Defense by HTTP, шаблон, предоставляющий возможности мониторинга устройств Cisco Secure Firewall Threat Defense с использованием REST API.

Обновленные шаблоны

- Шаблон *VMware Guest* в наборах шаблонов VMware и VMware FQDN был обновлен: добавлен новый элемент данных *Boot time*. Теперь этот элемент данных заменяет *Uptime of guest OS* в триггере *VM has been restarted*, чтобы избежать ложных срабатываний после миграций виртуальных машин, таких как vMotion.

## 10 Примечания к обновлению до 7.4.0

Эти примечания предназначены для обновления с Zabbix 7.2.x до Zabbix 7.4.0.

Все примечания сгруппированы в:

- **Критические изменения** - изменения, которые могут нарушить работу существующих установок, и другая важная информация, связанная с процессом обновления
- **Другое** - вся остальная информация, описывающая изменения в функциональности Zabbix

См. также:

- Процедура обновления - вся необходимая информация об обновлении с версий до Zabbix 7.4.0;
- Обновление кластера HA - инструкции по обновлению серверов в кластере **высокой доступности** (HA).

Процесс обновления

Чтобы успешно завершить обновление сервера Zabbix на MySQL/MariaDB, может потребоваться установить `GLOBAL log_bin_trust_function_creators = 1` в MySQL, если включено двоичное журналирование, отсутствуют привилегии суперпользователя и `log_bin_trust_function_creators = 1` не задан в файле конфигурации MySQL.

Чтобы задать переменную с помощью консоли MySQL, выполните:

```
mysql> SET GLOBAL log_bin_trust_function_creators = 1;
```

После успешного завершения обновления эту опцию можно отключить:

```
mysql> SET GLOBAL log_bin_trust_function_creators = 0;
```

**Критические изменения** Библиотека PCRE удалена

Библиотека PCRE (или PCRE1) была удалена. Теперь Zabbix собирается с PCRE2.

Управление собственными медиа пользователей

По умолчанию всем пользователям теперь разрешено управлять своими собственными медиа.

Права пользователя на изменение сведений о своих медиа могут быть предоставлены (или отозваны) в зависимости от его роли пользователя (см. параметр *Создание и изменение собственных медиа*).

Кроме того, права пользователя Super admin на изменение сведений о медиа других пользователей также могут быть предоставлены/отозваны в зависимости от его роли пользователя (см. параметр *Создание и изменение медиа пользователя*).

При обновлении с более старых версий оба параметра будут включены для всех ролей, у которых включен параметр *Доступ по умолчанию к новым действиям*. Обратите внимание: если параметр *Доступ по умолчанию к новым действиям* не включен, администраторы и супер-администраторы **могут потерять** возможность редактировать медиа после обновления.

Экранирование обратных слэшей в параметрах функций истории

Начиная с Zabbix 7.0.0, в строковых параметрах функций истории было добавлено корректное экранирование обратных слэшей.

Поскольку при обновлении с версий Zabbix до 7.0 в параметры добавляются дополнительные обратные слэши, это приводит к увеличению длины параметров, что может вызвать сбой функций триггера, если длина параметра превышает максимальный размер данных в 255 символов.

Чтобы избежать этой проблемы, рекомендуется вручную перенести длинные параметры в пользовательские макросы до обновления.

Для обновлений с версий Zabbix до 7.0 были добавлены дополнительные проверки итоговой длины параметра. Если итоговая длина превышает максимальный размер, такие параметры **не обновляются**, а в журналах выводится предупреждение с просьбой исправить перечисленные параметры вручную.

См. также проблемы обновления, связанные с экранированием.

Обновление плагина MSSQL Zabbix agent 2

Шаблон MSSQL by Zabbix agent 2 был обновлен: в него добавлены фильтры для включения или исключения обнаруженных членов кворума по имени, а также фильтр службы, который позволяет фильтровать по имени кластера, чтобы исключать пустые кластеры. Чтобы шаблон работал без ошибок, плагин MSSQL Zabbix agent 2 должен быть обновлен до версии 7.4.0 или выше.

Минимальная требуемая версия libssh2

Минимальная требуемая версия libssh2 была повышена с 1.0.0 до 1.8.0.

Прототипы узлов сети на обнаруженных узлах сети

Если ваша текущая конфигурация включает прототипы узлов сети, которые используют шаблоны, содержащие другие прототипы узлов сети, при обновлении до Zabbix 7.4 все обнаруженные узлы сети получат прототипы узлов сети, определенные в этих шаблонах. Если такое поведение не требуется, вручную удалите прототипы узлов сети из соответствующих шаблонов перед обновлением.

Чтобы определить обнаруженные узлы сети, которые получат прототипы узлов сети во время обновления, вы можете выполнить следующий SQL-запрос в вашей базе данных Zabbix:

```
SELECT h.hostid,ht.templateid
FROM hosts_templates ht
JOIN hosts h ON ht.hostid=h.hostid
WHERE h.flags=4
  AND EXISTS (
    SELECT NULL
    FROM items i,host_discovery hd
    WHERE i.hostid=ht.templateid
    AND hd.parent_itemid=i.itemid
    )
ORDER BY hostid;
```

Параметры подключения к базе данных

DBPort и DBSocket теперь взаимоисключающие. При указании параметров подключения к базе данных вы можете задать либо DBPort, либо DBSocket, либо оставить оба параметра не определенными, чтобы использовать значения по умолчанию.

**Другое** Неподдерживаемые элементы SNMP walk/discovery без действительных OID

Элементы SNMP walk, если ни один из OID или экземпляров OID не существует, теперь возвращают ошибку, и элемент становится неподдерживаемым. Сведения о несуществующем OID/экземпляре записываются в журнал с DebugLevel=5. Ранее в таких случаях возвращалась пустая строка.

Аналогично, элементы SNMP discovery, если ни один из OID или экземпляров OID не существует, теперь возвращают ошибку, и элемент становится неподдерживаемым. Ранее в таких случаях возвращался пустой массив.

Обратите внимание, что пустая строка/массив теперь возвращается только если OID/экземпляр существует, но для него нет данных.

Управление кэшем истории

В некоторых сценариях сбора данных отдельные элементы данных могут временно блокировать кэш истории сервера/прокси. Это может задерживать запись данных истории в базу данных и замедлять работу системы. Чтобы упростить управление кэшем истории, были внедрены следующие улучшения:

- **Ручная очистка кэша:** новая команда выполнения `history_cache_clear=target` для Zabbix сервер/прокси позволяет вручную очистить кэш истории для конкретного элемента данных по его ID.
- **Автоматическая очистка кэша:** при отключении элемента данных он немедленно удаляется из кэша истории (за исключением его последнего значения, которое сохраняется для журналов). Аналогично, при отключении узла сети все его элементы данных удаляются из кэша истории (за исключением их последних значений).
- **Журналирование диагностики кэша:** когда кэш истории заполнен, Zabbix сервер/прокси теперь записывает диагностическую информацию о кэше истории начиная с DebugLevel=3 (ранее DebugLevel=4). В журнале содержатся элементы данных с наибольшим количеством значений в кэше истории.

Для долгосрочной стабильности системы убедитесь, что сбор данных сбалансирован с доступными ресурсами (производительность базы данных, размер кэша, интервалы сбора, параметры элементов данных журнала и т. д.). Вы можете отслеживать кэш истории Zabbix с помощью внутреннего элемента данных zabbix[wcache]. Также можно рассмотреть увеличение размера кэша истории для Zabbix сервер/прокси.

Отдельный раздел меню для уведомлений пользователей

Для лучшей наглядности пользовательских способов доставки уведомлений теперь у уведомлений пользователей есть собственный раздел меню в *Настройках пользователя*.

Новый раздел *Уведомления* содержит две вкладки — *Медиа* и *Уведомления веб-интерфейса*, которые ранее входили в раздел профиля пользователя.

Корректная информация из вложенных групп узлов сети на картах

Информация из вложенных групп узлов сети теперь корректно отображается на картах, например:

- Метка группы узлов сети теперь отображает краткое описание проблемы со всех узлов сети во вложенных группах узлов сети;
- Представление "Элементы группы узлов сети" теперь отображает отдельный элемент карты для каждого узла сети во вложенных группах узлов сети;
- Метка карты теперь отображает краткое описание всех проблем, содержащихся во вложенных группах узлов сети.

Таблица базы данных для преобразованных настроек

Таблица `settings` теперь заменяет таблицу `config` для хранения параметров глобальной конфигурации. Новая таблица использует формат ключ-значение вместо хранения параметров в одной строке с отдельным столбцом для каждого

параметра. Пример того, как это влияет на обмен данными, см. в разделе Протокол обмена данными между сервером и прокси (объект "settings").

Увеличены максимальные размеры кэша для сервера и прокси

Максимальные размеры кэша были увеличены с 2 ГБ до 16 ГБ, чтобы отложить исчерпание кэша и обеспечить работу во время временных проблем (например, с конфигурацией, базой данных или сетью):

- сервер: увеличены максимальные значения для HistoryCacheSize, HistoryIndexCacheSize и TrendCacheSize;
- прокси: увеличены максимальные значения для HistoryCacheSize и HistoryIndexCacheSize.

Сокращен период хранения сеансов пользователей по умолчанию

Период хранения данных сеансов пользователей по умолчанию был сокращен с 365 дней до 31 дня. Это изменение затрагивает параметр hk_sessions, для которого теперь по умолчанию используется значение 31d вместо 365d.

Минимально поддерживаемая версия Go

Минимально поддерживаемая версия Go была повышена с 1.21 до 1.23.

Если ранее вы собирали Zabbix агент 2, загружаемые плагины агент 2 или веб-службу из исходного кода с использованием версии Go старее 1.23, рекомендуется пересобрать эти компоненты с использованием поддерживаемой версии Go, чтобы получать последние обновления безопасности и исправления ошибок. Компоненты, собранные с более старой версией Go, продолжат работать, но для их обновления потребуется более новая среда Go.

Обновление шаблонов для мастера узлов сети

Новый Мастер узлов сети предоставляет пошаговый интерфейс с подсказками для настройки объекта мониторинга (устройства, приложения, службы и т. д.) в Zabbix. Он упрощает настройку новых или существующих узлов сети, проводя пользователей через ключевые шаги, такие как выбор шаблона, установка агента Zabbix, добавление интерфейсов узла сети и многое другое.

После обновления с более ранней версии Zabbix шаблоны необходимо обновить, чтобы они работали с Мастером узлов сети. Инструкции см. в разделе Обновление шаблонов.

## 11 Примечания к обновлению до 7.4.x

На этой странице приведены общие примечания к обновлению для минорных выпусков основной версии Zabbix.

См. также примечания к обновлению основной версии.

**Примечания к обновлению до 7.4.11** Восстановлена поддержка макросов LLD в переопределениях правил LLD

При настройке операций переопределения правила обнаружения низкого уровня теперь снова можно использовать макросы LLD в параметрах прототипа элемента данных *Update interval*, *History* и *Trends*.

**Примечания к обновлению до 7.4.10** Элементы тревог VMware возвращают сведения об объекте

Возвращаемый JSON элементов `vmware.*.alarms.get[]` теперь включает объект `entity` для каждой тревоги. Этот объект возвращает ID, UUID и тип затронутого объекта (узел сети, виртуальная машина или хранилище данных).

Если у вас есть скрипты, которые разбирают вывод элемента тревоги VMware, возможно, потребуется обновить их, чтобы они обрабатывали дополнительные поля.

Обновленная библиотека Duo Universal PHP

Библиотека Duo Universal PHP (поставляется вместе с Zabbix) была обновлена с версии 1.1.1 до 1.1.2.

**Примечания к обновлению до 7.4.9** В этой минорной версии нет примечаний к обновлению.

**Примечания к обновлению до 7.4.8** Значения элементов данных, отображаемые как HTML с использованием iframe

Данные элемента данных в виджете Item history, при отображении в виде HTML-форматированного текста, теперь изолированы в iframe.

Обновленная библиотека SAML PHP Toolkit

Библиотека SAML PHP Toolkit (поставляемая вместе с Zabbix) была обновлена с версии 4.0.0 до 4.3.1.

**Примечания к обновлению до 7.4.7** Обновленная библиотека Duo Universal PHP

Библиотека Duo Universal PHP (поставляется вместе с Zabbix) была обновлена с версии 1.0.2 до 1.1.1.

**Примечания к обновлению до 7.4.6** Ограниченные разрешения для файлов журналов

Файлы журналов компонентов Zabbix теперь создаются с разрешениями на чтение и запись только для владельца файла. Кроме того, файлы доступны для чтения группе владельца. Все остальные разрешения запрещены.

В Linux это изменение разрешений с 0666 на 0640. Изменение затрагивает файлы журналов сервера, прокси и агента Zabbix. Разрешения для файлов журналов агента Zabbix 2 были изменены с 0644 на 0640 (путем удаления разрешения на чтение для остальных пользователей).

Кроме того, несколько **других файлов** теперь создаются с разрешениями 0640:

- UNIX pid-файлы для процессов Zabbix
- файлы экспорта данных
- временные файлы для проверок `icmpping*`
- файлы постоянного журнала

Запросы статуса репликации MySQL, обрабатываемые агентом Zabbix agent 2

Zabbix agent 2 плагин MySQL теперь поддерживает как `SHOW SLAVE STATUS`, так и `SHOW REPLICA STATUS`. Плагин автоматически выбирает и выполняет соответствующий оператор в зависимости от версии сервера и возвращает результат в обычном формате JSON.

Элементы данных agent 2, такие как `mysql.replication.discovery` и `mysql.replication.get_slave_status`, работают с серверами, которые используют либо `SHOW SLAVE STATUS`, либо `SHOW REPLICA STATUS`. Изменения в существующих элементах данных agent 2 или шаблонах agent 2 не требуются.

Минимально поддерживаемая версия Go

Минимально поддерживаемая версия Go повышена с 1.23 до 1.24.10.

Если ранее вы собирали Zabbix агент 2, загружаемые плагины агент 2 или веб-службу из исходного кода с использованием версии Go старее 1.24.10, рекомендуется пересобрать эти компоненты с использованием поддерживаемой версии Go, чтобы получать последние обновления безопасности и исправления ошибок. Компоненты, собранные с более старой версией Go, продолжат работать, но для их обновления потребуется более новая среда Go.

Включение файлов или папок в конфигурацию веб-службы

В файл конфигурации веб-службы Zabbix был добавлен параметр Include. Этот параметр позволяет включать в конфигурацию отдельные файлы или все файлы в каталоге.

**Примечания к обновлению до 7.4.5** В этой минорной версии нет примечаний к обновлению.

**Примечания к обновлению до 7.4.4 Внимание!** Обновление до этой версии не рекомендуется из-за ошибок, связанных с графиками и плагином MySQL для Zabbix agent 2.

**Примечания к обновлению до 7.4.3** Минимальная версия PHP cURL

Минимально поддерживаемая версия расширения cURL PHP extension теперь 7.19.4.

**Примечания к обновлению до 7.4.1** Поток вывода управления выполнением агента 2

Команды Zabbix agent 2 управления выполнением (`zabbix_agent2 -R <option>`) теперь записывают вывод в `stdout` (стандартный вывод) вместо `stderr` (стандартный поток ошибок). Пожалуйста, обновите все скрипты, которые зависят от предыдущего поведения.

## 5 Быстрый старт

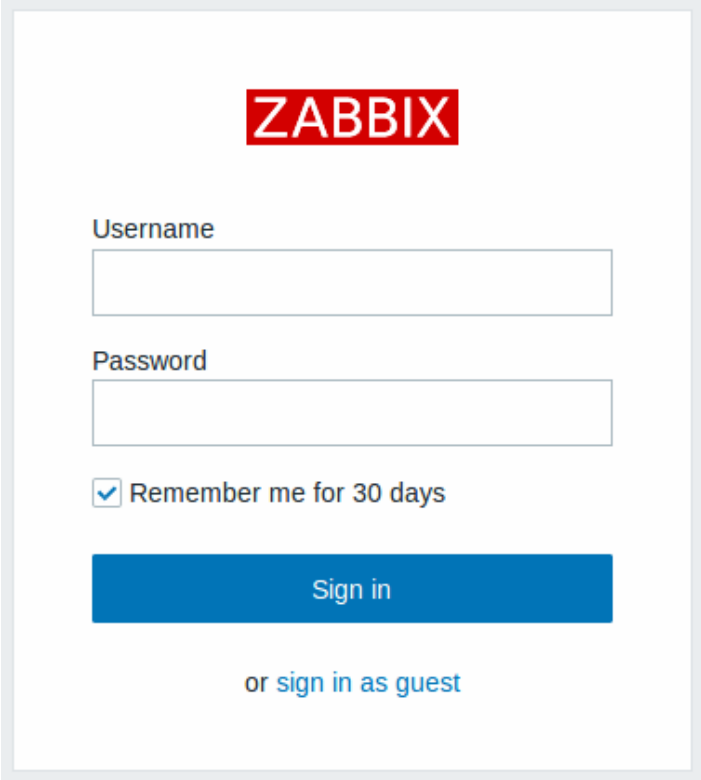
Этот раздел представляет собой пошаговое руководство по настройке базового мониторинга: как войти в Zabbix, добавить узел сети, определить элемент данных, создать триггер, получать уведомления и сохранить конфигурацию как шаблон.

**1 Login and configuring user**

Overview

In this section, you will learn how to log in and set up a system user in Zabbix.

Вход




Это приветственный экран Zabbix. Введите имя пользователя **Admin** и пароль **zabbix**, чтобы войти в систему как суперпользователь Zabbix. Будет предоставлен доступ ко всем разделам меню.

В целях безопасности настоятельно рекомендуется сразу после первого входа изменить пароль по умолчанию пароль для учетной записи Admin.

Постоянный вход в систему

Чтобы оставаться в системе до 30 дней, выберите *Запомнить на 30 дней* перед нажатием *Войти*.

*Запомнить на 30 дней* включено:

- Ваша сессия остается активной в течение 30 дней.
- *Автовыход* переопределяется, и вы остаетесь в системе до окончания этого периода.
- При последующих посещениях в течение 30 дней вы будете входить в систему автоматически, без повторного ввода учетных данных.

*Запомнить на 30 дней* отключено:

- Любой ранее включенный автоматический вход очищается.
- Сессия истечет в соответствии с настроенным интервалом *Автовыход*.

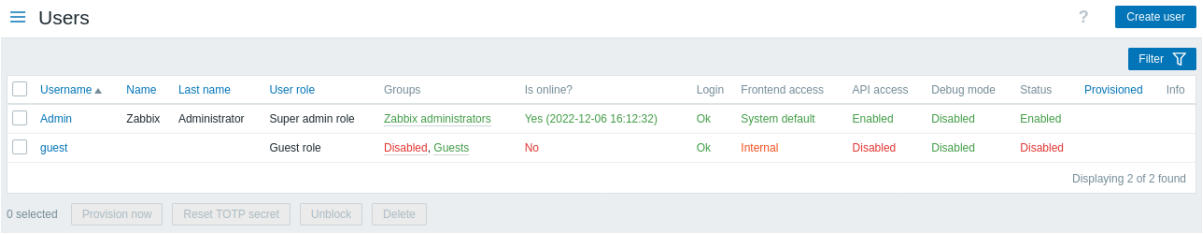
Protection against brute force attacks

In case of five consecutive failed login attempts, Zabbix interface will pause for 30 seconds in order to prevent brute force and dictionary attacks.

The IP address of a failed login attempt will be displayed after a successful login.

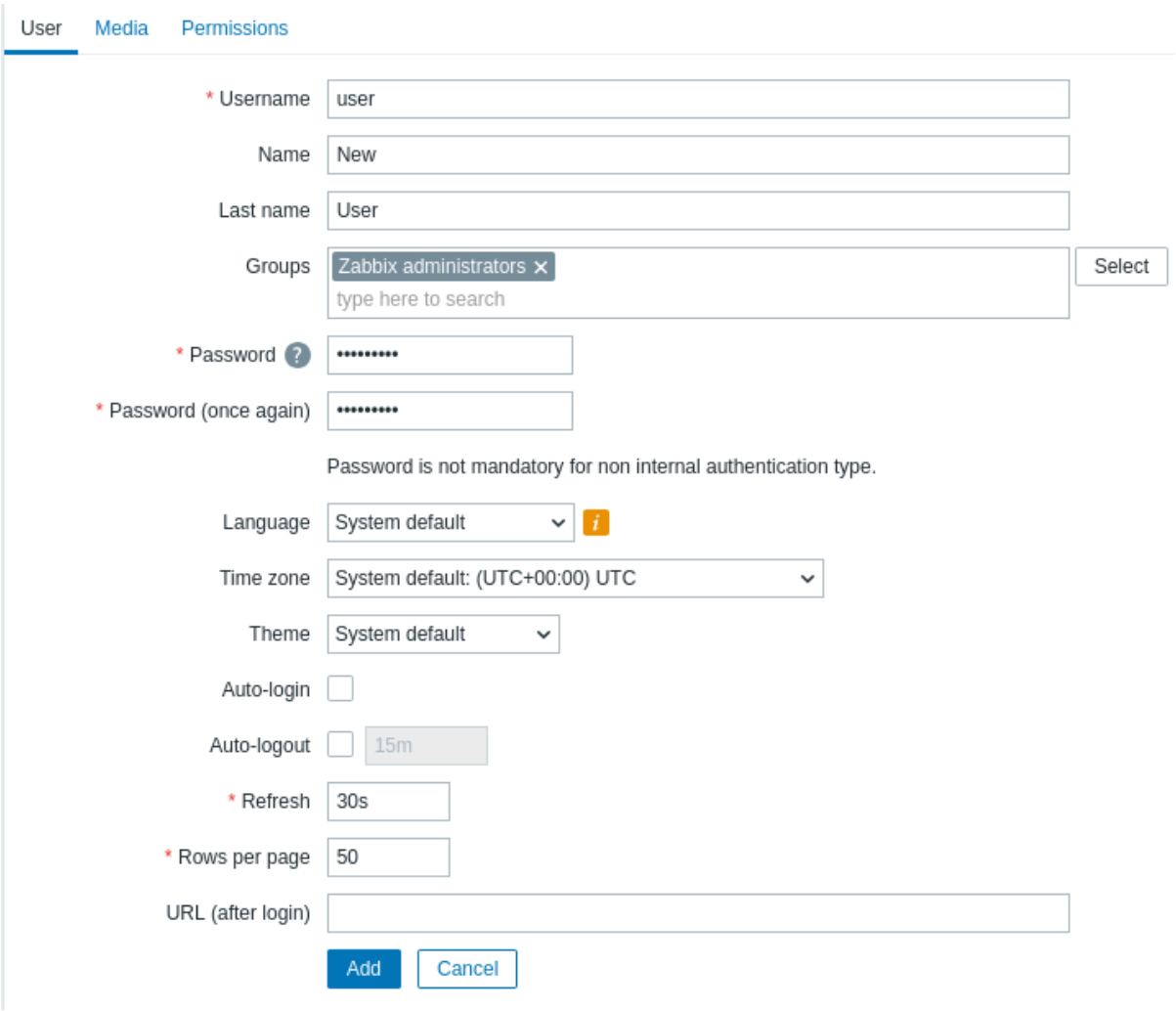
Добавление пользователя

Чтобы просмотреть информацию о пользователях, перейдите в *Пользователи* > *Пользователи* в вертикальном меню боковой панели.



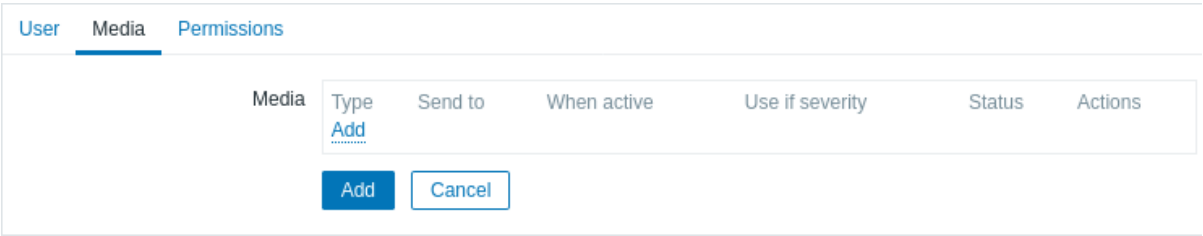

Чтобы добавить нового пользователя, выберите *Создать пользователя* в правом верхнем углу.

В форме нового пользователя обязательно добавьте пользователя в одну из существующих групп пользователей, например в 'Zabbix administrators'.



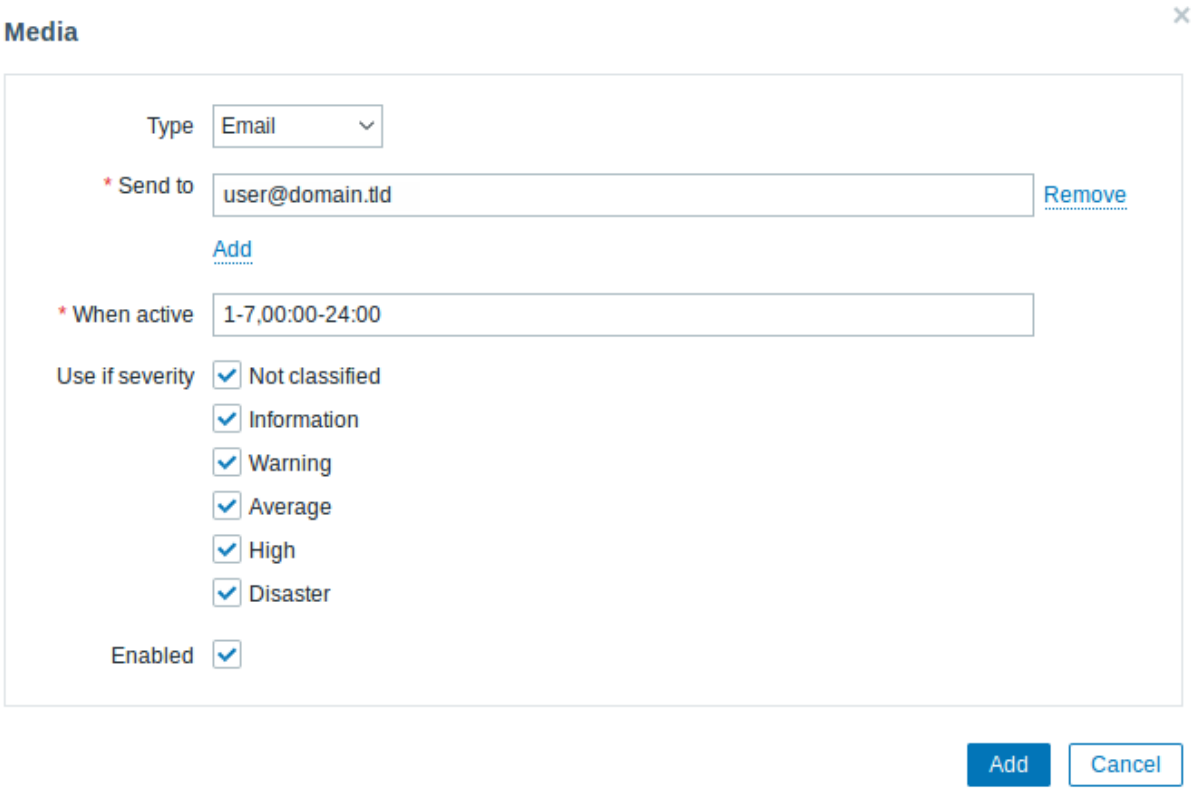

Все обязательные поля ввода отмечены красной звездочкой. Подробности о полях ввода в этой форме настройки см. на странице Настройки пользователя.

По умолчанию для новых пользователей не определены медиа (способы доставки уведомлений). Чтобы создать их, перейдите на вкладку 'Media' и нажмите *Добавить*.



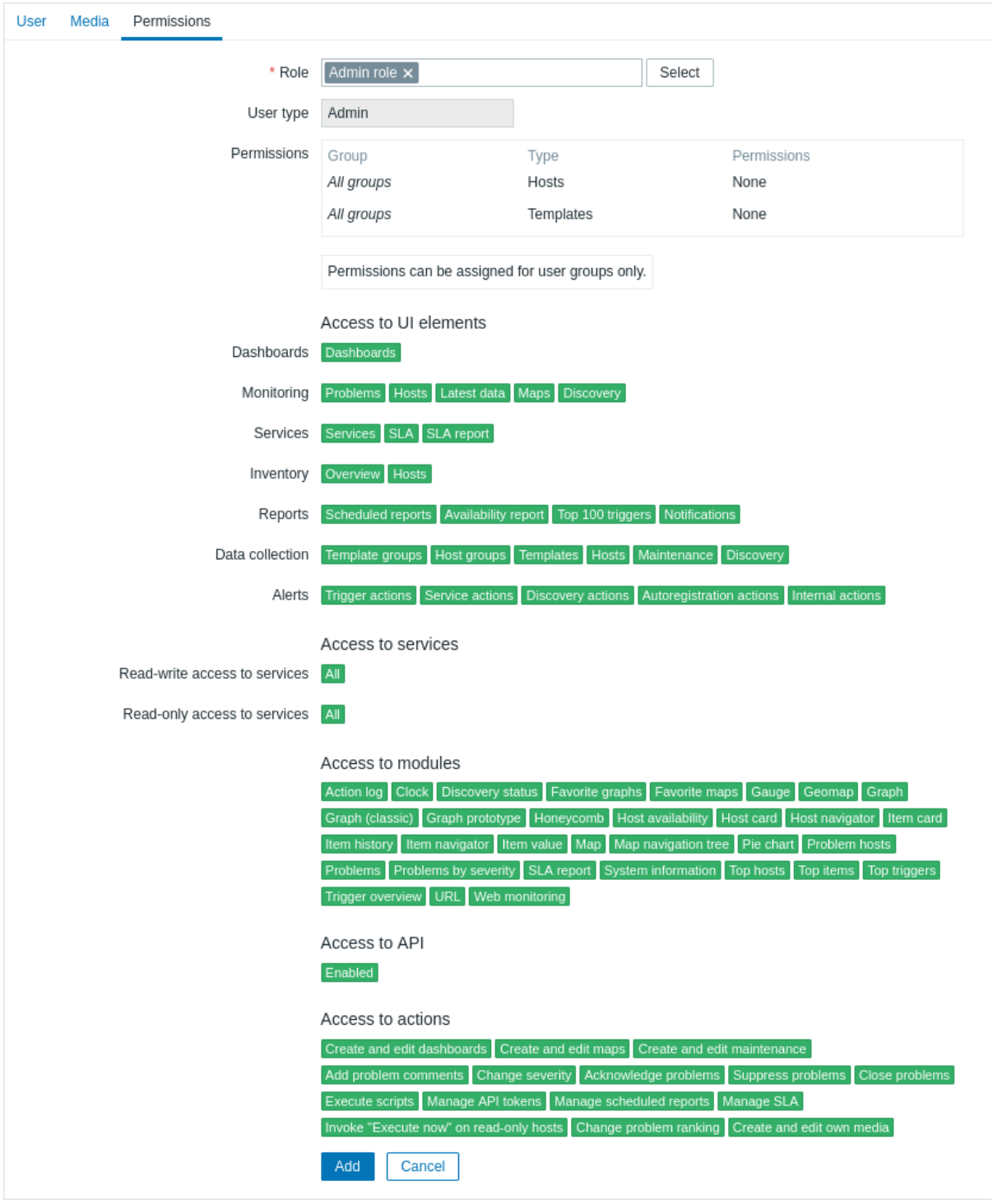

Во всплывающем окне введите адрес электронной почты пользователя.

Вы можете указать период времени, в течение которого медиа будет активно (описание формата см. на странице Описание временных периодов). По умолчанию медиа всегда активно. Вы также можете настроить уровни серьезности триггера, для которых медиа будет активно, но пока оставьте все уровни включенными.




Нажмите *Добавить*, чтобы сохранить медиа, затем перейдите на вкладку Permissions.

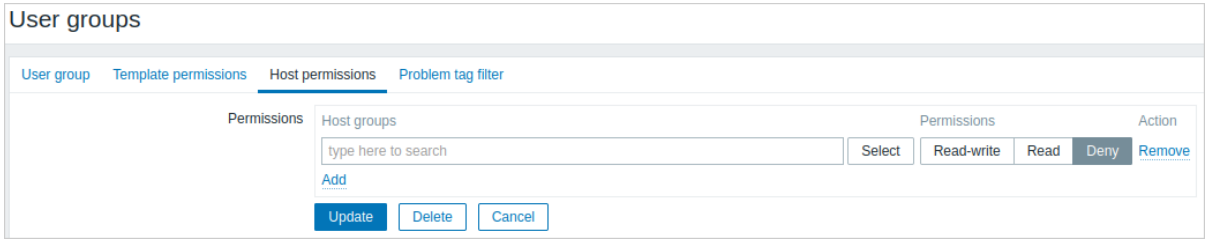
На вкладке Permissions есть обязательное поле *Role*. Роль определяет, какие элементы веб-интерфейса пользователь может просматривать и какие действия ему разрешено выполнять. Нажмите *Select* и выберите одну из ролей из списка. Например, выберите *Admin role*, чтобы разрешить доступ ко всем разделам веб-интерфейса Zabbix, кроме Administration. Позже вы сможете изменить права или создать дополнительные роли пользователей. После выбора роли права отобразятся на той же вкладке:



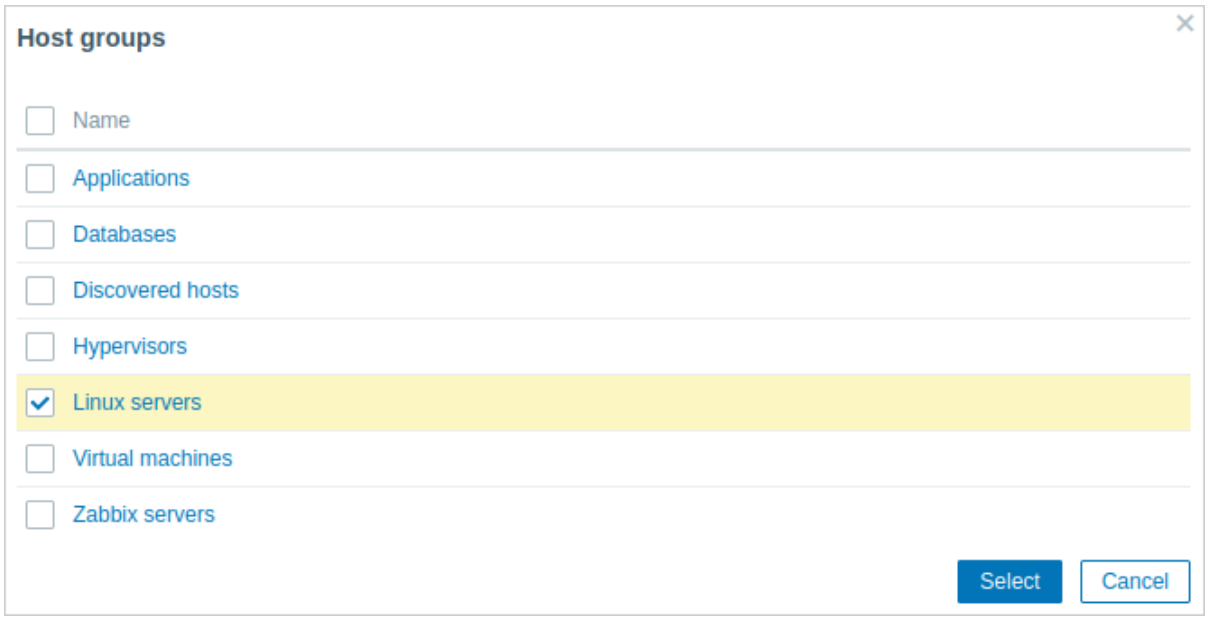

Нажмите *Add* в форме свойств пользователя, чтобы сохранить пользователя. Новый пользователь появится в списке пользователей.

Добавление прав доступа

По умолчанию новый пользователь не имеет прав доступа к узлам сети и шаблонам. Чтобы предоставить пользователю права, нажмите на группу пользователя в столбце *Groups* (в данном случае - 'Zabbix administrators'). В форме свойств *User groups* перейдите на вкладку *Host permissions*, чтобы назначить права для групп узлов сети. Нажмите Add в поле выбора группы узлов сети, чтобы отобразить его:




Затем нажмите *Select* рядом с полем, чтобы увидеть список групп узлов сети. Для этого пользователя требуется доступ только для чтения к группе *Linux servers*, поэтому отметьте соответствующий флажок в списке и нажмите *Select*, чтобы подтвердить свой выбор.




Нажмите кнопку *Read*, чтобы задать уровень прав доступа, а затем *Update*, чтобы сохранить изменения, внесенные в конфигурацию группы пользователей.

Чтобы предоставить права доступа к шаблонам, нужно перейти на вкладку *Template permissions* и указать группы шаблонов. Шаги аналогичны назначению прав для групп узлов сети. Обзор шаблонов доступен в разделе New template этого Quickstart.

**Attention:**
В Zabbix права доступа к узлам сети и шаблонам назначаются группам пользователей, а не отдельным пользователям.

Готово! Теперь вы можете попробовать войти, используя учетные данные нового пользователя.

**2. Новый узел сети**

Обзор

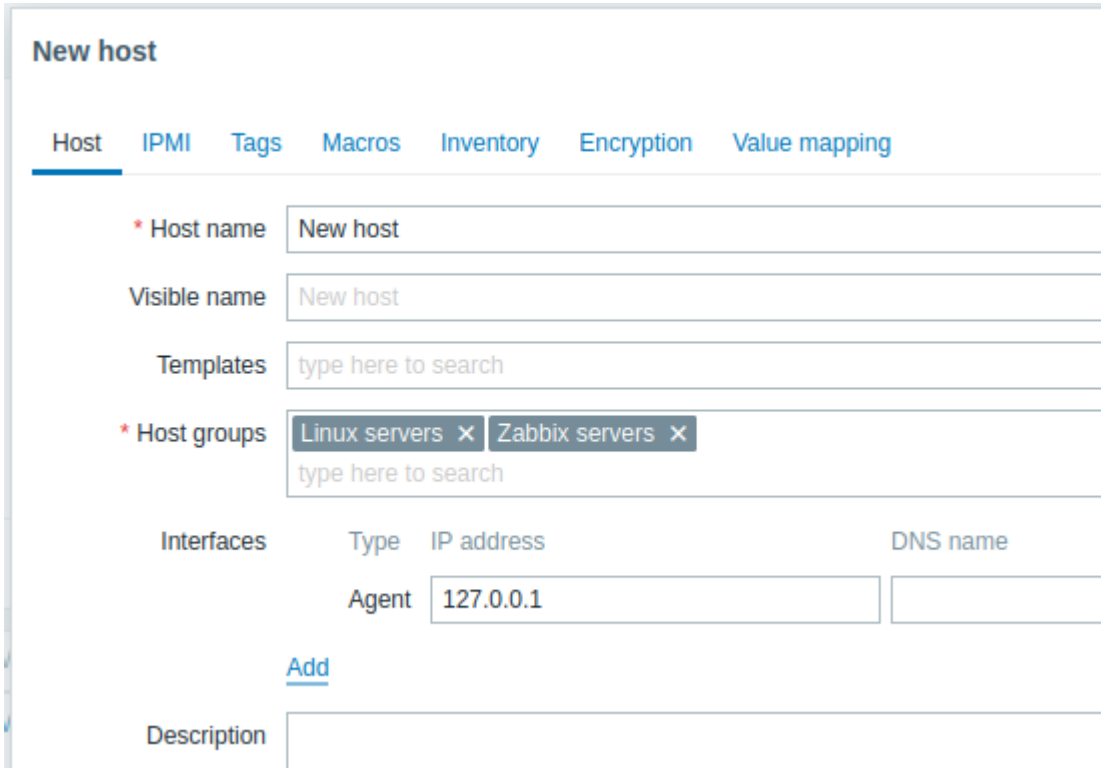
В этом разделе вы узнаете, как настроить новый узел сети.

Узел сети в Zabbix — это объект сети (физический, виртуальный), который вы хотите наблюдать. Определение того, что может быть «узлом сети» в Zabbix, очень гибкое. Это может быть физический сервер, сетевой коммутатор, виртуальная машина или какое-нибудь приложение.

Добавление узла сети

Информация о настроенных узлах сети в Zabbix доступна в разделах меню *Сбор данных* > *Узлы сети* и *Мониторинг* > *Узлы сети*. Уже существует один предопределенный узел сети, называемый "Zabbix server", но мы хотим научиться добавлять еще один.

Чтобы добавить новый узел сети, нажмите *Создать узел сети*. Откроется форма настройки узла сети.




Все обязательные поля ввода отмечены красной звездочкой.

Минимально необходимо указать следующее:

***Имя узла сети***

- Введите имя узла сети. Допускаются буквенно-цифровые символы, пробелы, точки, дефисы и символы подчеркивания.

***Группы узлов сети***

- Выберите одну или несколько существующих групп, нажав кнопку *Выбрать*, или введите имя несуществующей группы, чтобы создать новую группу.

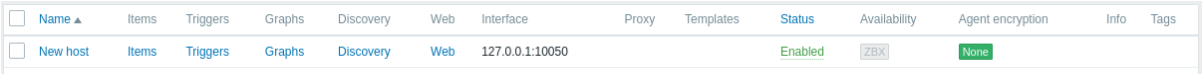
**Note:**
Все права доступа назначаются группам узлов сети, а не отдельным узлам сети. Поэтому узел сети должен принадлежать как минимум к одной группе.

***Интерфейсы: IP-адрес***

- Хотя технически это поле не является обязательным, интерфейс узла сети необходим для сбора некоторых метрик. Чтобы использовать пассивные проверки агента Zabbix, укажите в этом поле IP-адрес или DNS-имя агента. Обратите внимание, что также следует указать IP-адрес или DNS-имя сервера Zabbix в директиве 'Server' файла конфигурации агента Zabbix. Если агент Zabbix и сервер Zabbix установлены на одной машине, необходимо указать один и тот же IP-адрес/DNS-имя в обоих местах.

Другие параметры можно оставить без изменений, используя значения по умолчанию.

После завершения нажмите *Добавить*. Новый узел сети должен появиться в списке узлов сети.




Столбец Availability содержит индикаторы доступности узла сети для каждого интерфейса. Мы определили интерфейс агента Zabbix, поэтому можем использовать значок доступности агента (с надписью 'ZBX'), чтобы определить доступность узла сети:

- ZBX - статус узла сети не определен; проверка метрик еще не выполнялась
- ZBX - узел сети доступен, проверка метрик успешно выполнена
- ZBX - узел сети недоступен, проверка метрик завершилась с ошибкой (наведите курсор мыши на значок, чтобы увидеть сообщение об ошибке). Возможна ошибка связи, вероятно вызванная неверными учетными данными

интерфейса. Проверьте, что сервер Zabbix запущен, и попробуйте обновить страницу позже.

### 3. Новый элемент данных

Обзор

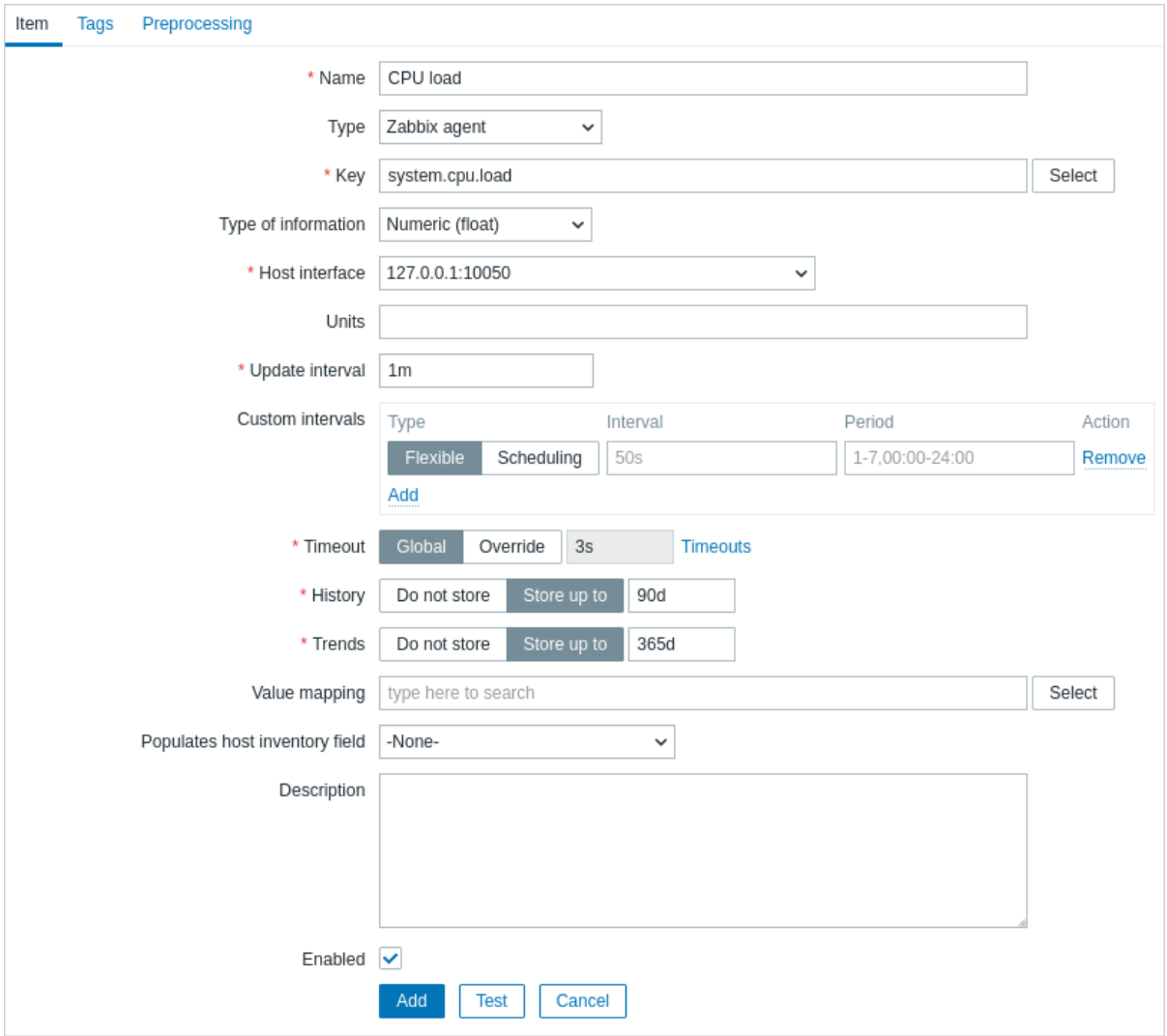
В этом разделе вы узнаете, как настроить элемент данных.

Элементы данных лежат в основе сбора данных в Zabbix. Без элементов данных нет данных — потому что только элемент данных определяет одну метрику, или какого рода данные собираются с узла сети.

Добавление элемента данных

Все элементы данных группируются вокруг узлов сети. Поэтому, чтобы настроить пример элемента данных, мы переходим в *Сбор данных* > *Узлы сети* и находим созданный нами "New host".

Нажмите ссылку *Элементы данных* в строке "New host", а затем нажмите *Создать элемент данных*. Откроется форма настройки элемента данных.




Все обязательные поля ввода отмечены красной звездочкой.

Для нашего примера элемента данных необходимо указать следующую информацию:

***Имя***

- Введите значение *CPU load*. Это будет имя элемента данных, отображаемое в списках и в других местах.

***Ключ***

- Вручную введите значение *system.cpu.load*. Это техническое имя элемента данных, которое определяет тип собираемой информации. Указанный ключ — это лишь один из предопределенных ключей, поставляемых с агентом Zabbix.

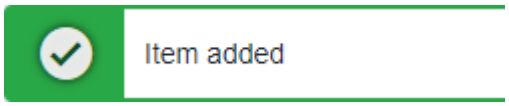
***Тип информации***

- Этот атрибут определяет формат ожидаемых данных. Для ключа *system.cpu.load* это поле будет автоматически установлено в *Numeric (float)*.

**Note:**
Также рекомендуется уменьшить количество дней хранения истории элемента данных до 7 или 14. Это хорошая практика, позволяющая снизить нагрузку на базу данных, которой не придется хранить большое количество исторических значений.

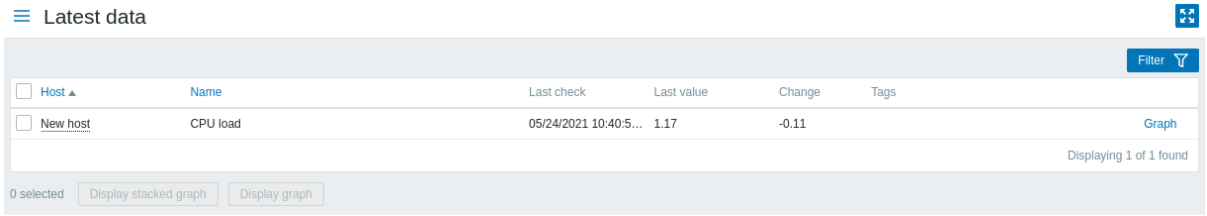
Другие параметры можно оставить без изменений, используя значения по умолчанию.

После завершения нажмите *Добавить*. Новый элемент данных должен появиться в списке элементов данных, и вы увидите сообщение об успешном выполнении.




Просмотр данных

После того как элемент данных определен, вам может быть интересно, действительно ли он собирает данные. Для этого перейдите в *Мониторинг* > *Последние данные*, выберите 'New host' в фильтре и нажмите *Применить*.




При этом может потребоваться до 60 секунд, чтобы появились первые данные. По умолчанию именно с такой периодичностью сервер считывает изменения конфигурации и подхватывает новые элементы данных для выполнения.

Если в столбце 'Change' нет значения, возможно, пока получено только одно значение. Подождите 30 секунд, пока не поступит еще одно значение.

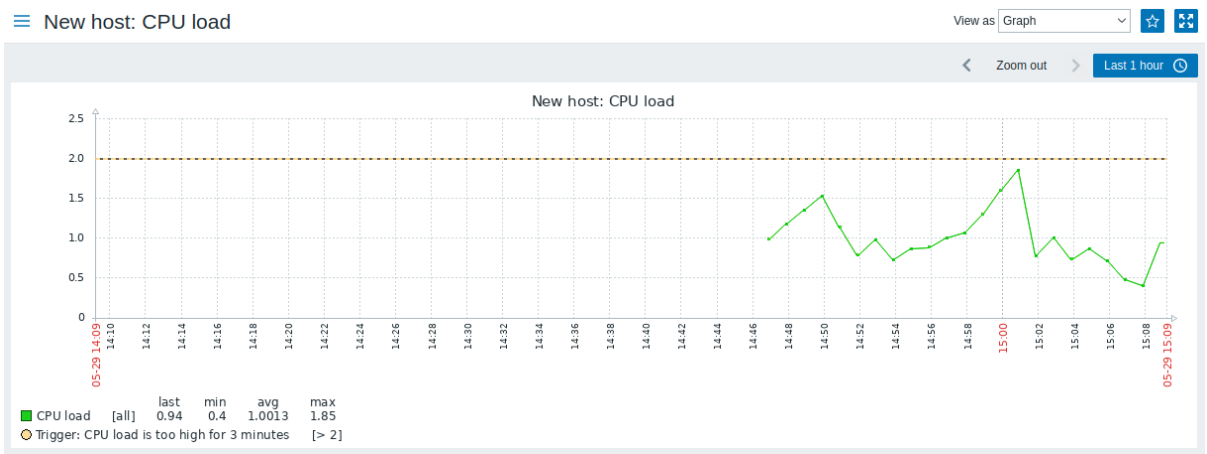
Если вы не видите информацию об элементе данных, как на снимке экрана, убедитесь, что:

- вы заполнили поля элемента данных 'Key' и 'Type of information' точно так же, как на снимке экрана;
- и агент, и сервер запущены;
- статус узла сети — 'Monitored', а значок доступности зеленый;
- выбранный в фильтре узел сети указан правильно;
- элемент данных включен.

Графики

Когда элемент данных поработает некоторое время, возможно, пришло время увидеть что-то визуальное. Простые графики доступны для любого отслеживаемого числового элемента данных без какой-либо дополнительной настройки. Эти графики генерируются во время выполнения.

Чтобы просмотреть график, перейдите в *Мониторинг* > *Последние данные* и нажмите ссылку 'Graph' рядом с элементом данных.



## 4. Новый триггер

Обзор

В этом разделе вы узнаете, как настроить триггер.

Элементы данных только собирают данные. Чтобы автоматически оценивать поступающие данные, нам нужно определить триггеры. Триггер содержит выражение, которое задает порог допустимого уровня для данных.

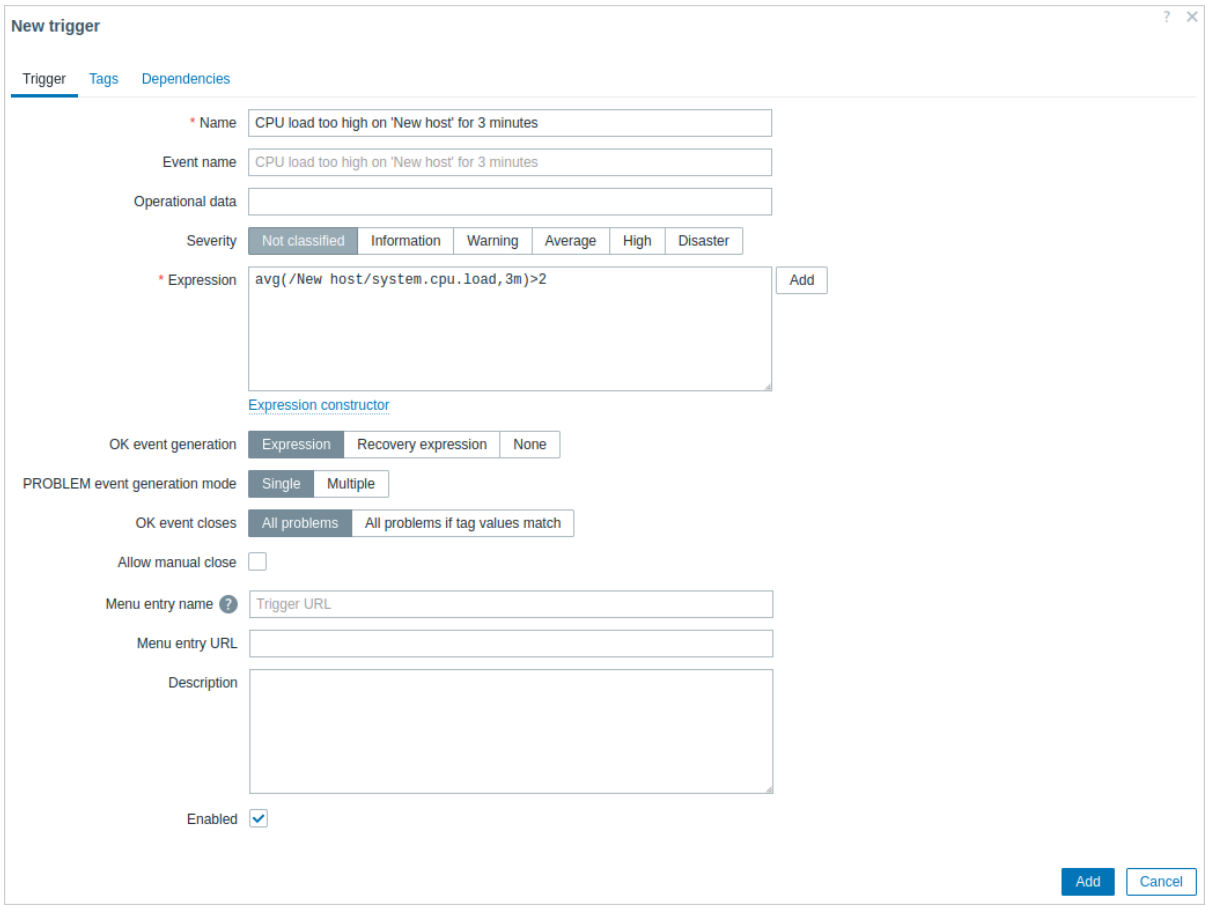
Если поступающие данные превысят этот уровень, триггер ”сработает” или перейдет в состояние ’Problem’ — это означает, что произошло что-то, что может потребовать внимания. Если уровень снова станет допустимым, триггер возвращается в состояние ’Ok’.

Добавление триггера

Чтобы настроить триггер для нашего элемента данных:

1. Перейдите в *Сбор данных* > *Узлы сети*.
2. Нажмите ссылку *Триггеры* в строке ’New host’.
3. Нажмите *Создать триггер*.

Откроется форма настройки триггера.




Для нашего триггера здесь нужно указать следующую основную информацию:

*Имя*

- Введите *CPU load too high on ’New host’ for 3 minutes* в качестве значения. Это будет имя триггера, отображаемое в списках и других местах.

*Выражение*

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

Это выражение триггера. Убедитесь, что выражение введено правильно, вплоть до последнего символа. Ключ элемента данных здесь (`system.cpu.load`) используется для ссылки на элемент данных. По сути, это конкретное выражение означает, что порог проблемы превышен, когда среднее значение загрузки CPU за 3 минуты больше 2. Подробнее о синтаксисе выражений триггеров.

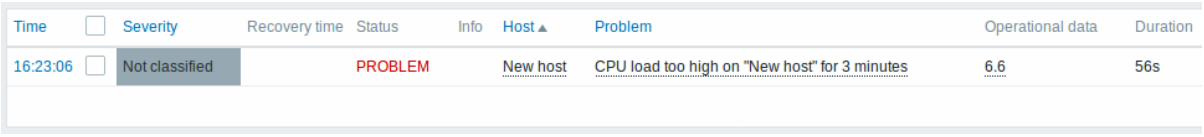
Другие параметры можно оставить без изменений, используя значения по умолчанию.

После завершения нажмите *Добавить*. Новый триггер должен появиться в списке триггеров.

Отображение статуса триггера

Если триггер определен, вам может быть интересно посмотреть его статус.

Если загрузка CPU превысила пороговый уровень, заданный в триггере, проблема будет отображаться в *Мониторинг* > *Проблемы*.



| Time | | Severity | Recovery time | Status | Info | Host ▲ | Problem | Operational data | Duration |
|---|---|---|---|---|---|---|---|---|---|
| 16:23:06 | | Not classified | | PROBLEM | | New host | CPU load too high on "New host" for 3 minutes | 6.6 | 56s |

Мигание в столбце статуса указывает на недавнее изменение статуса триггера, произошедшее за последние 30 минут.

**5 Receiving problem notification**

Обзор

В этом разделе вы узнаете, как настроить оповещения в виде уведомлений в Zabbix.

Поскольку элементы данных собирают данные, а триггеры предназначены для срабатывания при возникновении проблемных ситуаций, полезно также иметь механизм оповещения, который будет уведомлять о важных событиях даже тогда, когда веб-интерфейс Zabbix не проверяется активно.

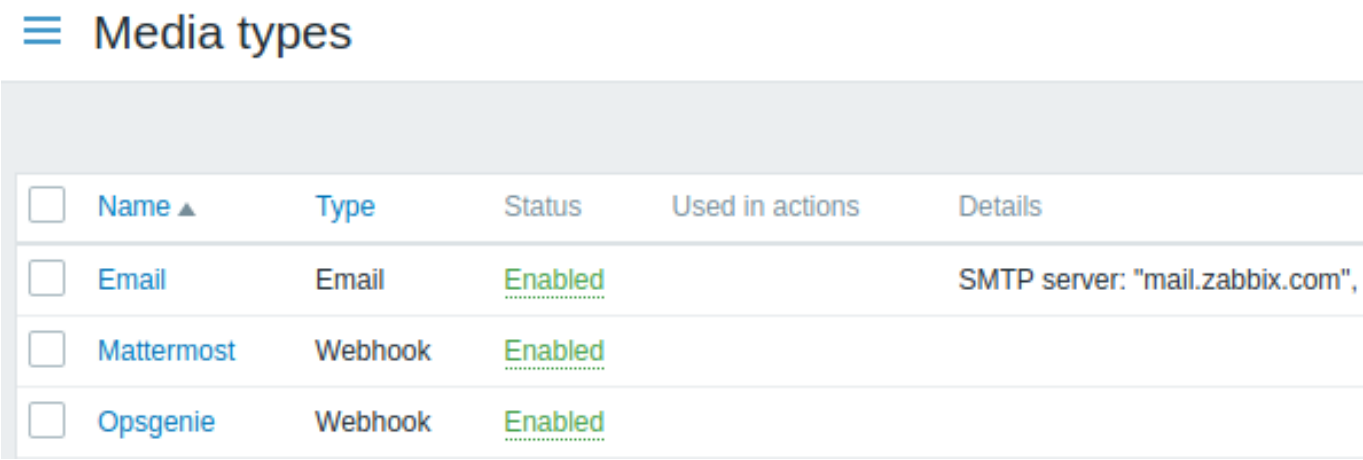
Именно это и делают уведомления.

Поскольку электронная почта является самым популярным способом доставки уведомлений о проблемах, мы рассмотрим, как настроить уведомление по электронной почте.

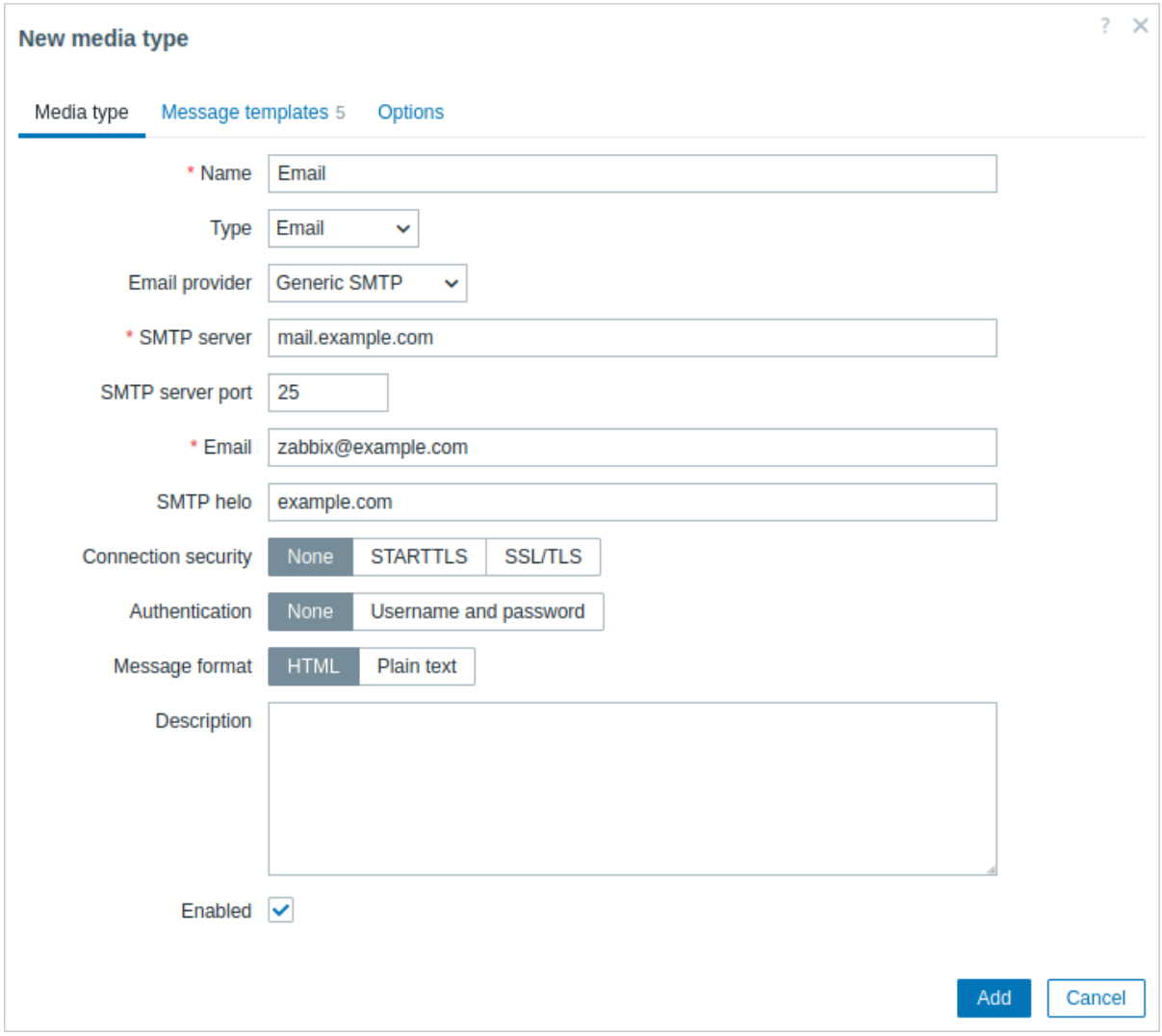
Настройки электронной почты

Изначально в Zabbix есть несколько предопределенных способов доставки уведомлений. Email — один из них.

Чтобы настроить параметры электронной почты, перейдите в *Уведомления* > *Типы медиа* и нажмите *Email* в списке предопределенных типов медиа.




Откроется форма настройки параметров электронной почты.



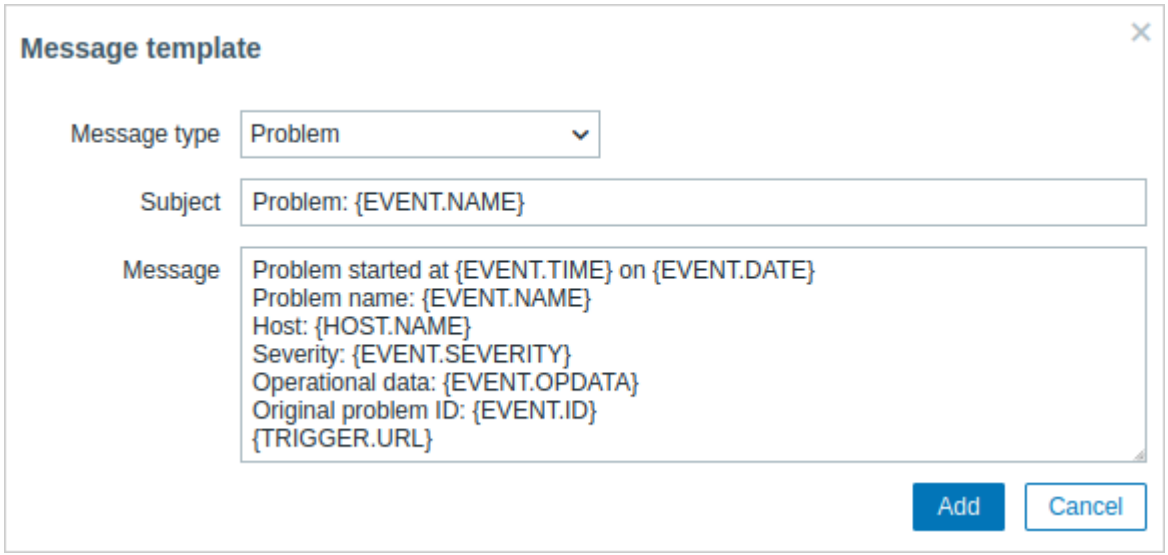

Все обязательные поля ввода отмечены красной звездочкой.

На вкладке *Тип медиа* задайте значения *SMTP server*, *SMTP helo* и *Email* в соответствии с вашей средой.

**Note:**
Значение в поле *Email* будет использоваться как адрес 'From' для уведомлений, отправляемых из Zabbix.

Далее необходимо определить содержимое сообщения о проблеме. Содержимое задается с помощью шаблона сообщения, который настраивается на вкладке *Message templates*.

Нажмите *Add*, чтобы создать шаблон сообщения, и выберите *Problem* в качестве типа сообщения.

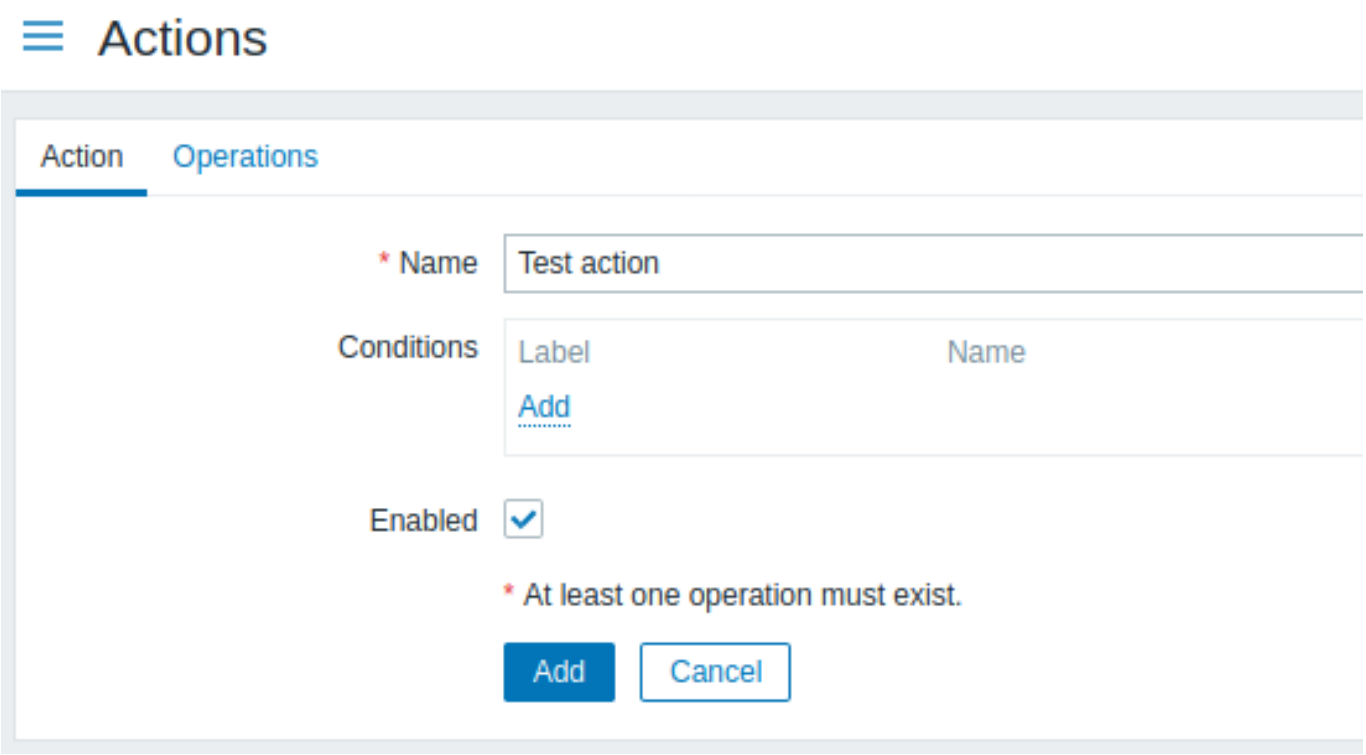


Когда все будет готово, нажмите *Add* и сохраните форму.

Теперь вы настроили *Email* как рабочий тип медиа. Тип медиа также должен быть привязан к пользователям путем указания конкретных адресов доставки (как мы делали при настройке нового пользователя), иначе он не будет использоваться.

Новое действие

Доставка уведомлений — одна из задач, которые выполняют действия в Zabbix. Поэтому, чтобы настроить уведомление, перейдите в *Alerts* > *Actions* > *Trigger actions* и нажмите *Create action*.

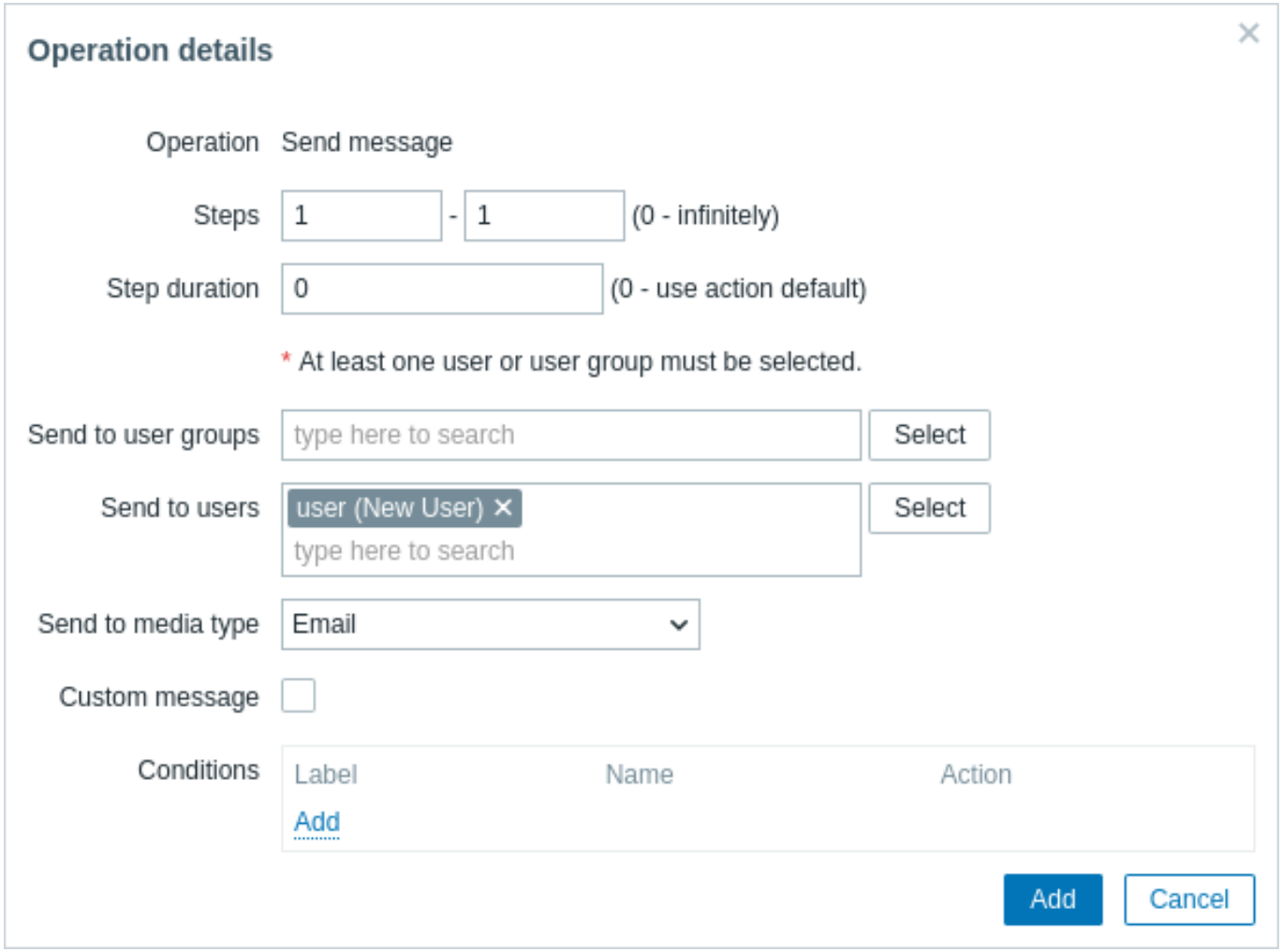



Все обязательные поля ввода отмечены красной звездочкой.

В этой форме введите имя действия.

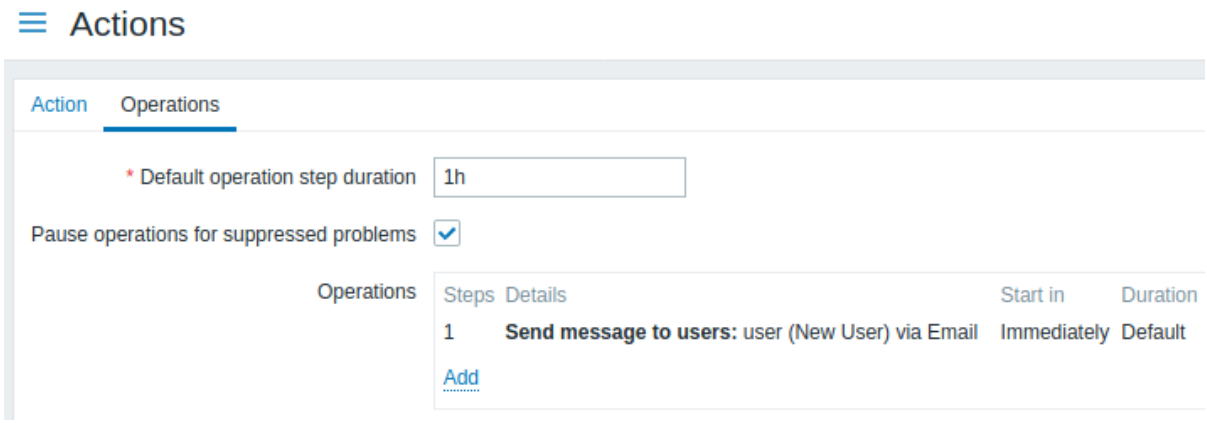
В самом простом случае, если мы не добавляем никаких более конкретных условий, действие будет выполнено при любом изменении триггера с 'Ok' на 'Problem'.

Нам все еще нужно определить, что должно делать действие, — это настраивается на вкладке *Operations*. Нажмите *Add* в блоке *Operations*, чтобы открыть новую форму операции.




Все обязательные поля ввода отмечены красной звездочкой.

Здесь нажмите *Select* в блоке *Send to Users* и выберите пользователя (’user’), которого мы определили. Выберите ”Email” в качестве значения *Send to media type*. После этого нажмите *Add*, и операция должна быть добавлена:




Для простой настройки действия этого достаточно, поэтому нажмите *Add* в форме действия.

Receiving notification

Now, with delivering notifications configured, it would be fun to actually receive one. To help with that, we might on purpose increase the load on our host - so that our trigger ”fires” and we receive a problem notification.

Open the console on your host and run:

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

You may run one or several of these processes.

Now go to *Monitoring* > *Latest data* and see how the values of ’CPU Load’ have increased.Remember, for our trigger to ”fire”, the ’CPU Load’ value has to go over ’2’ for 3 minutes running. Once it does:

- in *Monitoring* > *Problems* you should see the trigger with a flashing 'Problem' status;
- you should receive a problem notification in your email.

**Attention:**
If notifications do not work:
- verify once again that both the email settings and the action have been configured properly
- make sure the user you created has at least read permissions on the host which generated the event, as noted in the *Adding user* step. The user, being part of the 'Zabbix administrators' user group must have at least read access to 'Linux servers' host group that our host belongs to.
- Additionally, you can check out the action log by going to *Reports* > *Action log*.

### 6. Новый шаблон

Обзор

В этом разделе вы узнаете, как настроить шаблон.

Ранее мы узнали, как настроить элемент данных, триггер и как получать оповещения о проблемах с этим узлом сети.

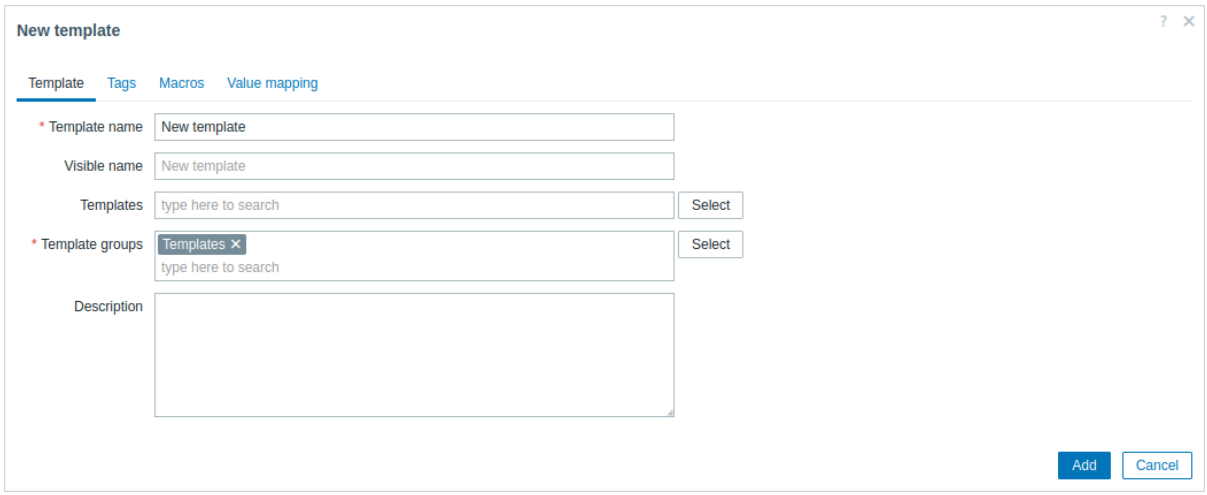
Хотя все эти шаги дают большую гибкость как таковые, может показаться, что потребуется множество шагов, если речь идёт, скажем, о тысяче узлов сети. Автоматизация процесса была бы кстати.

В этом нам придут на помощь шаблоны. Шаблоны позволяют группировать полезные элементы данных, триггеры и другие сущности, чтобы они могли быть использованы снова и снова простым применением шаблона к узлам сети за один шаг.

Когда шаблон присоединяется к узлу сети, то узел наследует все сущности шаблона. Таким образом, в основном заранее подготовленная куча проверок может быть применена очень быстро.

Добавление шаблона

Чтобы начать работу с шаблонами, сначала нужно создать шаблон. Для этого в разделе *Сбор данных* > *Шаблоны* нажмите *Создать шаблон*. Откроется форма настройки шаблона.




Все обязательные поля ввода отмечены красной звездочкой.

Необходимые параметры для ввода здесь:

***Имя шаблона***

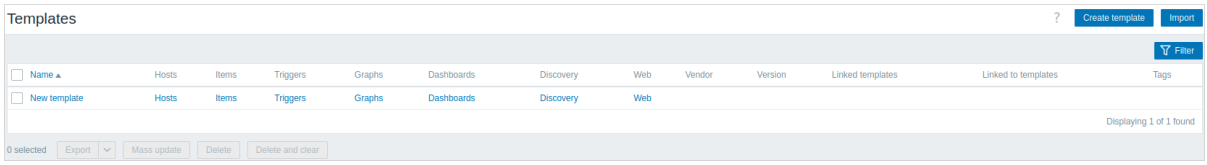
- Введите имя шаблона. Допускаются буквенно-цифровые символы, пробелы и символы подчеркивания.

***Группы шаблонов***

- Выберите одну или несколько групп, нажав кнопку *Выбрать*. Шаблон должен принадлежать к группе.

**Note:**
Права доступа к группам шаблонов назначаются в конфигурации группы пользователей на вкладке **Права на шаблоны** так же, как и права на узлы сети. Все права доступа назначаются группам, а не отдельным шаблонам, поэтому включение шаблона хотя бы в одну группу обязательно.

После завершения нажмите *Добавить*. Новый шаблон должен отображаться в списке шаблонов. Также можно использовать фильтр, чтобы найти свой шаблон.



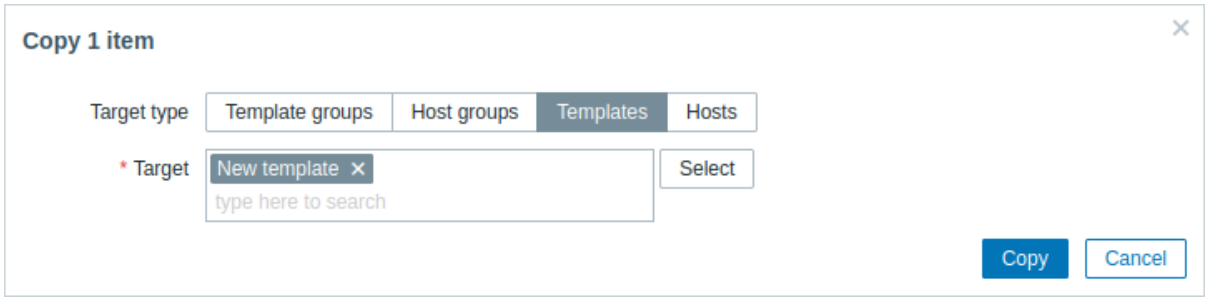

Как видите, шаблон уже есть, но он пока пуст — в нем нет ни элементов данных, ни триггеров, ни других сущностей.

Добавление элемента данных в шаблон

Чтобы добавить элемент данных в шаблон, откройте список элементов данных для 'New host', перейдя в *Сбор данных* → *Узлы сети* и нажав *Элементы данных* рядом с 'New host'.

Затем:

- Отметьте флажок элемента данных 'CPU Load' в списке.
- Нажмите *Копировать* под списком.
- Выберите вкладку *Шаблоны*.
- Выберите шаблон, в который нужно скопировать элемент данных.




Все обязательные поля ввода отмечены красной звездочкой.

- Нажмите *Копировать*.

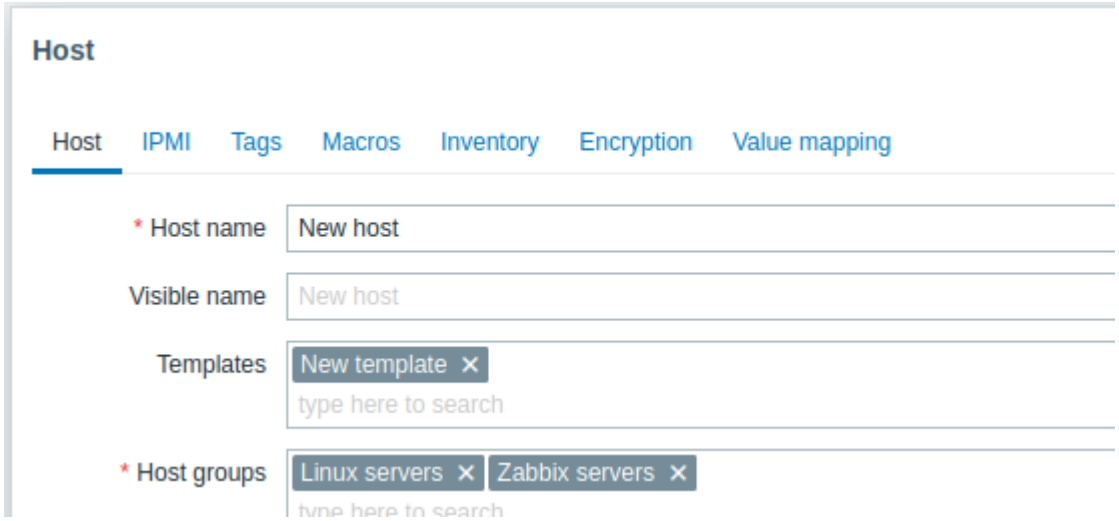
Если теперь перейти в *Сбор данных* > *Шаблоны*, в 'New template' должен появиться один новый элемент данных.

Пока мы остановимся только на одном элементе данных, но аналогичным образом вы можете добавить в шаблон любые другие элементы данных, триггеры или другие сущности, чтобы получить достаточно полный набор сущностей для заданной цели (мониторинг ОС, мониторинг одного приложения).

Привязка шаблона к узлу сети

Когда шаблон готов, остается только добавить его к узлу сети. Для этого перейдите в *Data collection* > *Hosts*, нажмите 'New host', чтобы открыть форму его настройки, и найдите поле **Templates**.

Начните вводить *New template* в поле *Templates*. Имя созданного нами шаблона должно появиться в выпадающем списке. Прокрутите вниз, чтобы выбрать его. Убедитесь, что он появился в поле *Templates*.




Нажмите *Update* в форме, чтобы сохранить изменения. Теперь шаблон добавлен к узлу сети вместе со всеми сущностями, которые он содержит.

Таким образом его можно применить и к любому другому узлу сети. Любые изменения в элементах данных, триггерах и других сущностях на уровне шаблона будут распространяться на узлы сети, к которым этот шаблон привязан.

Соединение предопределённых шаблонов с узлами сети

Как вы могли заметить, Zabbix поставляется с набором предопределённых шаблонов для различных ОС, устройств и приложений. Чтобы начать наблюдение как можно быстрее, вы можете присоединить соответствующий шаблон к узлу сети; но нужно учесть, что эти шаблоны потребуется доработать для вашей среды. Некоторые проверки могут не понадобиться, а интервалы проверок могут быть слишком частыми.

Доступна более подробная информация о шаблонах.

## 6. Готовое решение Zabbix

**Обзор** Appliance Zabbix предоставляет способ мгновенно развернуть сервер Zabbix и веб-интерфейс вместо их ручной настройки или повторного использования существующего сервера для Zabbix.

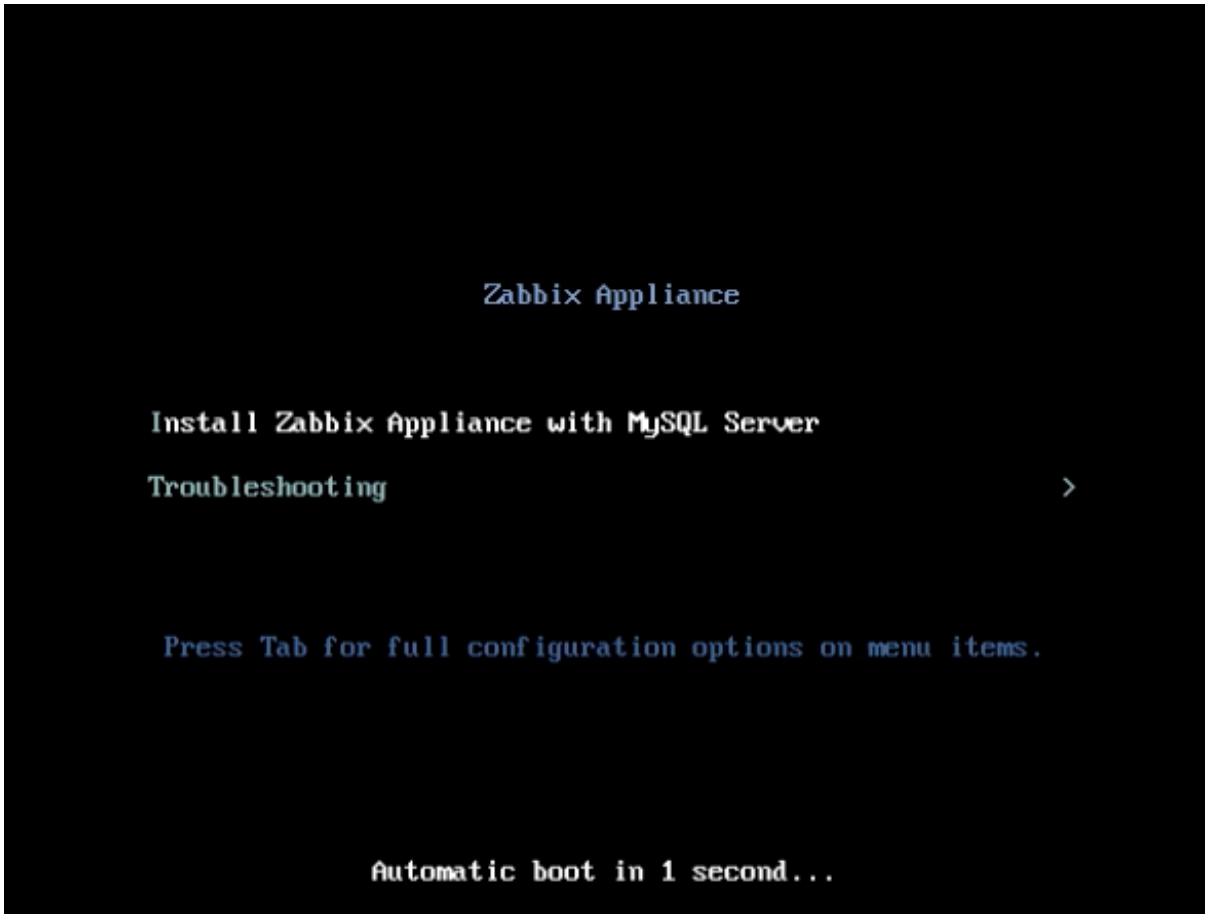
Appliance основан на AlmaLinux 8 (x86_64) и содержит предварительно настроенный сервер Zabbix, работающий на MySQL, а также веб-интерфейс, работающий на веб-сервере Nginx.

> **Attention:**
> Этот appliance предназначен для оценки Zabbix. Использовать его в серьезных производственных средах не рекомендуется.

Образы appliance доступны для загрузки в следующих форматах:

- Установочный CD/DVD (.iso)
- VMware (.vmx) - см. примечания
- Open virtualization format (.ovf)
- Microsoft Hyper-V (.vhd/.vhdx) - см. примечания
- KVM, Parallels, QEMU, USB-накопитель, VirtualBox, Xen (.raw) - см. примечания
- KVM, QEMU (.qcow2)

Меню загрузки установочного CD/DVD Zabbix:



**Быстрый старт** Предварительные требования

1. Убедитесь, что на хост-машине достаточно ресурсов для соответствия системным требованиям виртуальной машины:

- *RAM*: 4 GB
- *Disk space*: не менее 8 GB должно быть выделено для виртуальной машины
- *CPU*: минимум 2 ядра

2. Если программное обеспечение для виртуализации еще не установлено, установите его для загрузки образа appliance (например, VirtualBox).
3. Скачайте appliance в формате, поддерживаемом вашим программным обеспечением для виртуализации.
4. Проверьте сетевые настройки, чтобы убедиться, что на хост-машине включен DHCP.

Установка

1. Загрузите виртуальную машину appliance из скачанного образа. Инструкции см. в документации к вашему программному обеспечению для виртуализации, например, документация VirtualBox.
2. Настройте сетевые параметры виртуальной машины так, чтобы обеспечить доступ из браузера на узле сети. Это можно сделать, включив *режим Bridged*.
3. Войдите в виртуальную машину, используя стандартные системные учетные данные.
4. Чтобы получить IP-адрес, выполните на виртуальной машине следующую команду:

```
ip addr show
```

5. Откройте браузер на узле сети и перейдите по IP-адресу, который appliance получил по DHCP.
6. Войдите в Zabbix, используя стандартные учетные данные, и начните мониторинг.

**Конфигурация** В этом разделе описаны часто требуемые настройки конфигурации по умолчанию, а также доступные параметры настройки.

Учетные данные

Система

- Имя пользователя: root
- Пароль: zabbix

Веб-интерфейс Zabbix

- Имя пользователя: Admin
- Пароль: zabbix

После входа в систему вы можете изменить пароль по умолчанию в настройках профиля пользователя или создать новых пользователей и удалить пользователя по умолчанию.

База данных

Пароли для всех пользователей базы данных случайным образом генерируются в процессе установки. Для базы данных определены следующие пользователи:

*Root:*

- Имя пользователя: root
- Пароль: пароль хранится в файле `/root/.my.cnf`. Вводить пароль для учетной записи root не требуется.

*Сервер Zabbix:*

- Имя пользователя: zabbix_srv
- Пароль: пароль хранится в файле `/etc/zabbix/zabbix\_server.conf`

*Веб-интерфейс Zabbix:*

- Имя пользователя: zabbix_web
- Пароль: пароль хранится в файле `/etc/zabbix/web/zabbix.conf.php`

Чтобы изменить пароль пользователя базы данных, измените его в MySQL и в соответствующем конфигурационном файле.

Доступ к веб-интерфейсу

К веб-интерфейсу Zabbix можно получить доступ по адресу *http://<IP-адрес виртуальной машины>*.

По умолчанию доступ разрешен откуда угодно. Чтобы ограничить доступ, измените `/etc/nginx/conf.d/zabbix.conf`. После сохранения измененного файла перезапустите Nginx, войдя по SSH как пользователь root и выполнив:

```
systemctl restart nginx
```

Статический IP-адрес

По умолчанию устройство использует DHCP для получения IP-адреса. Чтобы задать статический IP-адрес:

- Войдите в систему как пользователь root.
- Выполните следующие команды, заменив значения на ваши собственные IP-адреса:

```
nmcli connection modify eth0 ipv4.addresses 192.168.1.10/24    # IP-         /     CIDR
nmcli connection modify eth0 ipv4.gateway 192.168.1.1     # IP-
nmcli connection modify eth0 ipv4.dns 8.8.8.8    # IP-    DNS-
nmcli connection modify eth0 ipv4.method manual
systemctl restart network
```

В качестве альтернативы вы можете открыть файл `/etc/NetworkManager/system-connections/eth0.nmconnection` и добавить следующие строки:

```
[ipv4]
address1=192.168.1.10/24,192.168.1.1
dns=8.8.8.8
method=manual
```

После сохранения измененного файла выполните команду `systemctl restart network`.

Конфигурация брандмауэра

Для управления настройками брандмауэра устройство использует iptables с предопределенными правилами:

- Открыть порт SSH (22 TCP)
- Открыть порты Zabbix agent (10050 TCP) и Zabbix trapper (10051 TCP)
- Открыть порты HTTP (80 TCP) и HTTPS (443 TCP)
- Открыть порт SNMP trap (162 UDP)
- Разрешить исходящие соединения к порту NTP (123 UDP)
- Ограничить ICMP-пакеты до 5 пакетов в секунду
- Отклонять все остальные входящие соединения

Чтобы открыть дополнительные порты, измените файл `/etc/sysconfig/iptables` и перезагрузите правила брандмауэра:

```
systemctl reload iptables
```

Репозитории

Zabbix appliance использует пакет *zabbix-release* из репозитория Zabbix. Репозитории настраиваются в каталоге `/etc/yum.repos.d/*`.

Часовой пояс

По умолчанию appliance использует UTC для системных часов. Чтобы изменить часовой пояс, скопируйте соответствующий файл из `/usr/share/zoneinfo` в `/etc/localtime`, например:

```
cp /usr/share/zoneinfo/Europe/Riga /etc/localtime
```

**Note:**
Часовой пояс веб-интерфейса Zabbix задается отдельно и может быть изменен в настройках веб-интерфейса. Часовой пояс по умолчанию для веб-интерфейса Zabbix — Europe/Riga.

Расположение файлов

- Файлы конфигурации находятся в `/etc/zabbix`
- Файлы журналов Zabbix server, proxy и agent находятся в `/var/log/zabbix`
- Zabbix frontend находится в `/usr/share/zabbix`
- Домашний каталог пользователя *zabbix* находится в `/var/lib/zabbix`

Системные службы

Доступны службы systemd. Чтобы просмотреть список служб Zabbix, выполните следующую команду на виртуальной машине:

```
systemctl list-units zabbix*
```

**Примечания, специфичные для формата изображения** VMware

Образы в формате *vmdk* можно напрямую использовать в продуктах VMware Player, Server и Workstation. Для использования в ESX, ESXi и vSphere их необходимо преобразовать с помощью VMware vCenter Converter (для загрузки требуется аутентификация). Если вы используете VMware vCenter Converter, могут возникнуть проблемы с гибридным сетевым адаптером. В этом случае можно попробовать указать адаптер E1000 в процессе преобразования. Кроме того, после завершения преобразования можно удалить существующий адаптер и добавить адаптер E1000.

HDD/flash-образ (raw)

Чтобы загрузить образ, выполните:

```
dd if=./zabbix_appliance_7.4.0.raw of=/dev/sdc bs=4k conv=fdatasync
```

Замените `/dev/sdc` на путь к устройству вашего диска Flash/HDD.

Hyper-V

Если appliance не удается запустить в Hyper-V, попробуйте нажать **Ctrl+Alt+F2**, чтобы переключиться на сеанс TTY.

**Устранение неполадок** Если при попытке войти во веб-интерфейс вы видите сообщение об ошибке *Access denied for user 'replace_user'@'localhost' (using password: YES)*, это может означать, что установка все еще выполняется.

Если ошибка сохраняется после ожидания в течение нескольких минут или вы наблюдаете любое другое неожиданное поведение, вероятно, процесс установки не был успешно завершен. В этом случае мы рекомендуем удалить текущий appliance и развернуть его заново, следуя тем же инструкциям по установке. Этот шаг обычно устраняет проблему.

Обратите внимание, что попытки вручную исправить поврежденную установку не рекомендуются, так как это может привести к дальнейшим осложнениям.

## 7. Настройка

**Обзор** Настройка Zabbix включает в себя указание того, какие узлы сети и системы следует отслеживать, определение того, какие данные следует собирать, и установление способа доставки оповещений при возникновении проблем.

Используйте боковое меню для навигации по соответствующим аспектам настройки.

**Ключевые задачи настройки**

- Настройка нового узла сети или группы узлов сети → Узлы сети и группы узлов сети
- Определение того, какие данные собирать с узлов сети → Элементы данных
- Создание правил для обнаружения проблем → Обнаружение проблем с помощью триггеров
- Настройка способов доставки оповещений → Оповещения о событиях
- Отображение данных мониторинга с помощью панелей, графиков и карт сети → Визуализация
- Повторное использование настроек на нескольких узлах сети → Шаблоны и группы шаблонов, Templates out of the box
- Управление доступом и правами пользователей → Пользователи и группы пользователей

### 1. Настройка шаблона

Обзор

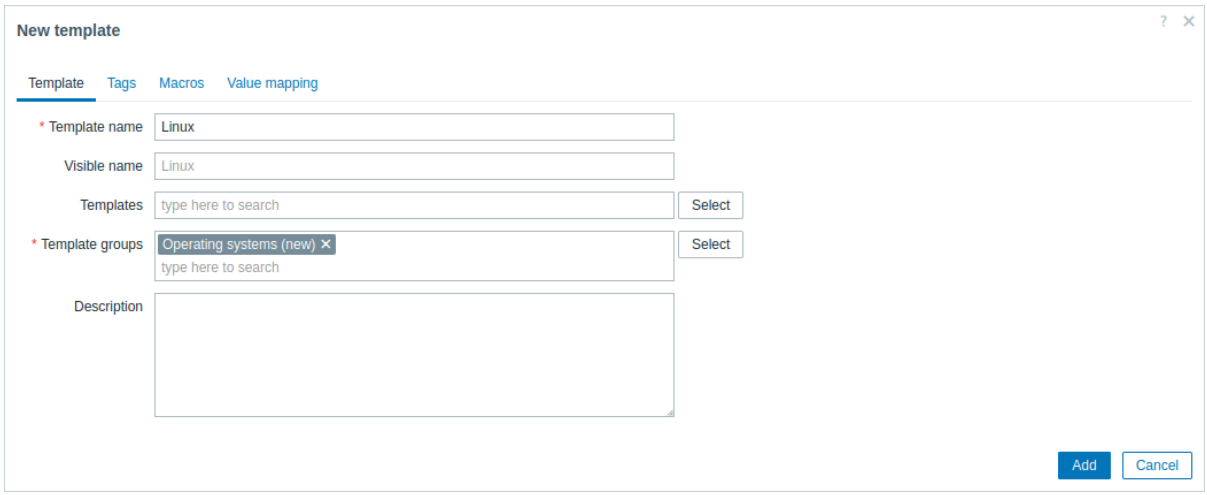
Для настройки шаблона необходимо сначала создать его, указав общие параметры, и только затем добавлять в него объекты (элементы данных, триггеры, графики и прочее).

Создание шаблона

Чтобы создать шаблон, выполните следующие действия:

1. Перейдите в *Сбор данных* → *Шаблоны*.
2. Нажмите *Создать шаблон*.
3. Отредактируйте атрибуты шаблона.

Вкладка **Шаблон** содержит общие атрибуты шаблона.

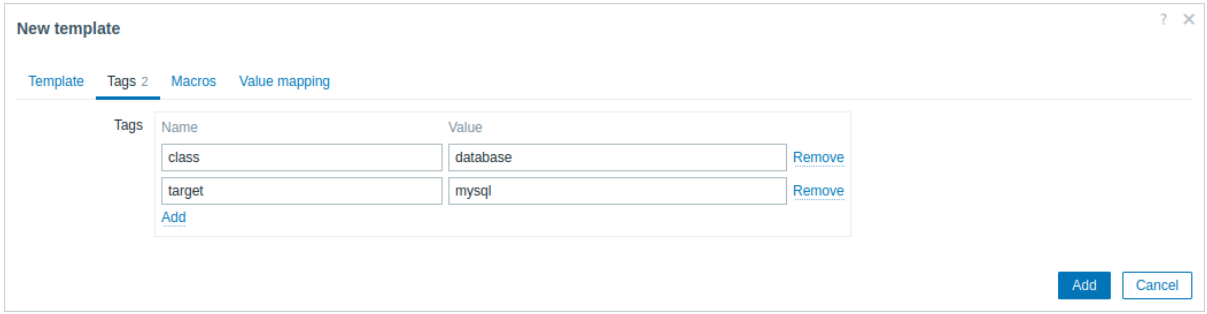



Все обязательные поля ввода отмечены красной звездочкой.

Атрибуты шаблона:

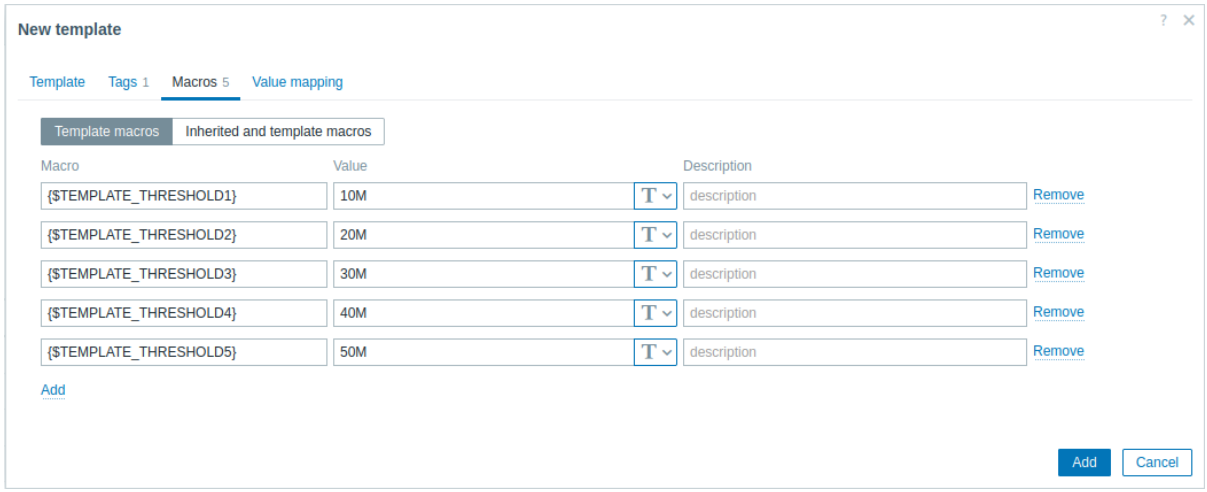
| Parameter | Description |
|---|---|
| *Template name* | Уникальное имя шаблона.<br>Допускаются буквенно-цифровые символы, пробелы, точки, дефисы и символы подчеркивания.<br>Начальные и конечные пробелы не допускаются. |
| *Visible name* | Если задать это имя, оно будет отображаться в списках, картах и т. д. |
| *Templates* | Свяжите с этим шаблоном один или несколько шаблонов. Все сущности (элементы данных, триггеры и т. д.) будут унаследованы от связанных шаблонов.<br>Чтобы связать новый шаблон, введите его имя в поле *Templates* — отобразится выпадающий список подходящих шаблонов. Либо нажмите *Select*, чтобы открыть всплывающее окно *Templates*, в котором изначально не отображаются шаблоны. Чтобы заполнить список, либо введите имя группы шаблонов в поле *Template group* (и выберите один из предложенных вариантов), либо нажмите кнопку *Select* рядом с полем *Template group*, чтобы открыть всплывающее окно *Template groups*. После выбора группы шаблонов окно *Templates* обновится и покажет шаблоны, принадлежащие этой группе. Выбранные шаблоны будут связаны при сохранении или обновлении конфигурации.<br>Чтобы отменить связь шаблона, используйте один из двух вариантов в блоке *Templates*:<br>*Unlink* - отменить связь шаблона, но сохранить его сущности (элементы данных, триггеры и т. д.);<br>*Unlink and clear* - отменить связь шаблона и удалить все его сущности (элементы данных, триггеры и т. д.). |
| *Template groups* | Группы шаблонов, к которым принадлежит шаблон. |
| *Description* | Описание шаблона. |
| *Vendor and version* | Поставщик и версия шаблона; отображается только при обновлении существующих шаблонов (готовые шаблоны, предоставляемые Zabbix, импортированные шаблоны или шаблоны, измененные через Template API), если конфигурация шаблона содержит такую информацию.<br>Не может быть изменено в веб-интерфейсе Zabbix.<br>Для готовых шаблонов версия отображается следующим образом: основная версия Zabbix, разделитель (”-”), номер ревизии (увеличивается с каждой новой версией шаблона и сбрасывается с каждой основной версией Zabbix). Например, 7.0-0, 7.0-5, 7.4-0, 7.4-3. |

Вкладка **Теги** позволяет задавать теги на уровне шаблона tags. Все проблемы узлов сети, связанных с этим шаблоном, будут помечены значениями, указанными здесь.

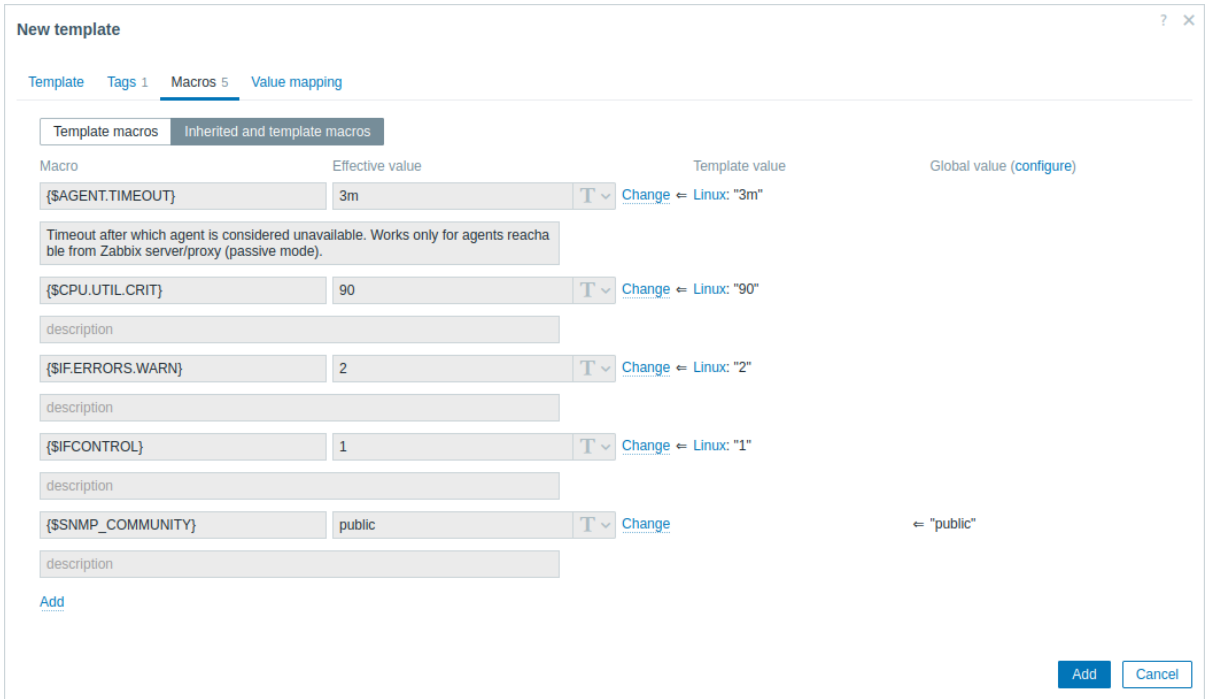



В тегах поддерживаются пользовательские макросы, макросы {INVENTORY.*}, макросы {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} и {HOST.ID}.

Вкладка **Макросы** позволяет задавать пользовательские макросы на уровне шаблона в виде пар имя-значение. Обратите внимание, что значения макросов могут храниться как обычный текст, секретный текст или секрет Vault. Также поддерживается добавление описания.




Если выбрать опцию *Inherited and template macros*, можно также просматривать макросы из связанных шаблонов и глобальные макросы, которые будет наследовать шаблон, а также значения, к которым будут разрешаться макросы.




Для удобства предоставляются ссылки на соответствующие шаблоны, а также ссылка на настройку глобальных макросов. Также можно редактировать макрос связанного шаблона или глобальный макрос на уровне шаблона, фактически создавая копию макроса в шаблоне.

Вкладка **Преобразование значений** позволяет настроить удобное для человека представление данных элемента данных в преобразованиях значений.

Кнопки:

| | |
|---|---|
| Add | Добавить шаблон. Добавленный шаблон должен появиться в списке. |
| Update | Обновить свойства существующего шаблона. |
| Clone | Создать еще один шаблон на основе свойств текущего шаблона. Это **включает** сущности (элементы данных, триггеры и т. д.), унаследованные от связанных шаблонов, а также напрямую привязанные к текущему шаблону, но **исключает** поставщика и версию текущего шаблона, чтобы клонированный шаблон можно было отличить от исходного. |
| Delete | Удалить шаблон; сущности шаблона (элементы данных, триггеры и т. д.) останутся у связанных узлов сети. |
| Delete and clear | Удалить шаблон и все его сущности у связанных узлов сети. |
| Cancel | Отменить редактирование свойств шаблона. |

Добавление элементов данных, триггеров, графиков

**Attention:**
Элементы данных сначала нужно добавить в шаблон. Триггеры и графики нельзя добавить без соответствующего элемента данных.

Есть два способа добавить элементы данных в шаблон:

1. Чтобы создать новые элементы данных, следуйте рекомендациям в разделе Создание элемента данных.
2. Чтобы добавить существующие элементы данных в шаблон:

- Перейдите в *Сбор данных* > *Узлы сети* (или *Шаблоны*).
- Нажмите *Элементы данных* в строке нужного узла сети/шаблона.
- Отметьте флажками элементы данных, которые вы хотите добавить в шаблон.
- Нажмите *Копировать* под списком элементов данных.
- Выберите шаблон (или группу шаблонов), в который нужно скопировать элементы данных, и нажмите *Копировать*.<br>
  Все выбранные элементы данных должны быть скопированы в шаблон.

Добавление триггеров и графиков выполняется аналогичным образом (из списков триггеров и графиков соответственно), при этом также следует помнить, что их можно добавить только после добавления необходимых элементов данных.

Добавление панелей мониторинга

Чтобы добавить панели мониторинга в шаблон в *Сбор данных* → *Шаблоны*, выполните следующее:

1. Нажмите *Панели мониторинга* в строке шаблона.
2. Настройте панель мониторинга, следуя рекомендациям по настройке панелей мониторинга.

**Attention:**
При настройке виджетов на панели мониторинга шаблона (вместо глобальной панели мониторинга) параметры, связанные с узлом сети, недоступны, а у некоторых параметров отличается название. Это связано с тем, что панели мониторинга шаблона отображают данные только с узла сети, к которому привязан шаблон. Например, параметры *Host groups*, *Exclude host groups* и *Hosts* в виджете *Problems* недоступны, параметр *Host groups* в виджете *Host availability* недоступен, а параметр *Show hosts in maintenance* переименован в *Show data in maintenance* и т. д. Дополнительные сведения о доступности параметров в виджетах панели мониторинга шаблона см. в разделе о конкретных параметрах каждого виджета панели мониторинга.

**Note:**
Подробности о доступе к панелям мониторинга узлов сети, созданным на основе панелей мониторинга шаблона, см. в разделе панели мониторинга узлов сети.

Настройка правил низкоуровневого обнаружения

Смотрите раздел по низкоуровневому обнаружению в этом руководстве.

Добавление веб-сценариев

Чтобы добавить веб-сценарии в шаблон в *Сбор данных* → *Шаблоны*, выполните следующие действия:

1. Нажмите *Web* в строке шаблона.
2. Настройте веб-сценарий обычным способом, как описано в настройке веб-сценария.

**2 Настройка группы шаблонов**

Обзор

Группы шаблонов используются для логической группировки шаблонов и назначения им прав пользователей.

Каждый шаблон должен быть назначен как минимум в одну группу шаблонов. Шаблон может принадлежать нескольким группам шаблонов, и каждая группа шаблонов может содержать несколько шаблонов.

Обратите внимание, что в Zabbix все права основаны на группах: группах пользователей, группах узлов сети и группах шаблонов. Поэтому даже если одному пользователю нужен доступ к одному шаблону, он предоставляется путем добавления пользователя в группу пользователей, которая имеет право доступа к группе шаблонов, содержащей этот шаблон.

Конфигурация

> **Attention:**
> Только пользователи с ролью Super admin могут создавать группы шаблонов.

В веб-интерфейсе Zabbix есть два способа создания группы шаблонов.

**Вариант 1**:

- Перейдите в: *Сбор данных* → *Группы шаблонов*
- Нажмите *Создать группу шаблонов* в правом верхнем углу экрана
- Введите имя группы в форме

**Вариант 2**: при настройке шаблона введите несуществующее имя группы в поле *Группы шаблонов*.

После создания группы шаблонов вы можете нажать на имя шаблона в списке в разделе *Сбор данных* → *Группы шаблонов*, чтобы изменить имя группы, клонировать группу или удалить ее.

Удаление группы шаблонов удаляет только логическую группу, но не шаблоны в этой группе. Невозможно удалить группу шаблонов, если она является единственной группой для какого-либо существующего шаблона.

Создание подгрупп шаблонов

Подгруппа шаблонов (или вложенная группа шаблонов) является дочерней по отношению к родительской группе шаблонов, которая ее содержит.

Подгруппа создается с помощью символа косой черты '/' в поле ввода имени группы, чтобы обозначить ее отношение к родительской группе(ам). Например:

- ввод `Linux servers/Databases` создает подгруппу `Linux servers/Databases` в родительской группе `Linux servers`.
- ввод `Linux servers/Databases/MySQL/Tokyo` создает соответствующую подгруппу во вложенных родительских группах `Linux servers`, `Linux servers/Databases`, `Linux servers/Databases/MySQL`.

При создании подгруппы не допускается использование начальных или конечных косых черт, а также нескольких косых черт подряд. Экранирование '/' не поддерживается.

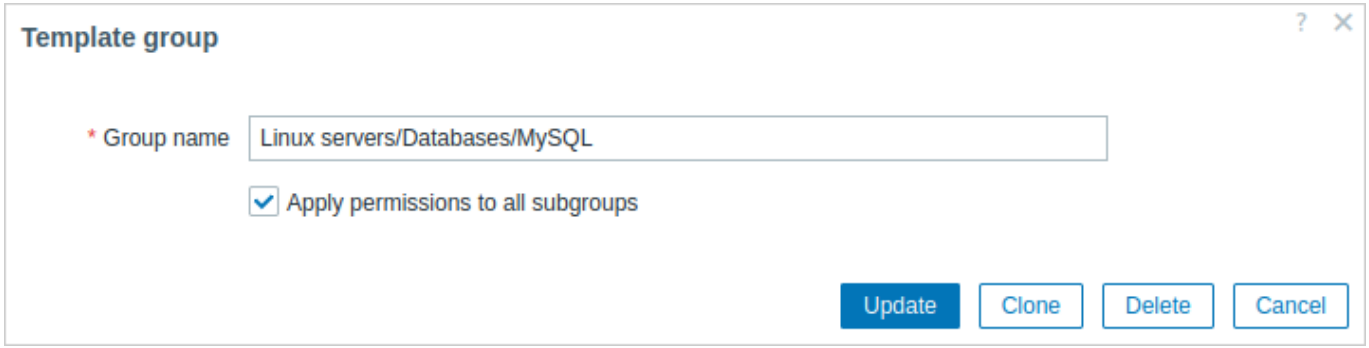
Перед созданием подгруппы не требуется предварительно создавать родительскую группу(ы) шаблонов. Вы можете начать либо с создания подгруппы (например, `Linux servers/Databases`), либо с любой родительской группы шаблонов

(в нашем примере, `Linux servers`). Если вы начнете с создания подгруппы, родительская группа(ы) шаблонов **не** будут созданы автоматически.

Права доступа к группам шаблонов

- При создании подгруппы в существующей родительской группе шаблонов (например, при создании `Linux servers/Databases`, когда `Linux servers` уже существует), группа пользователей наследует права доступа к подгруппе от родительской группы.
- При создании родительской группы шаблонов для уже существующей подгруппы (например, при создании `Linux servers`, когда `Linux servers/Databases` уже существует), права доступа к родительской группе не задаются.

При редактировании любой группы шаблонов можно также задать дополнительную опцию *Применить права доступа ко всем подгруппам*.




Если установить этот флажок и нажать *Обновить*, к текущим и будущим подгруппам редактируемой группы шаблонов будут применены те же права доступа.

Таким образом, если для подгрупп редактируемой группы шаблонов каким-либо группам пользователей были назначены разные права доступа, установка флажка предоставит всем текущим и будущим подгруппам те же права пользователей, что и у редактируемой группы.

Обратите внимание, что эта опция не сохраняется в базе данных и переопределяет существующие права доступа. Любые изменения, внесенные с помощью этой опции, можно отменить только вручную.

### 3. Присоединение/отсоединение

Обзор

Присоединением называется процесс, при котором шаблоны применяются к узлам сети, тогда как отсоединение — удаление связи шаблона с узлом сети.

Привязка шаблона

Чтобы привязать шаблон к узлу сети, выполните следующее:

1. Перейдите в *Сбор данных* → *Узлы сети*.
2. Щелкните нужный узел сети.
3. Начните вводить имя шаблона в поле *Шаблоны*. Появится список подходящих шаблонов; прокрутите вниз, чтобы выбрать нужный. Кроме того, можно нажать *Выбрать* рядом с полем и выбрать один или несколько шаблонов из списка во всплывающем окне.
4. Нажмите *Добавить/Обновить* в форме атрибутов узла сети.

Теперь у узла сети будут все сущности шаблона. Это включает элементы данных, триггеры, графики, правила низкоуровневого обнаружения, веб-сценарии, а также панели мониторинга.

**Attention:**
Привязка нескольких шаблонов к одному и тому же узлу сети завершится неудачей, если эти шаблоны содержат элементы данных с одинаковым ключом элемента данных. И поскольку триггеры и графики используют элементы данных, их тоже нельзя привязать к одному узлу сети из нескольких шаблонов, если используются одинаковые ключи элементов данных.

Когда сущности (элементы данных, триггеры и т. д.) добавляются из шаблона:

- ранее существовавшие одинаковые сущности на узле сети обновляются как сущности шаблона, и **все существующие пользовательские настройки сущности на уровне узла сети будут потеряны**;
- сущности из шаблона добавляются;

- все напрямую привязанные сущности, которые до привязки шаблона существовали только на узле сети, остаются без изменений.

В списках все сущности из шаблона теперь имеют префикс в виде имени шаблона, что указывает на их принадлежность к конкретному шаблону. Само имя шаблона (серым текстом) является ссылкой, позволяющей перейти к списку этих сущностей на уровне шаблона.

**Note:**
Для некоторых элементов данных, таких как внешние проверки, проверки агента HTTP, простые проверки, проверки SSH и проверки Telnet, интерфейс узла сети необязателен. Если на момент привязки шаблона у узла сети не определен интерфейс, эти элементы данных будут добавлены без интерфейса. Если позже вы добавите интерфейс узла сети, он не будет автоматически назначен уже существующим элементам данных. Чтобы назначить недавно добавленный интерфейс узла сети всем элементам данных шаблона сразу, отвяжите шаблон от узла сети, а затем привяжите его снова. Чтобы сохранить историю элемента данных, используйте опцию *Отвязать*, не используйте *Отвязать и очистить*.

Если какая-либо сущность не имеет префикса в виде имени шаблона, это означает, что она существовала на узле сети до этого и не была добавлена шаблоном.

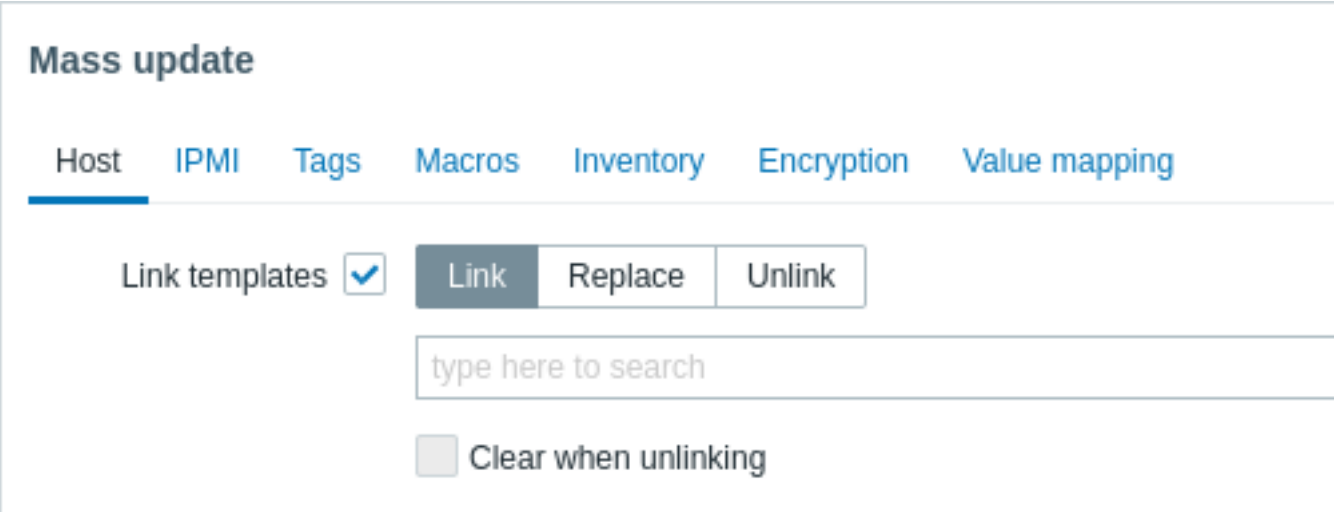
Критерии уникальности объекта

При добавлении объектов (элементов данных, триггеров и т.п.) из шаблона важно знать: какие из этих объектов уже существуют у узла сети и должны быть обновлены, а какие объекты различаются. Критериями уникальности при принятии решения на сходство/различия являются:

- для элементов данных — ключ элемента данных;
- для триггеров — имя и выражение триггера;
- для пользовательских графиков — имя графика и его элементы данных.

Привязка шаблонов к нескольким узлам сети

Чтобы обновить привязку шаблонов для многих узлов сети, в *Сбор данных* → *Узлы сети* выберите нужные узлы сети, отметив их флажками, затем нажмите **Массовое обновление** под списком и выберите *Привязать шаблоны*:




Чтобы привязать дополнительные шаблоны, начните вводить имя шаблона в поле автодополнения, пока не появится выпадающий список с подходящими шаблонами. Просто прокрутите список вниз и выберите шаблон для привязки.

Параметр *Заменить* позволит привязать новый шаблон, одновременно отвязав любой шаблон, который был ранее привязан к узлам сети. Параметр *Отвязать* позволит указать, какие шаблоны нужно отвязать. Параметр *Очистить при отвязке* позволит не только отвязать ранее привязанные шаблоны, но и удалить все унаследованные от них сущности (элементы данных, триггеры и т. д.).

**Note:**
В Zabbix доступен большой набор предопределенных шаблонов. Вы можете использовать их как пример, но не рекомендуется применять их без изменений в рабочей среде, так как они могут содержать слишком много элементов данных и слишком часто опрашивать данные. Если вы все же хотите использовать их, настройте их под свои реальные потребности.

Изменение соединённых объектов

Если вы попытаетесь изменить элемент данных или триггер, который присоединён из шаблона, вы можете увидеть, что многие ключевые опции недоступны для редактирования. Такой подход имеет смысл, так как идея шаблонов — чтобы изменять объекты в одно касание на уровне шаблона. Тем не менее, вы всё ещё можете, например, активировать/деактивировать элемент данных на отдельном узле сети, указать интервал обновления, срок хранения истории и некоторые другие параметры.

Если вы хотите изменить полностью объект, вы должны отредактировать его на уровне шаблона (ссылка на уровень шаблона отображается в имени диалога), но имейте ввиду, что все эти изменения будут влиять на все узлы сети, которые соединены с этим шаблоном.

> **Attention:**
> Любые настройки объектов, реализованные на уровне шаблона, перезапишут ранее сделанные настройки этих объектов на уровне узлов сети.

Отвязка шаблона

Чтобы отвязать шаблон от узла сети, выполните следующие действия:

1. Перейдите в *Сбор данных* → *Узлы сети*.
2. Нажмите на нужный узел сети и найдите поле *Шаблоны*.
3. Нажмите *Отвязать* или *Отвязать и очистить* рядом с шаблоном, чтобы отвязать шаблон.
4. Нажмите *Обновить* в форме атрибутов узла сети.

При выборе опции *Отвязать* будет просто удалена связь с шаблоном, при этом все его сущности останутся у узла сети. Это включает элементы данных, триггеры, графики, правила обнаружения низкого уровня и веб-сценарии, но не включает панели мониторинга. Обратите внимание, что карты значений и теги, унаследованные от связанных шаблонов, также будут удалены.

При выборе опции *Отвязать и очистить* будут удалены и связь с шаблоном, и все его сущности (элементы данных, триггеры и т. д.).

#### 4. Наследование

Обзор

Наследование — это способ одного шаблона включать в себя один или несколько других шаблонов.

Иногда имеет смысл разделить объекты отдельных шаблонов на различные сервисы, приложения и прочее, в итоге у вас может получиться несколько шаблонов, которые нужно присоединить к довольно большому количеству узлов сети. Для простоты картины, возможно вложить в один шаблон несколько других шаблонов.

Преимущества наследования в том, что вы должны будете присоединить к узлу сети только один шаблон («основной», родительский шаблон), и этот узел сети автоматически унаследует все объекты из присоединённых шаблонов («вложенные», дочерние шаблоны). Например, если мы присоединим шаблоны *T1* и *T2* к шаблону *T3*, мы дополняем *T3* объектами из *T1* и *T2*, но не наоборот. Если мы присоединим шаблон *T1* к шаблонам *T2* и *T3*, мы дополняем шаблоны *T2* и *T3* объектами из *T1*.

Настройка вложенных шаблонов

Чтобы связать шаблоны, необходимо взять существующий шаблон (или создать новый), а затем:

1. Откройте форму настройки шаблона.
2. Найдите поле *Templates*.
3. Нажмите *Select*, чтобы открыть всплывающее окно *Templates*.
4. Во всплывающем окне выберите нужные шаблоны, а затем нажмите *Select*, чтобы добавить шаблоны в список.
5. Нажмите *Add* или *Update* в форме настройки шаблона.

Таким образом, все сущности настроенного шаблона, а также все сущности связанных шаблонов, теперь будут отображаться в настройке шаблона. Это включает элементы данных, триггеры, графики, правила обнаружения низкого уровня и веб-сценарии, но не включает дашборды. Однако дашборды связанных шаблонов, тем не менее, будут унаследованы узлами сети.

Чтобы отвязать любой из связанных шаблонов, нажмите *Unlink* или *Unlink and clear* в форме настройки шаблона, а затем нажмите *Update*.

Опция *Unlink* просто удалит связь с связанным шаблоном, не удаляя при этом ни одну из его сущностей (элементы данных, триггеры и т. д.).

Опция *Unlink and clear* удалит как связь с связанным шаблоном, так и все его сущности (элементы данных, триггеры и т. д.).

### 5. Массовое обновление

Обзор

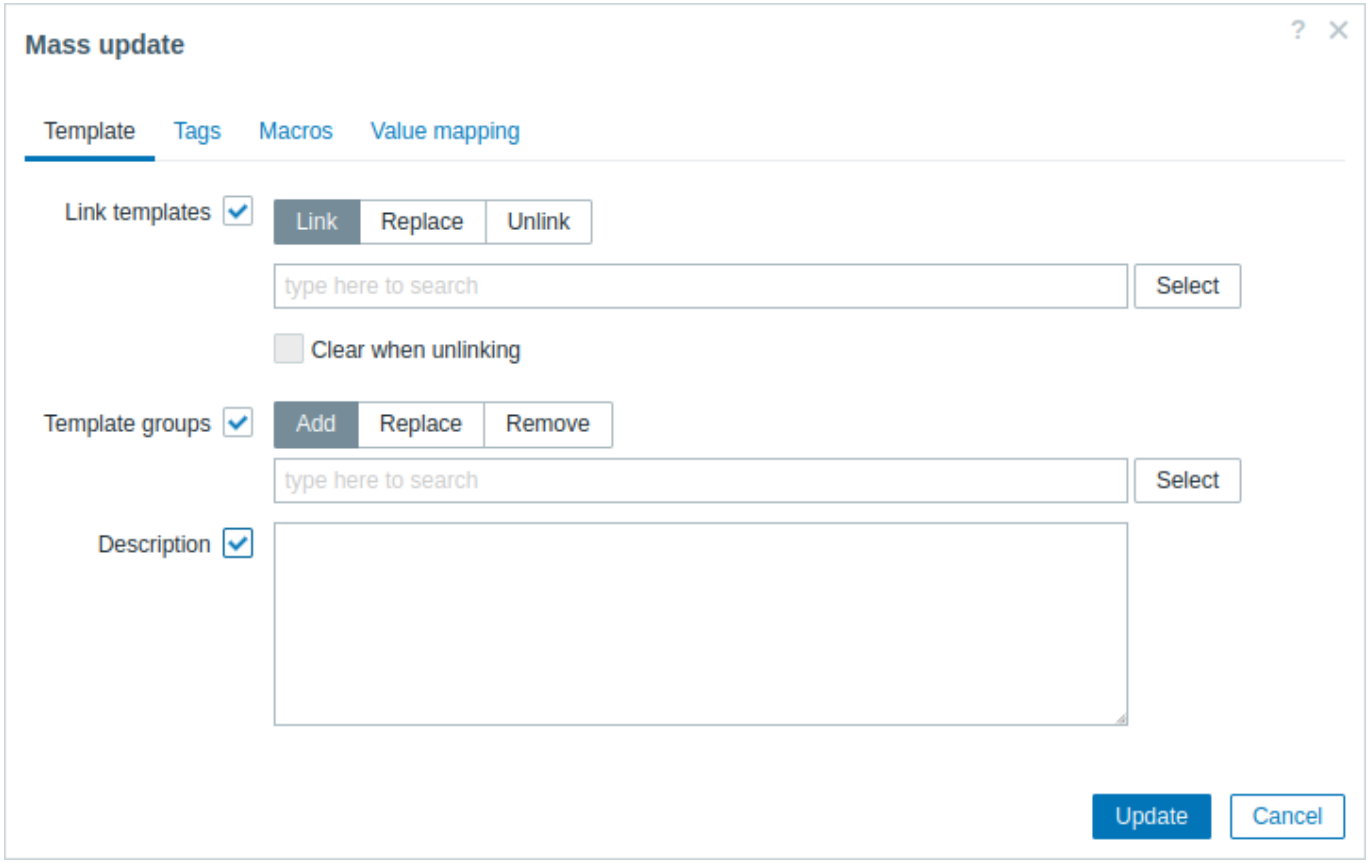
Иногда вы можете захотеть изменить какой-либо атрибут для нескольких шаблонов одновременно. Вместо того, чтобы открывать каждый отдельный шаблон для редактирования, вы можете воспользоваться функцией массового обновления.

Использование массового обновления

Чтобы массово обновить некоторые шаблоны, выполните следующие действия:

1. Отметьте флажки перед шаблонами, которые вы хотите обновить, в списке шаблонов.
2. Нажмите *Массовое обновление* под списком.
3. Перейдите на вкладку с нужными атрибутами (*Шаблон*, *Теги*, *Макросы* или *Преобразование значений*).
4. Отметьте флажки у любых атрибутов, которые нужно обновить, и введите для них новое значение.

Вкладка **Шаблон (Template)** содержит общие параметры массового обновления шаблонов.




Для обновления параметра *Присоединить шаблоны (Link templates)* доступны следующие опции, при выборе соответствующей кнопки:

- *Связь (Link)* — укажите, какие дополнительные шаблоны присоединить.
- *Замена (Replace)* — укажите, какие шаблоны необходимо присоединить, в то время как любые ранее присоединённые шаблоны будут отсоединены.
- *Отсоединить (Unlink)* — укажите, какие шаблоны необходимо отсоединить.

Чтобы указать шаблоны для присоединения/отсоединения, начните печатать имя шаблона в поле с автодополнением до появления всплывающего списка с соответствующими шаблонами. Прокрутите список и выберите необходимый шаблон.

Опция *Очистить при отсоединении (Clear when unlinking)* позволяет не только отсоединить ранее присоединённые шаблоны, но также и удалить все элементы, унаследованные от этих шаблонов (элементы данных, триггеры, графики и т.п.).

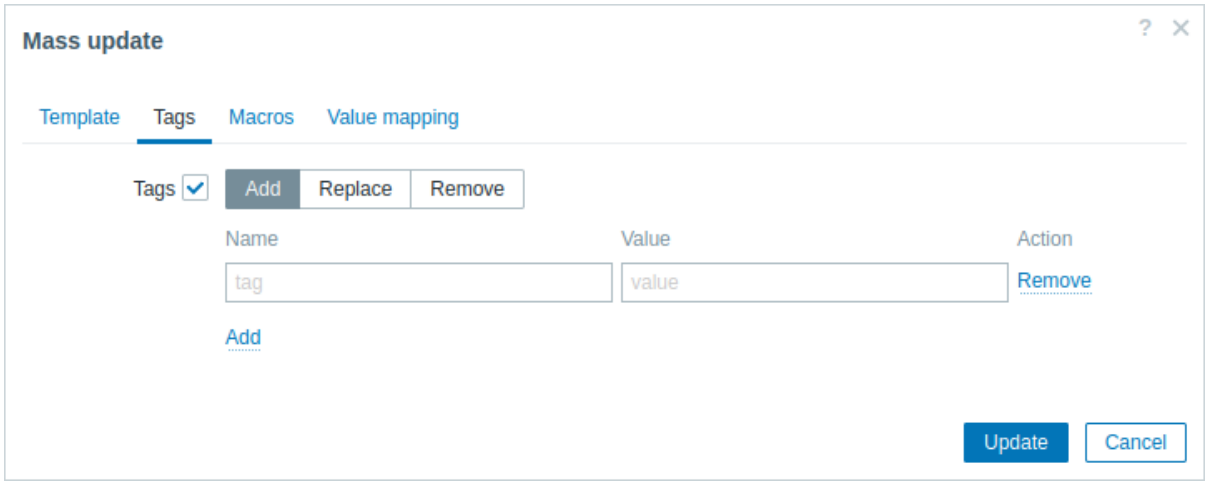
Для обновления параметра *Группы шаблонов (Template groups)* доступны следующие опции, при выборе соответствующей кнопки:

- *Добавить (Add)* — позволяет добавить к шаблонам дополнительные группы шаблонов из существующих или ввести совершенно новые группы шаблонов.
- *Замена (Replace)* — удаляет шаблон из существующих групп шаблонов и заменяет их указанными в этом поле (существующими или новыми группами шаблонов).

- *Удалить (Remove)* — удаляет с шаблонов указанные группы шаблонов.

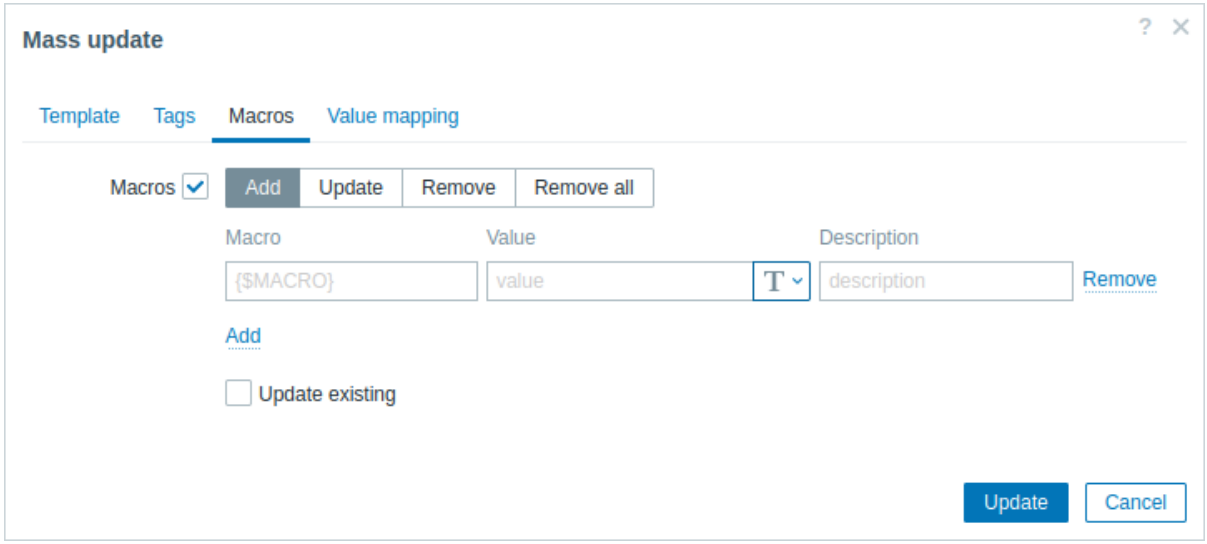
Эти поля имеют функцию автодополнения — начните набирать, и в них появится выпадающий список совпадающих групп шаблонов. Если группа шаблонов новая, она также появится в выпадающем списке и будет отмечена в конце строки как *(новая)*. Просто прокрутите вниз для выбора.

Вкладка **Теги (Tags)** позволяет вам массово обновлять теги на уровне шаблона.




В тегах поддерживаются пользовательские макросы, макросы {INVENTORY.*}, а также {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} и {HOST.ID}. Обратите внимание, теги с одинаковыми именами, но с разными значениями, не считаются «дубликатами», и их можно добавлять на один шаблон.

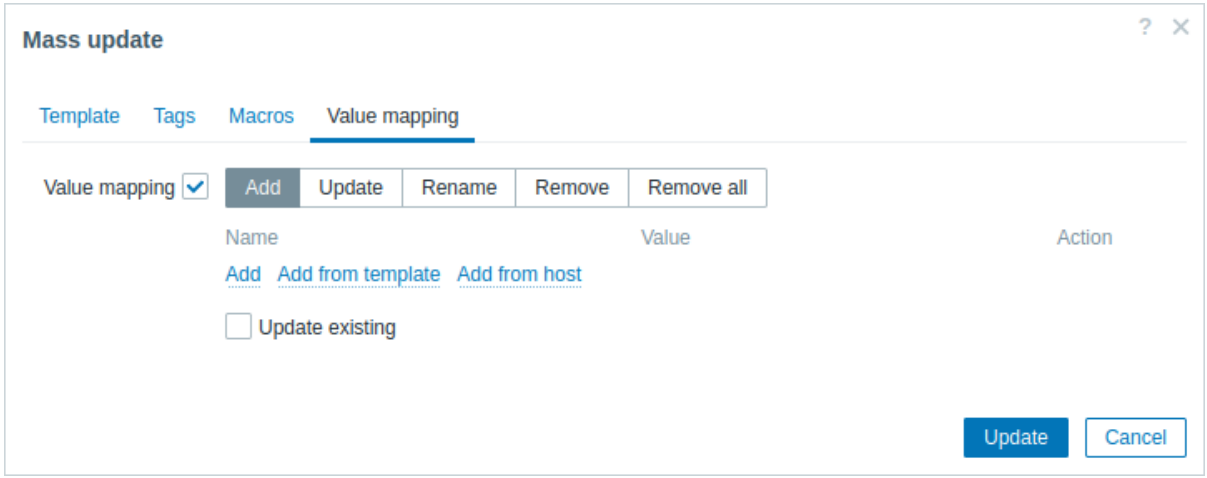
Вкладка **Макросы (Macros)** позволяет вам массово обновлять макросы на уровне шаблона.




Для обновления макросов доступны следующие опции, при выборе соответствующей кнопки:

- *Добавить (Add)* — позволяет задать дополнительные пользовательские макросы для шаблонов. Если выбрана опция *Обновить существующее (Update existing)*, тогда значение, тип и описание указанного имени макроса будут обновлены. Если не выбрана, в случае если макрос с таким же именем уже существует у шаблона, он не будет обновлён.
- *Обновить (Update)* — заменит значения, типы и описания макросов, указанных в списке. Если выбрана опция *Добавить отсутствующее (Add missing)*, тогда макрос, который ранее не существовал у шаблона, будет добавлен как новый. Если не выбрана, будут обновлены только те макросы, которые уже существуют в шаблоне.
- *Удалить (Remove)* — удалит указанные макросы из шаблонов. Если выбрана опция *Исключая выбранные (Except selected)*, тогда все макросы, за исключением выбранных в списке, будут удалены. Если не выбрана, будут удалены только те макросы, что указаны в списке.
- *Удалить все (Remove all)* — удалит все пользовательские макросы из шаблонов. Если опция *Я подтверждаю удаление всех макросов (I confirm to remove all macros)* не выбрана, новое всплывающее окно запросит подтверждение удаления всех макросов.

Вкладка **Преобразование значений (Value mapping)** позволяет вам массово обновлять преобразования значений.




Для обновления преобразований значений доступны кнопки со следующими опциями:

- *Добавить (Add)* — добавление преобразований значений к шаблонам. Если выбрана опция *Обновить существующее (Update existing)*, будут обновлены все свойства преобразования значений с указанным именем. В противном случае, если преобразование значений с таким именем уже существует, оно не будет обновлено.
- *Обновить (Update)* — обновление существующих преобразований значений. Если выбрана опция *Добавить отсутствующее (Add missing)*, тогда преобразование значений, которое ранее не существовало у шаблона, будет добавлено как новое. В противном случае будут обновлены только те преобразования значений, которые уже существуют в шаблоне.
- *Переименовать (Rename)* — задать новое имя существующему преобразованию значений
- *Удалить (Remove)* — удаление указанных преобразований значений из шаблонов. Если выбрана опция *Исключая выбранные (Except selected)*, будут удалены все преобразования значений **за исключением** тех, которые выбраны.
- *Удалить все (Remove all)* — удаление всех преобразований значений из шаблонов. Если опция *Я подтверждаю удаление всех преобразований значений (I confirm to remove all value maps)* не выбрана, новое всплывающее окно запросит подтверждение удаления всех преобразований значений.

Параметры *Добавить из шаблона (Add from template)* и *Добавить с узла сети (Add from host)* доступны для операций добавления/обновления преобразования значений. Они позволяют выбирать преобразования значений из шаблона или узла сети соответственно.

Когда все необходимые изменения будут внесены, нажмите *Update*. Атрибуты будут соответствующим образом обновлены для всех выбранных шаблонов.

**1. Узлы сети и группы узлов сети**

Что такое ”узел сети”?

В Zabbix ”узел сети” — это любое физическое или виртуальное устройство, приложение, служба или любая другая логически связанная совокупность контролируемых параметров.

Создание узлов сети — одна из первых задач мониторинга в Zabbix. Например, если вы хотите отслеживать некоторые параметры на сервере ”x”, сначала необходимо создать узел сети, например ”Server X”, а затем можно добавить к нему элементы данных для мониторинга.

Узлы сети организуются в группы узлов сети.

Перейдите к:

- Мастер узлов сети для пошагового интерфейса, который поможет создать и настроить новый или существующий узел сети.
- Настройка узла сети для классического способа создания и настройки узла сети.

**1 Мастер узла сети**

Обзор

Мастер узлов сети — это пошаговый интерфейс с подсказками для настройки объекта мониторинга (устройства, приложения, службы и т. д.) в Zabbix. Он проведет вас через следующие шаги:

- Выбор шаблона

- Создание или выбор узла сети
- Установка агента Zabbix или agent 2
- Добавление интерфейса узла сети
- Применение дополнительной конфигурации к объекту мониторинга или узлу сети Zabbix (если это требуется шаблоном)

Чтобы открыть Мастер узлов сети в веб-интерфейсе Zabbix, выполните следующие действия:

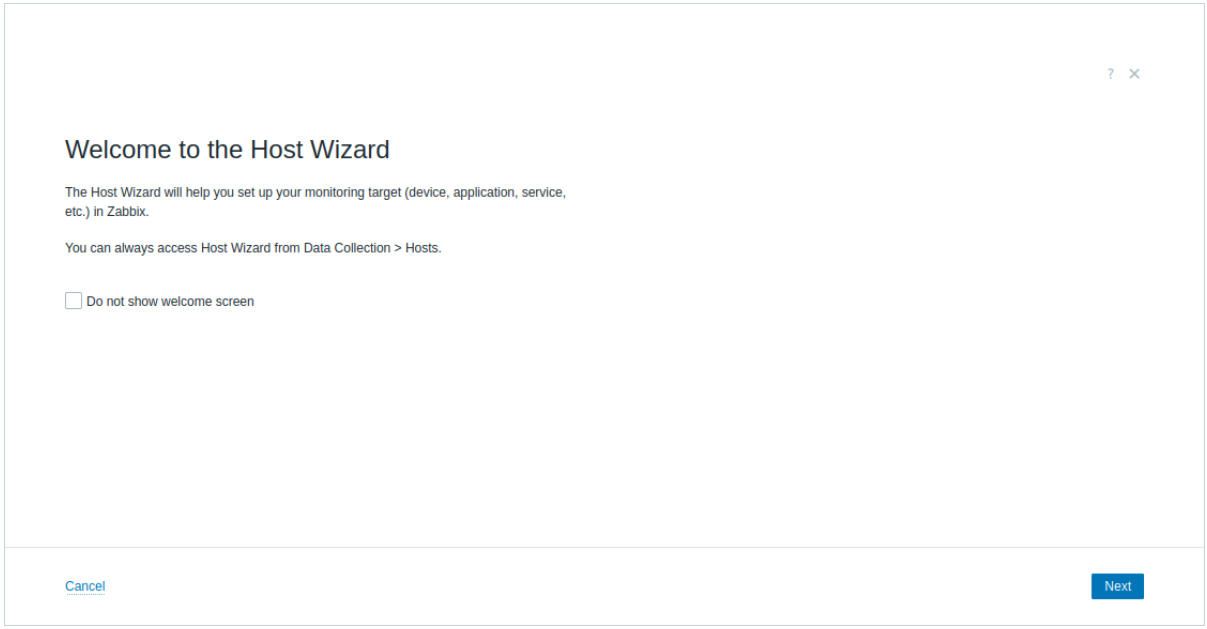
- Перейдите в: *Сбор данных* > *Узлы сети*
- Нажмите *Мастер узлов сети* в правом верхнем углу экрана
- Следуйте инструкциям мастера

Вы также можете изменить существующий узел сети, щелкнув по его имени (или по значку с тремя точками рядом с ним) в различных разделах веб-интерфейса, чтобы открыть Мастер узлов сети из меню узла сети. Обратите внимание, что некоторые шаги пропускаются при редактировании существующего узла сети, если он уже включает конфигурацию, требуемую выбранным шаблоном.

В качестве альтернативы вы можете использовать классический способ создания и настройки узла сети.

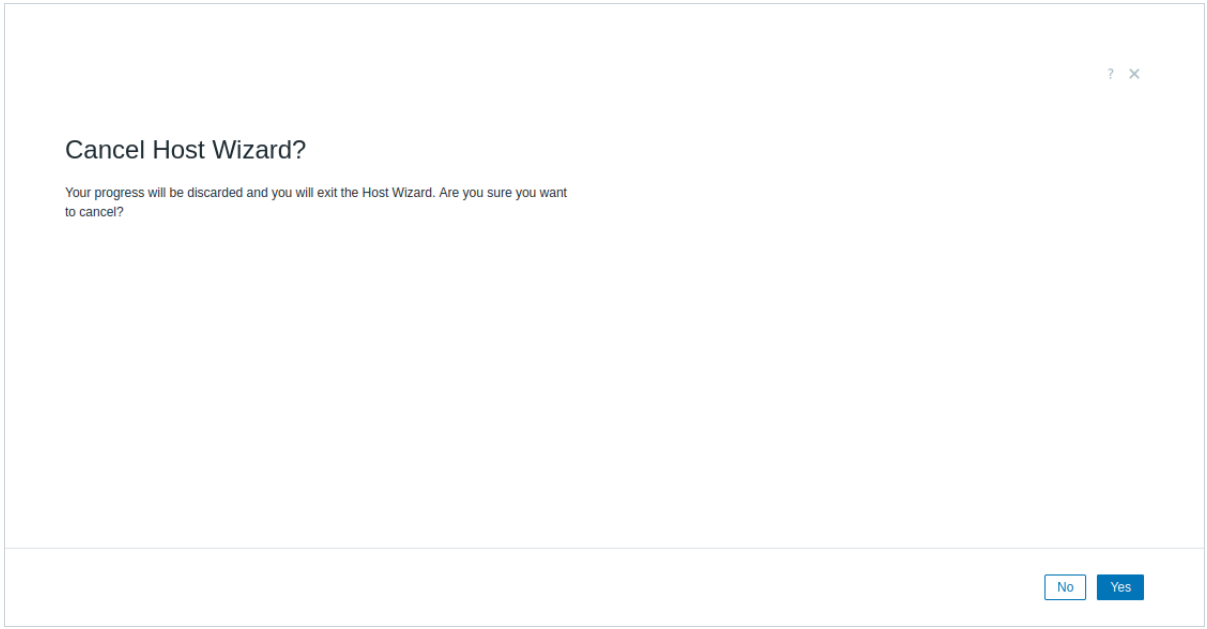
Добро пожаловать в мастер узла сети

Когда вы запускаете мастер узла сети, отображается экран приветствия:




Чтобы в будущем пропускать этот экран, вы можете установить флажок *Do not show welcome screen* и нажать *Next*.

Если вы начнете настраивать параметры на следующих шагах мастера узла сети и попытаетесь закрыть мастер, нажав значок закрытия или клавишу ESC, появится экран подтверждения:




Нажатие *Yes* завершит работу мастера узла сети без сохранения изменений. Нажатие *No*, ESC или значка закрытия вернет вас к последнему шагу.

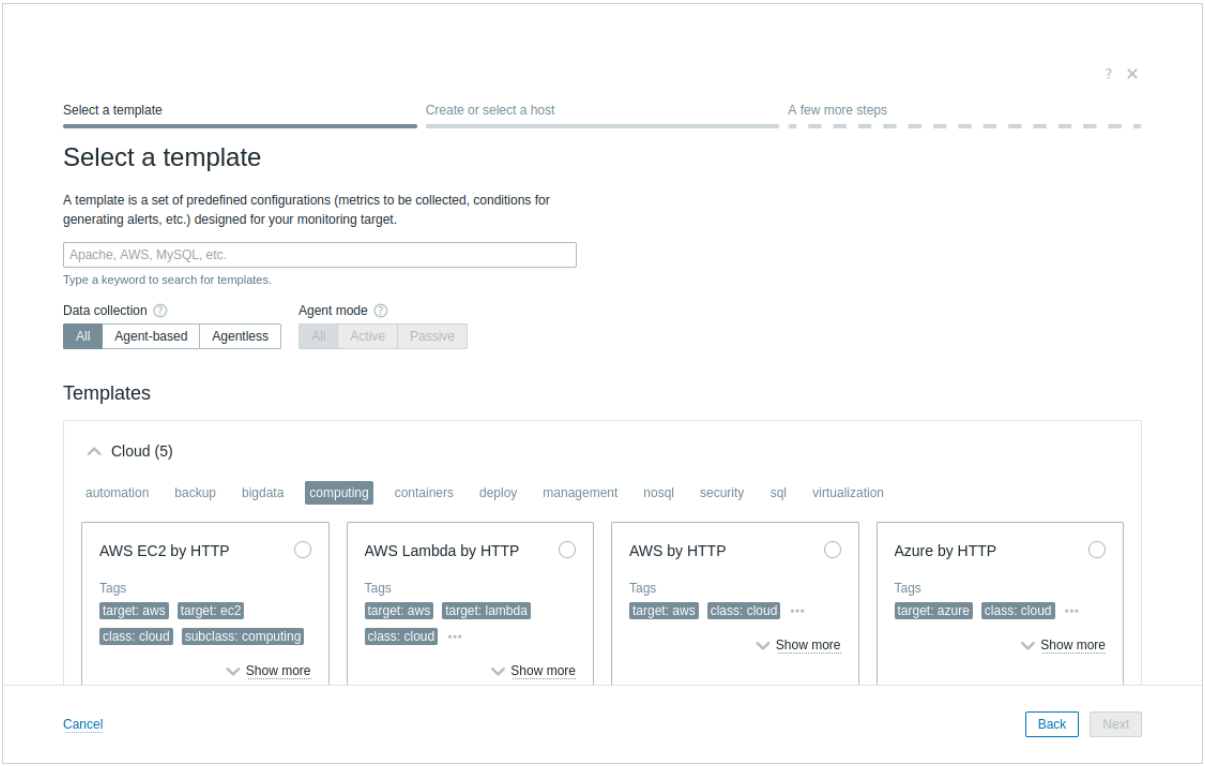
Выберите шаблон

Первый шаг настройки объекта мониторинга — выбрать шаблон — набор заранее определенных конфигураций (метрики для сбора, условия для генерации оповещений и т. д.), предназначенных для вашего объекта мониторинга.

Готовые шаблоны Zabbix предлагают разнообразные заранее определенные конфигурации мониторинга для операционных систем, приложений, баз данных, сетевых устройств, служб и многого другого.

На этом шаге вы можете:

- Просматривать шаблоны по классу (Cloud, Database, Network и т. д.) и подклассу (например, automation, backup), которые основаны на тегах шаблона.
- Искать шаблоны по ключевым словам в имени шаблона или в именах и значениях тегов шаблона.
- Фильтровать шаблоны по способу сбора данных (на основе агента или без агента; шаблоны на основе агента включают как минимум один элемент данных агент Zabbix).
- Фильтровать шаблоны по режиму агента (active или passive; активные шаблоны включают как минимум один элемент данных Zabbix agent (active), а пассивные шаблоны включают как минимум один пассивный элемент; подробности см. в разделе Пассивные и активные проверки агента).
- При использовании мастера узла сети для настройки существующего узла сети вы также можете фильтровать шаблоны, уже привязанные к узлу сети.




**Attention:**
При выборе шаблона может появиться следующее сообщение: *Некоторые шаблоны (n) несовместимы с мастером узла сети. Узнайте, как их обновить. Пользовательские шаблоны не поддерживаются.* Это означает, что некоторые готовые шаблоны пока не совместимы с мастером узла сети и требуют обновления; см. инструкции по обновлению шаблона.

Выбранный вами шаблон определит следующие шаги в мастере узла сети. Например, если вы выберете шаблон *MySQL by Zabbix agent*, который использует агент Zabbix для сбора данных, мастер проведет вас через процесс установки агента.

Создание или выбор узла сети

Шаблон должен быть связан с узлом сети — сущностью в Zabbix, которая представляет цель мониторинга. После связывания узел сети получает все сущности шаблона, такие как элементы данных (метрики, которые нужно собирать) и триггеры (условия для создания оповещений), необходимые для мониторинга.

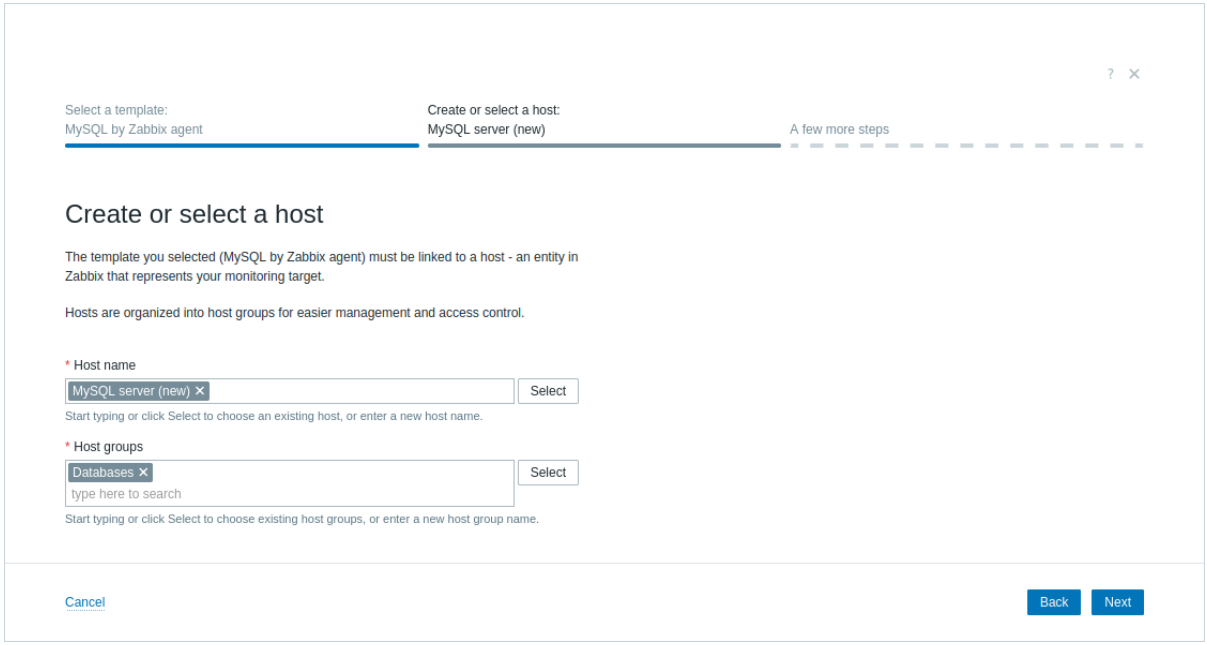
Каждый узел сети также должен принадлежать как минимум одной группе узлов сети, которая используется для организации узлов сети и назначения им прав пользователей.

На этом шаге вы можете:

- Создать новый узел сети и группу узлов сети, указав имена для обоих.
- Создать новый узел сети, указав его имя, и выбрать существующую группу узлов сети, к которой будет назначен новый узел сети.
- Выбрать существующий узел сети; вы также можете назначить этот узел сети дополнительным группам узлов сети, не удаляя его из текущих групп.

**Note:**
Выбор существующего узла сети не удаляет и не перезаписывает его текущую конфигурацию, если это явно не указано на последующих шагах. Обнаруженные узлы сети выбрать нельзя.

Например, вы можете создать новый узел сети с именем *MySQL server*, чтобы представить локально установленный сервер MySQL, который вы хотите мониторить. Ранее выбранный шаблон *MySQL by Zabbix agent* будет связан с этим узлом сети. Кроме того, вы можете выбрать существующую группу узлов сети *Databases* (или создать новую, например *MySQL servers*), чтобы объединить узел сети с другими узлами сети, связанными с базами данных, и впоследствии упростить управление правами доступа.




Установка агента Zabbix

Шаблоны на основе агента (например, *MySQL by Zabbix agent*) требуют установки Zabbix agent или agent 2 — процесса, развернутого на целевом узле сети для активного мониторинга локальных ресурсов и приложений (если это требуется шаблоном).

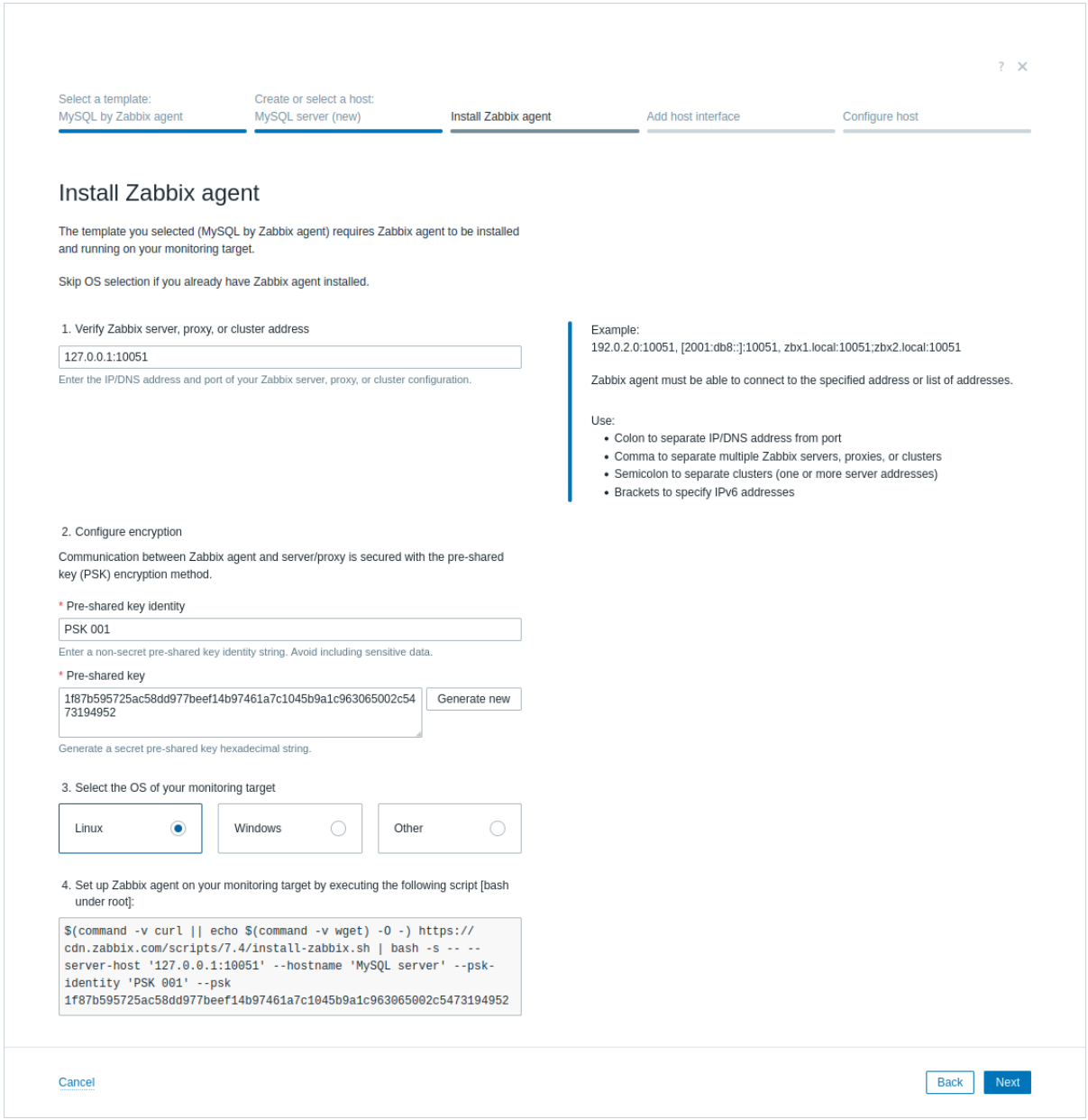
На этом шаге:

- Проверьте адрес сервера Zabbix, прокси или кластера (например, 192.0.2.1:10051). Это значение будет использоваться для настройки параметров Server и ServerActive для агента Zabbix.
- Введите новый, уникальный и не секретный идентификатор предварительно разделяемого ключа (например, *PSK 001* или *mysql-agent-psk1*) и сгенерируйте предварительно разделяемый ключ. Эти значения будут использоваться для настройки шифрования pre-shared key (PSK) для узла сети Zabbix (например, *MySQL server*) и агента Zabbix.

**Attention:**
При редактировании существующего узла сети конфигурация PSK перезапишет все существующие настройки шифрования на узле сети.

- Выберите операционную систему машины, на которой размещен ваш целевой объект мониторинга.
- Установите агент Zabbix в этой системе, запустив предоставленный скрипт на ней или следуя инструкциям по установке.

Например, вы можете задать *Pre-shared key identity* как *PSK 001* и сгенерировать новый *Pre-shared key*. Затем вы можете выбрать *Linux* в качестве операционной системы и запустить предоставленный скрипт в этой системе. После установки агента Zabbix вернитесь к мастеру узла сети.

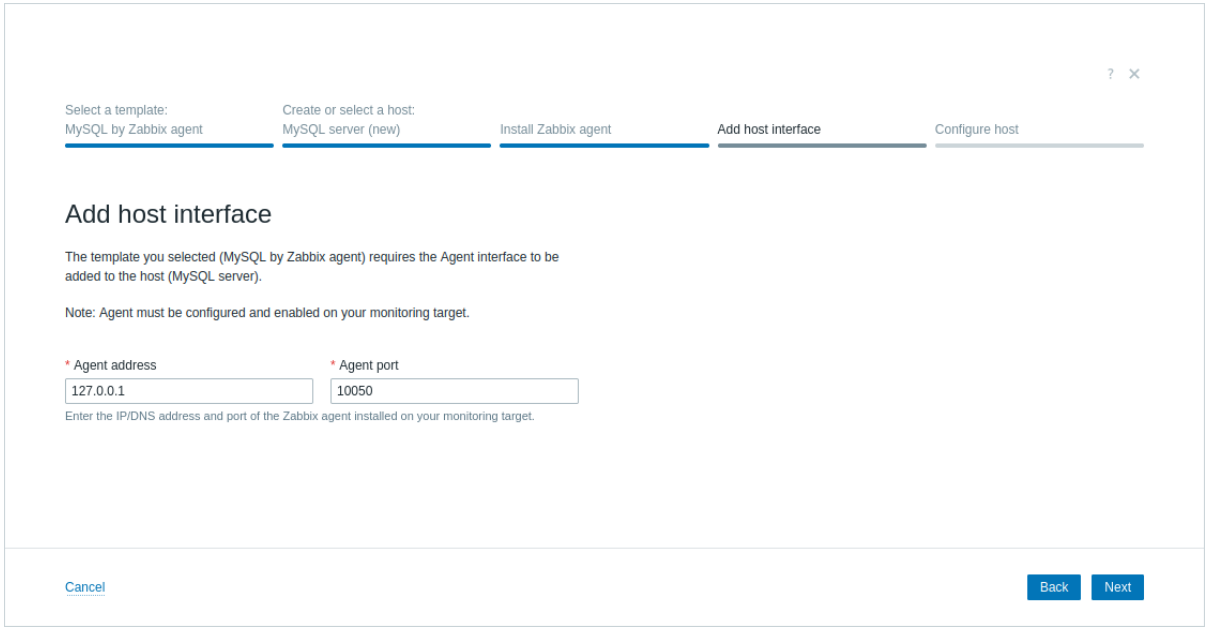



Добавление интерфейса узла сети

Интерфейс узла сети определяет, как сервер Zabbix подключается к целевому объекту мониторинга по сети. Он задает такие параметры подключения, как IP-адрес, DNS-имя, порт и тип интерфейса (Agent, SNMP, JMX или IPMI) в зависимости от метода сбора данных, требуемого выбранным шаблоном.

На этом шаге введите сведения об интерфейсе узла сети, требуемые выбранным шаблоном.

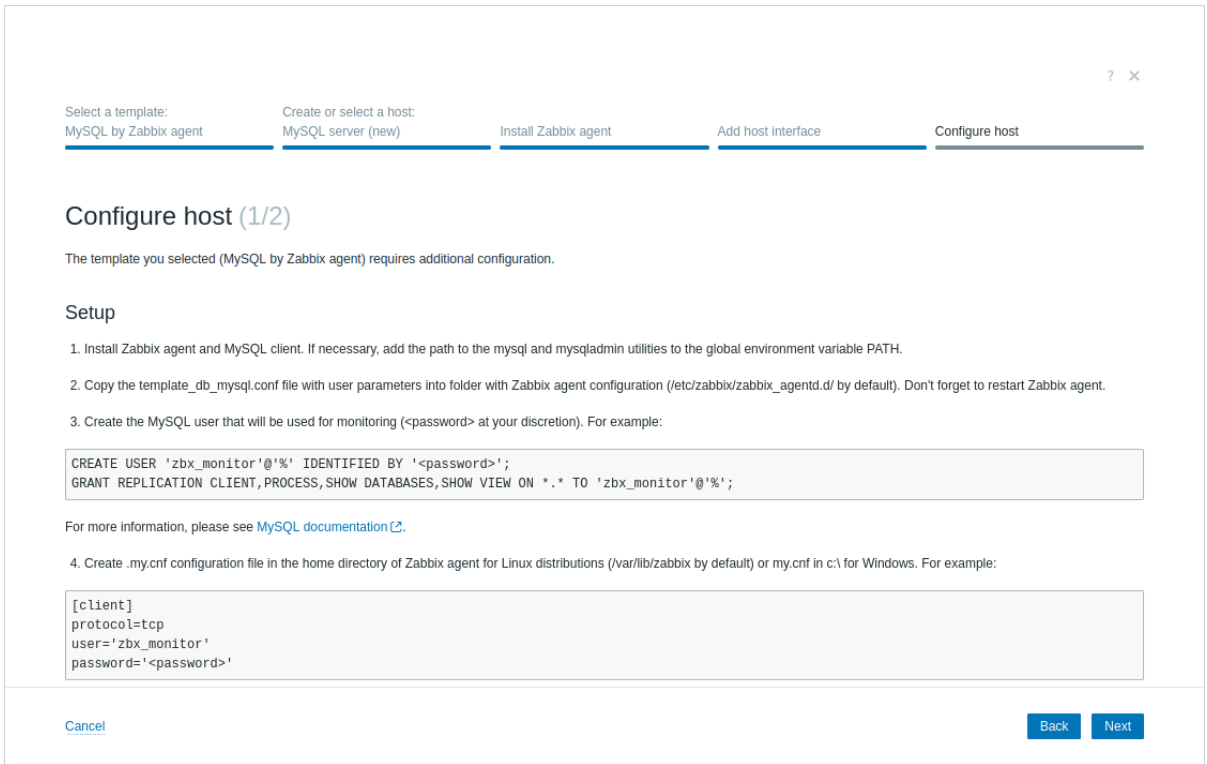
Например, вы можете использовать адрес агента по умолчанию (127.0.0.1) и порт (10050), если сервер Zabbix, агент Zabbix и сервер MySQL работают локально на одном и том же компьютере.



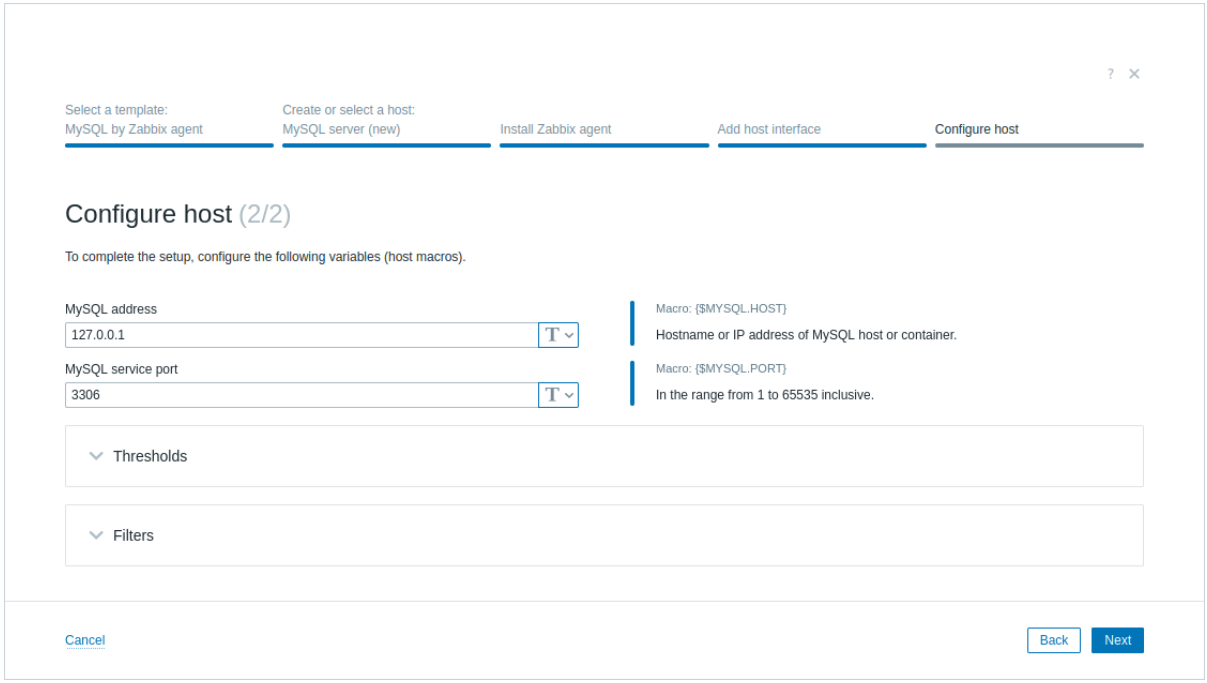

Настройка узла сети

Некоторые шаблоны требуют дополнительной настройки перед созданием узла сети. Это может включать:

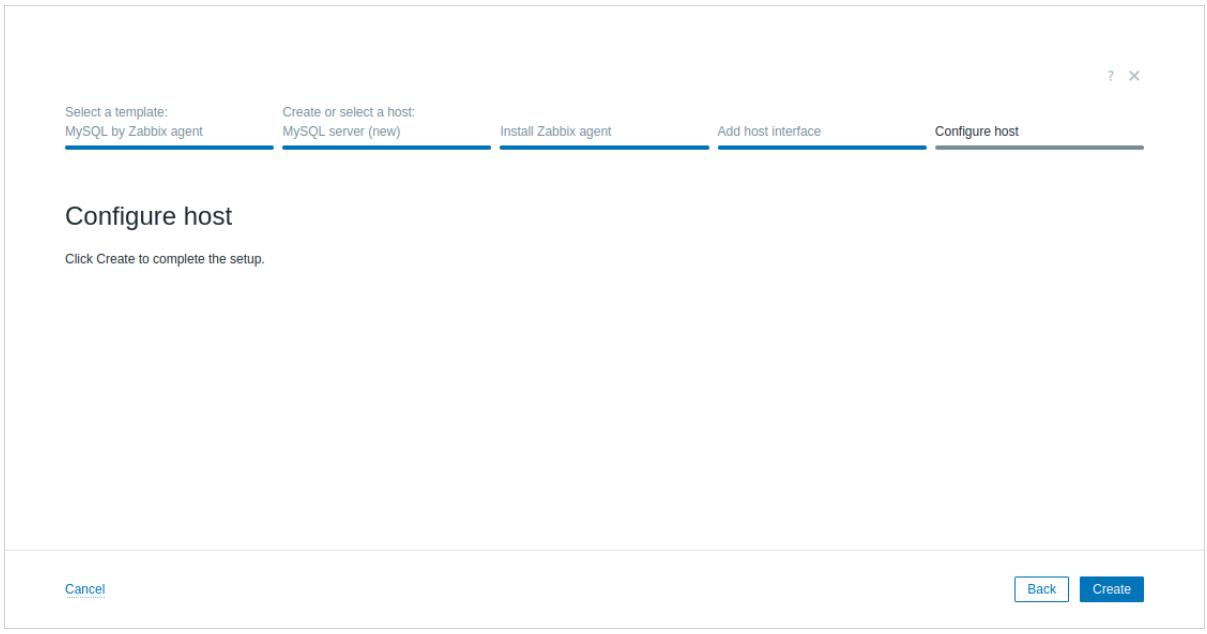
- Ручную настройку объекта мониторинга (например, включение определенных служб, создание пользователей службы или предоставление разрешений):




- Определение макросов узла сети — переменных, которые управляют поведением элемента данных, параметрами подключения, учетными данными для аутентификации и другими параметрами:



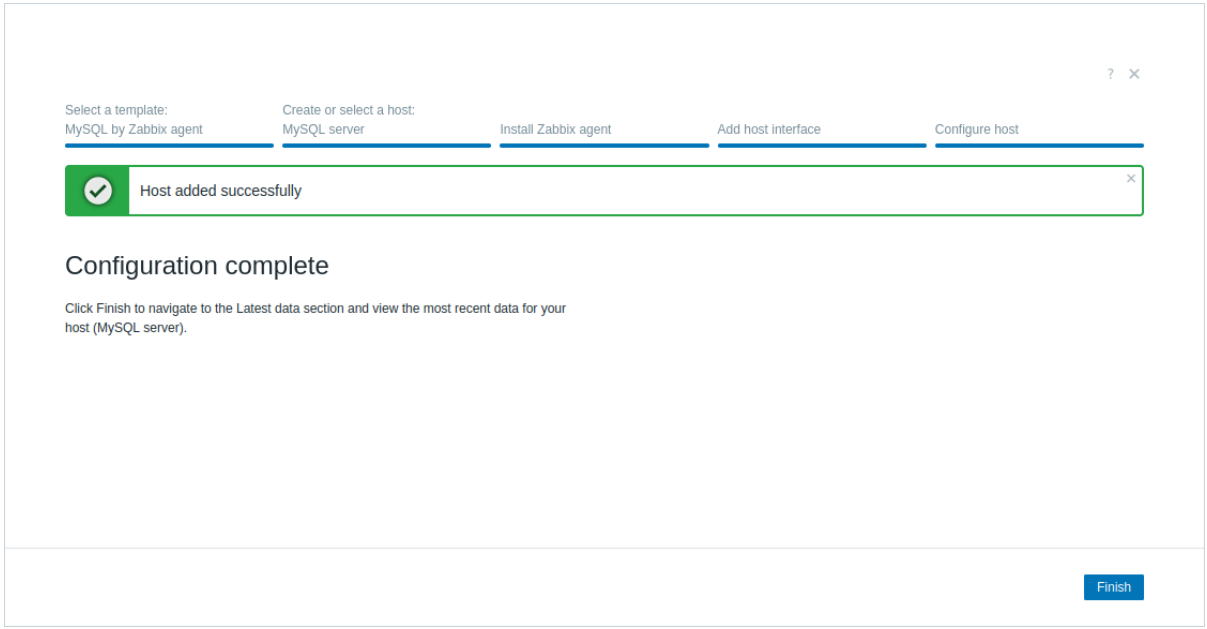

После выполнения всех необходимых шагов настройки нажмите *Create*, чтобы добавить узел сети в Zabbix (или *Update*, чтобы обновить конфигурацию существующего узла сети):




Настройка завершена

На этом этапе Zabbix уже отслеживает ваш целевой узел сети.

Для новых узлов сети нажмите *Finish*, чтобы перейти в раздел Latest data и просмотреть самые последние данные для вашего узла сети. Для существующих узлов сети нажмите *Finish*, чтобы закрыть мастер узла сети и вернуться к экрану, с которого он был открыт.




**2 Настройка узла сети**

Обзор

Чтобы настроить узел сети в веб-интерфейсе Zabbix, выполните следующие действия:

- Перейдите в: *Data collection* > *Hosts* или *Monitoring* > *Hosts*
- Нажмите *Create host* в правом верхнем углу экрана (или на имя узла сети, чтобы изменить существующий узел сети)
- Введите параметры узла сети в форме

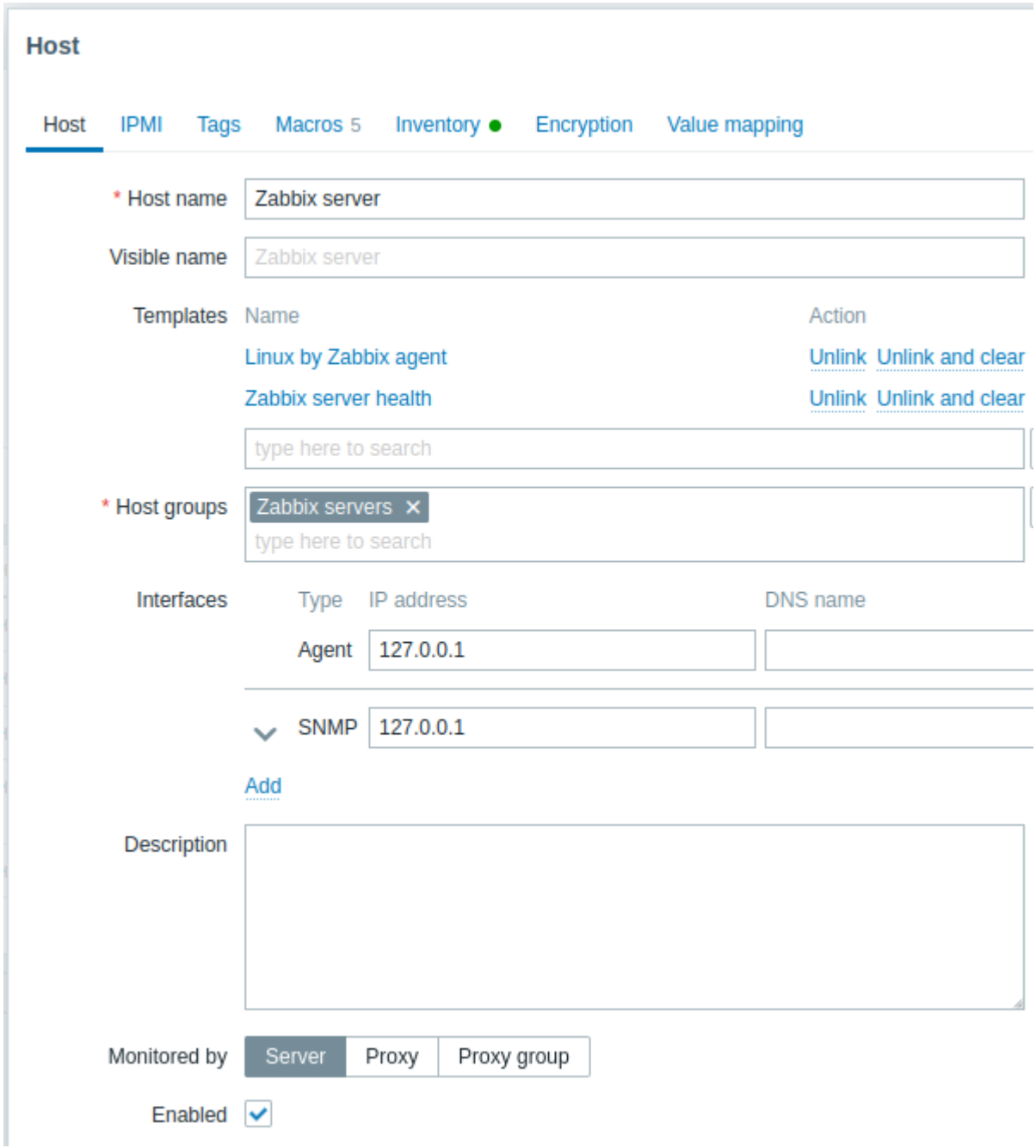
Вы также можете использовать кнопку *Clone* в форме настройки существующего узла сети, чтобы создать новый узел сети. Этот узел сети будет иметь все свойства существующего узла сети, включая связанные шаблоны, сущности (элементы данных, триггеры и т. д.) из этих шаблонов, а также сущности, напрямую привязанные к существующему узлу сети.

Обратите внимание, что при клонировании узла сети все сущности шаблона сохранятся в том виде, в котором они изначально заданы в шаблоне. Любые изменения этих сущностей, внесенные на уровне существующего узла сети (например, измененный интервал элемента данных, измененное регулярное выражение или добавленные прототипы к правилу низкоуровневого обнаружения), не будут клонированы в новый узел сети; вместо этого они будут такими же, как в шаблоне.

Кроме того, вы можете использовать Host Wizard для настройки узла сети через пошаговый интерфейс с подсказками.

Конфигурация

Вкладка **Узел сети** содержит общие атрибуты узла сети:




Все обязательные поля ввода отмечены красной звездочкой.

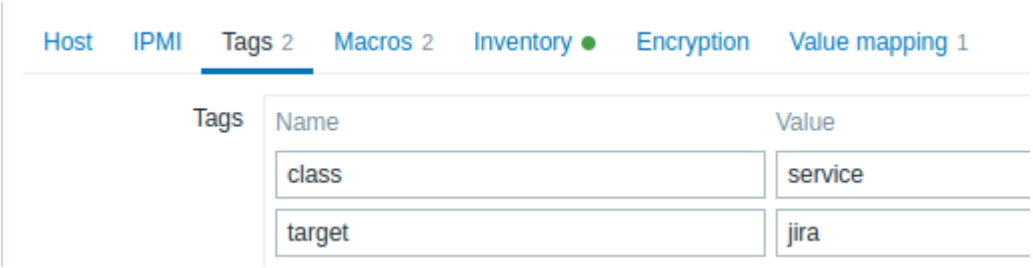
| Параметр | Описание |
|---|---|
| *Имя узла сети* | Введите уникальное имя узла сети. Допускаются буквенно-цифровые символы, пробелы, точки, дефисы и символы подчеркивания. Однако начальные и конечные пробелы не допускаются.<br>*Примечание:* Если на настраиваемом узле сети запущен агент Zabbix, параметр *Hostname* в файле конфигурации агента должен иметь то же значение, что и имя узла сети, введенное здесь. Имя в этом параметре необходимо для обработки активных проверок. |
| *Отображаемое имя* | Введите уникальное отображаемое имя узла сети. Если вы зададите это имя, именно оно будет отображаться в списках, картах и т. д. вместо технического имени узла сети. Этот атрибут поддерживает UTF-8. |

| Параметр | Описание |
|---|---|
| *Шаблоны* | Свяжите с узлом сети шаблоны. Все сущности (элементы данных, триггеры, и т. д.) будут унаследованы от шаблона.<br>Чтобы связать новый шаблон, начните вводить имя шаблона в текстовом поле. Появится список подходящих шаблонов; прокрутите вниз, чтобы выбрать нужный.<br>Также можно нажать *Выбрать* рядом с полем и выбрать шаблоны из списка во всплывающем окне. Все выбранные шаблоны будут связаны с узлом сети при сохранении или обновлении формы настройки узла сети.<br>Чтобы отвязать шаблон, используйте один из двух вариантов в блоке *Связанные шаблоны*:<br>*Отвязать* - отвязать шаблон, сохранив его сущности (элементы данных, триггеры, и т. д.);<br>*Отвязать и очистить* - отвязать шаблон и удалить все его сущности (элементы данных, триггеры, и т. д.).<br>Названия перечисленных шаблонов являются кликабельными ссылками, ведущими к форме настройки шаблона. |
| *Группы узлов сети* | Выберите группы узлов сети, к которым принадлежит узел сети. Узел сети должен принадлежать как минимум к одной группе узлов сети. Новую группу можно создать и связать с узлом сети, добавив несуществующее имя группы. |
| *Интерфейсы* | Для узла сети поддерживается несколько типов интерфейсов: *Agent*, *SNMP*, *JMX* и *IPMI*.<br>По умолчанию интерфейсы не заданы. Чтобы добавить новый интерфейс, нажмите *Добавить* в блоке *Интерфейсы*, выберите тип интерфейса и укажите сведения *IP/DNS*, *Подключаться к* и *Порт*.<br>*Примечание:* Интерфейсы, которые используются в каких-либо элементах данных, удалить нельзя, и ссылка *Удалить* для них неактивна.<br>Значение ”IP” или ”DNS” из интерфейса SNMP также используется для SNMP traps. При сопоставлении используется только выбранное значение ”IP” или ”DNS” в интерфейсе узла сети.<br>Дополнительные сведения о настройке интерфейса SNMP (v1, v2 и v3) см. в разделе Настройка мониторинга SNMP. |
| *IP-адрес* | IP-адрес узла сети (необязательно). |
| *Имя DNS* | Имя DNS узла сети (необязательно). |
| *Подключаться к* | При нажатии соответствующей кнопки Zabbix server будет указано, что использовать для получения данных от агентов:<br>**IP** - подключаться к IP-адресу узла сети (рекомендуется)<br>**DNS** - подключаться к имени DNS узла сети |
| *Порт* | Номер TCP/UDP-порта. Значения по умолчанию: 10050 для агента Zabbix, 161 для агента SNMP, 12345 для JMX и 623 для IPMI. |
| *По умолчанию* | Установите переключатель, чтобы задать интерфейс по умолчанию. |
| *Описание* | Введите описание узла сети. |
| *Мониторинг выполняется через* | Выберите, кем выполняется мониторинг узла сети:<br>**Server** - узел сети мониторится Zabbix server;<br>**Proxy** - узел сети мониторится автономным прокси;<br>**Proxy group** - узел сети мониторится группой прокси. |
| *Proxy* | Отображается назначенное имя прокси (только если Zabbix server назначил его из выбранной группы прокси).<br>Это поле отображается только если узел сети мониторится группой прокси. |
| *Включено* | Если флажок установлен, узел сети включен - готов к мониторингу.<br><br>Если флажок снят, узел сети отключен - не мониторится:<br>Для пассивных запросов данных, инициируемых Zabbix server/proxy (например, агент Zabbix, агент SNMP, простые проверки), мониторинг отключается после синхронизации конфигурации. Триггеры и действия, связанные с узлом сети, также отключаются только после перезагрузки конфигурации.<br>Для активных проверок агента Zabbix мониторинг прекращается в течение времени (примерно 5 секунд), за которое агент Zabbix получает информацию о том, что узел сети был отключен. В этот короткий промежуток узел сети продолжит локально собирать данные для активных проверок и пытаться отправить их на сервер/proxy; однако, поскольку узел сети помечен как *Disabled*, сервер/proxy отклонит данные.<br><br>При отключении узла сети его элементы данных немедленно удаляются из кэша истории (за исключением последних значений, которые сохраняются для журналов). |

Вкладка **IPMI** содержит атрибуты управления IPMI.

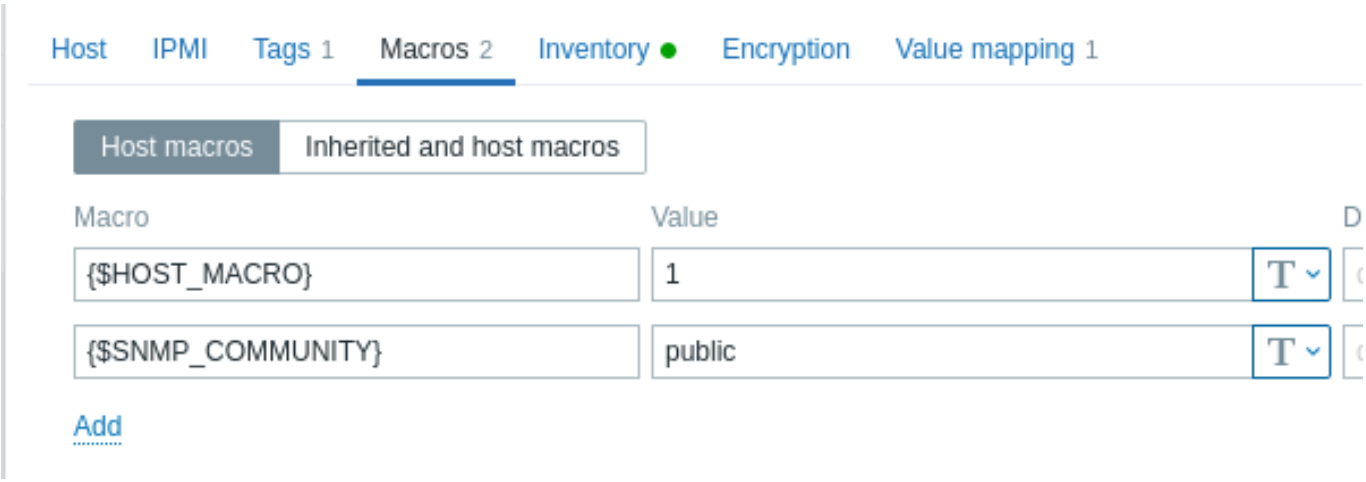
| Параметр | Описание |
|---|---|
| *Алгоритм аутентификации* | Выберите алгоритм аутентификации. |
| *Уровень привилегий* | Выберите уровень привилегий. |
| *Имя пользователя* | Имя пользователя для аутентификации. Можно использовать пользовательские макросы. |
| *Пароль* | Пароль для аутентификации. Можно использовать пользовательские макросы. |

Вкладка **Теги** позволяет определить теги на уровне узла сети. Все проблемы этого узла сети будут помечены значениями, введенными здесь.

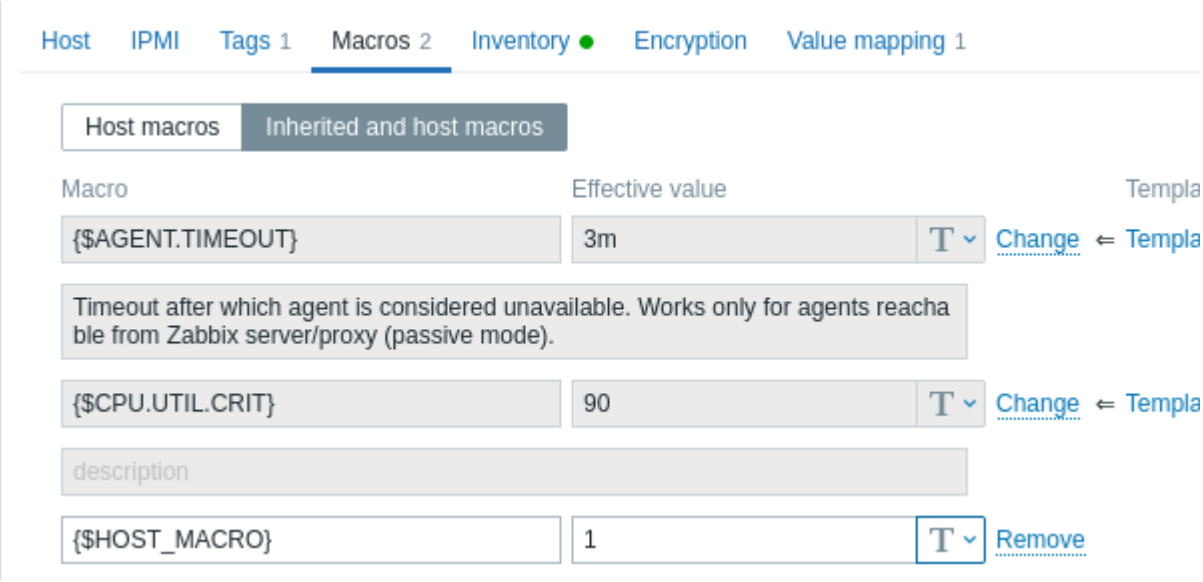



В тегах поддерживаются пользовательские макросы, макросы {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} и {HOST.ID}.

Вкладка **Макросы** позволяет определить пользовательские макросы на уровне узла сети в виде пар имя-значение. Обратите внимание, что значения макросов могут храниться как обычный текст, секретный текст или секрет Vault. Также поддерживается добавление описания.

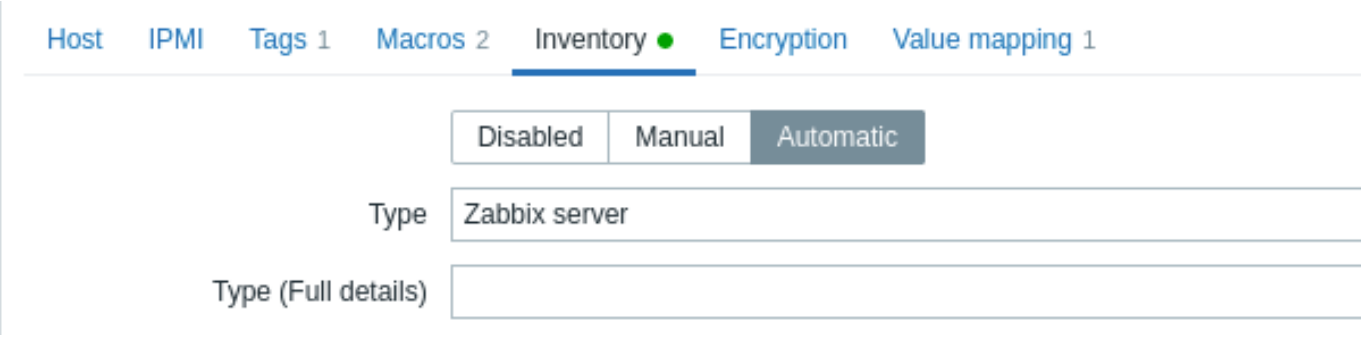



Здесь также можно просмотреть пользовательские макросы уровня шаблона и глобальные пользовательские макросы, если выбрать опцию *Унаследованные и макросы узла сети*. Здесь отображаются все определенные пользовательские макросы для узла сети со значением, к которому они приводятся, а также их источник.




Для удобства предоставлены ссылки на соответствующие шаблоны и настройки глобальных макросов. Также можно редактировать макрос шаблона/глобальный макрос на уровне узла сети, фактически создавая копию макроса на узле сети.

Вкладка **Инвентаризация** позволяет вручную ввести сведения инвентаризации для узла сети. Также можно включить *Автоматическое* заполнение инвентаризации или отключить заполнение инвентаризации для этого узла сети.




Если инвентаризация включена (вручную или автоматически), рядом с названием вкладки отображается зеленая точка.

Шифрование

Вкладка **Шифрование** позволяет вам требовать шифрование соединений с узлом сети.

| Параметр | Описание |
|---|---|
| *Подключения к узлу сети (Connections to host)* | Каким образом Zabbix сервер или прокси подключаются к Zabbix агенту на хосте: без шифрования (по умолчанию), используя общий ключ (PSK, pre-shared key) или сертификат. |
| *Соединения с узла сети (Connections from host)* | Выберите, какой тип подключений разрешён с узла сети (то есть, с Zabbix агента и Zabbix sender). Можно выбрать несколько типов соединений одновременно (полезно при тестировании и переключении на другой тип соединения). По умолчанию — «Без шифрования». |
| *Издатель (Issuer)* | Разрешённый эмитент сертификата. Сертификат сначала подтверждается CA (certificate authority, центром сертификации). Если он действительный, подписан с помощью CA, тогда можно использовать поле *Издатель* для более строгого ограничения разрешённых CA. Это поле предназначено для использования, если ваша Zabbix инсталляция использует сертификаты от нескольких CA. Если поле не заполнено, то принимается любой CA. |
| *Тема (Subject)* | Разрешённая тема сертификата. Сертификат сначала подтверждается CA. Если он действительный, подписан с помощью CA, тогда можно использовать поле *Тема*, чтобы разрешить только одно значение строки *Тема*. Если поле пустое, тогда принимается любой сертификат, подписанный настроенным CA. |
| *Идентификатор PSK (PSK identity)* | Строка идентификации общего ключа (pre-shared key).<br>Не указывайте конфиденциальную информацию в идентификаторе PSK, так как он передаётся по сети в незашифрованном виде для информирования получателя, какой PSK использовать. |

| Параметр | Описание |
|---|---|
| *PSK* | Pre-shared key (строка в шестнадцатеричном формате). Максимальная длина: 512 шестнадцатеричных цифр (256-байт PSK), если Zabbix использует библиотеки GnuTLS или OpenSSL, 64 шестнадцатеричных цифры (32-байт PSK), если Zabbix использует библиотеку mbed TLS (PolarSSL). Пример: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952 |

Преобразования значений

Вкладка **Преобразования значений (Value mapping)** позволяет настраивать в преобразованиях значений удобные для человеческого восприятия представления значений элементов данных.

### 3 Настройка группы узлов сети

Обзор

Группы узлов сети используются для логического объединения узлов сети и назначения им прав доступа пользователей.

Каждый узел сети должен быть назначен как минимум в одну группу узлов сети. Узел сети может принадлежать нескольким группам узлов сети, и каждая группа узлов сети может содержать несколько узлов сети.

Обратите внимание, что в Zabbix все права доступа основаны на группах: группах пользователей, группах узлов сети и группах шаблонов. Поэтому даже если одному пользователю нужен доступ к одному узлу сети, он предоставляется путем добавления пользователя в группу пользователей, у которой есть право доступа к группе узлов сети, содержащей этот узел сети.

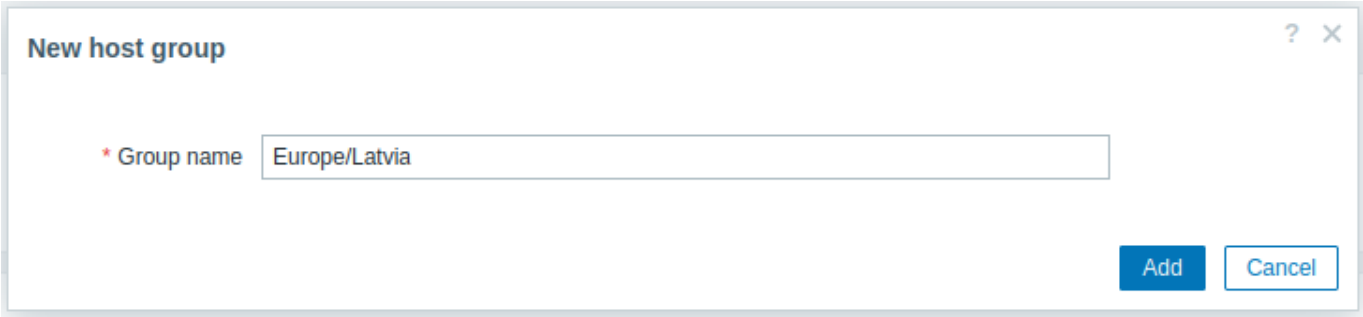
Конфигурация

**Attention:**
Только пользователи с ролью Super admin могут создавать группы узлов сети.

В веб-интерфейсе Zabbix есть два способа создания группы узлов сети.

**Вариант 1**:

- Перейдите в: *Сбор данных → Группы узлов сети*
- Нажмите *Создать группу узлов сети* в правом верхнем углу экрана
- Введите имя группы в форме




**Вариант 2**: при настройке узла сети введите несуществующее имя группы в поле *Группы узлов сети*.

После создания группы узлов сети вы можете нажать на имя группы в списке в разделе *Сбор данных → Группы узлов сети*, чтобы изменить имя группы, клонировать группу или удалить ее.

При удалении группы узлов сети удаляется только логическая группа, а не узлы сети в этой группе. Невозможно удалить группу узлов сети, если она является единственной группой для какого-либо существующего узла сети.

Создание подгрупп узлов сети

Подгруппа узлов сети (или вложенная группа узлов сети) является дочерней по отношению к родительской группе узлов сети, в которую она входит.

Подгруппа создается с помощью символа косой черты '/' в поле ввода имени группы, чтобы обозначить ее связь с родительской группой(ами). Например:

- ввод `Europe/Latvia` создает подгруппу `Europe/Latvia` в родительской группе `Europe`.

- ввод `Europe/Latvia/Riga/Zabbix servers` создает соответствующую подгруппу во вложенных родительских группах `Europe`, `Europe/Latvia`, `Europe/Latvia/Riga`.

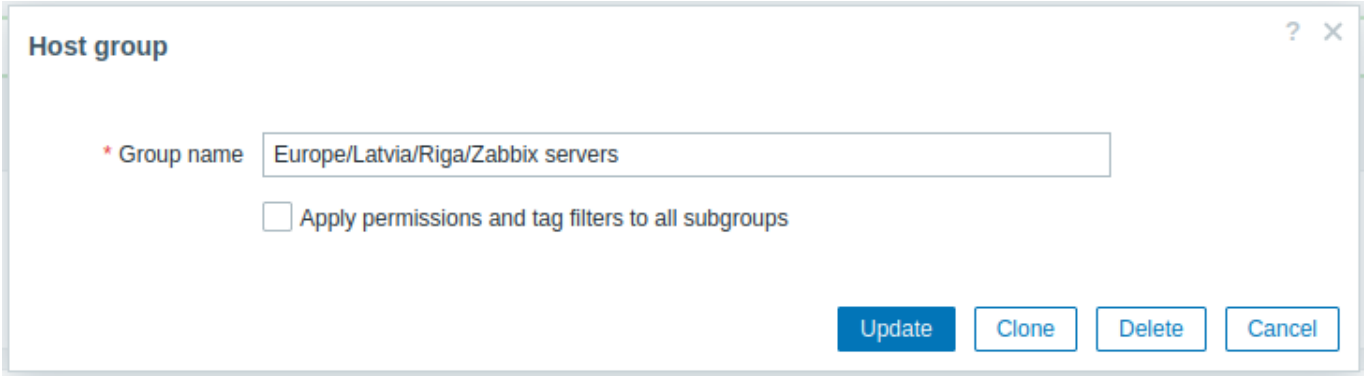
При создании подгруппы не допускается использование начальной или конечной косой черты, а также нескольких косых черт подряд. Экранирование символа '/' не поддерживается.

Перед созданием подгруппы не требуется предварительно создавать какие-либо родительские группы узлов сети. Вы можете начать либо с создания подгруппы (например, `Europe/Latvia`), либо с создания любой родительской группы узлов сети (в нашем примере, `Europe`). Если вы начнете с создания подгруппы, родительская группа(ы) узлов сети **не** будут созданы автоматически.

Права доступа к группам узлов сети

- При создании подгруппы в существующей родительской группе узлов сети (например, при создании `Europe/Latvia`, когда `Europe` уже существует), группа пользователей наследует права доступа к подгруппе от родительской группы.
- При создании родительской группы узлов сети для уже существующей подгруппы (например, при создании `Europe`, когда `Europe/Latvia` уже существует), права доступа к родительской группе не задаются.

При редактировании любой группы узлов сети можно также задать дополнительную опцию *Применить права доступа и фильтры тегов ко всем подгруппам*.




Если установить этот флажок и нажать *Обновить*, будут применены те же уровни прав доступа и фильтры тегов ко всем текущим и будущим подгруппам редактируемой группы узлов сети.

Таким образом, если каким-либо группам пользователей были назначены разные права доступа к подгруппам редактируемой группы узлов сети, установка флажка предоставит всем текущим и будущим подгруппам те же права доступа пользователей и права доступа на основе тегов, что и у редактируемой группы.

Обратите внимание, что эта опция не сохраняется в базе данных и переопределяет существующие права доступа. Любые изменения, внесенные с помощью этой опции, можно отменить только вручную.

## 4 Инвентаризация

Обзор

Вы можете хранить инвентарные данные сетевых устройств в Zabbix.

В веб-интерфейсе Zabbix существует специальное меню *Инвентаризация (Inventory)*. Однако, поначалу вы не увидите там никаких данных, и это не то место, где вы вводите данные. Построение инвентарных данных выполняется вручную при настройке узла сети или автоматически с использованием каких-либо опций автоматического заполнения.

Построение инвентарных данных

Ручной режим

При настройке узла сети на вкладке *Инвентаризация* вы можете ввести такие подробности, как тип устройства, серийный номер, местоположение, ответственное лицо, ссылки и прочее — они пополнят собой инвентарные данные.

Если в инвентарные данные включён URL и он начинается с «http» или «https», в разделе *Инвентаризация* это отобразится в виде ссылки.

Автоматический режим

Также инвентарные данные узлов сети могут заполняться автоматически. Чтобы эта функция заработала, при настройке узла сети на вкладке *Инвентаризация* для режима инвентарных данных должно быть выставлено *Автоматически (Automatic)*.

Затем вы можете настроить элементы данных узла сети таким образом, чтобы они заполняли любые поля инвентарных данных своими значениями. Для этого при настройке элемента данных в соответствующем атрибуте (называемом *Заполнение поля инвентаря узла сети (Item will populate host inventory field)*) нужно указать поле назначения.

Элементы данных, которые особенно полезны для автоматического сбора инвентарных данных:

- system.hw.chassis[full|type|vendor|model|serial] — по умолчанию [full], требуются права root
- system.hw.cpu[all|cpunum,full|maxfreq|vendor|model|curfreq] — по умолчанию [all,full]
- system.hw.devices[pci|usb] — по умолчанию [pci]
- system.hw.macaddr[интерфейс,short|full] — по умолчанию [all,full], интерфейс является регулярным выражением
- system.sw.arch
- system.sw.os[name|short|full] — по умолчанию [name]
- system.sw.packages[пакет,менеджер,short|full] — по умолчанию [all,all,full], пакет является регулярным выражением

Выбор режима инвентаризации

Режим инвентаризации можно выбрать в диалоге настройки узла сети.

Режим инвентаризации по умолчанию для новых узлов сети выбирается на основе настройки *Режим инвентаризации узлов сети по умолчанию* в *Администрирование* → *Общие* → *Другое* (*Administration* → *General* →*Other*).

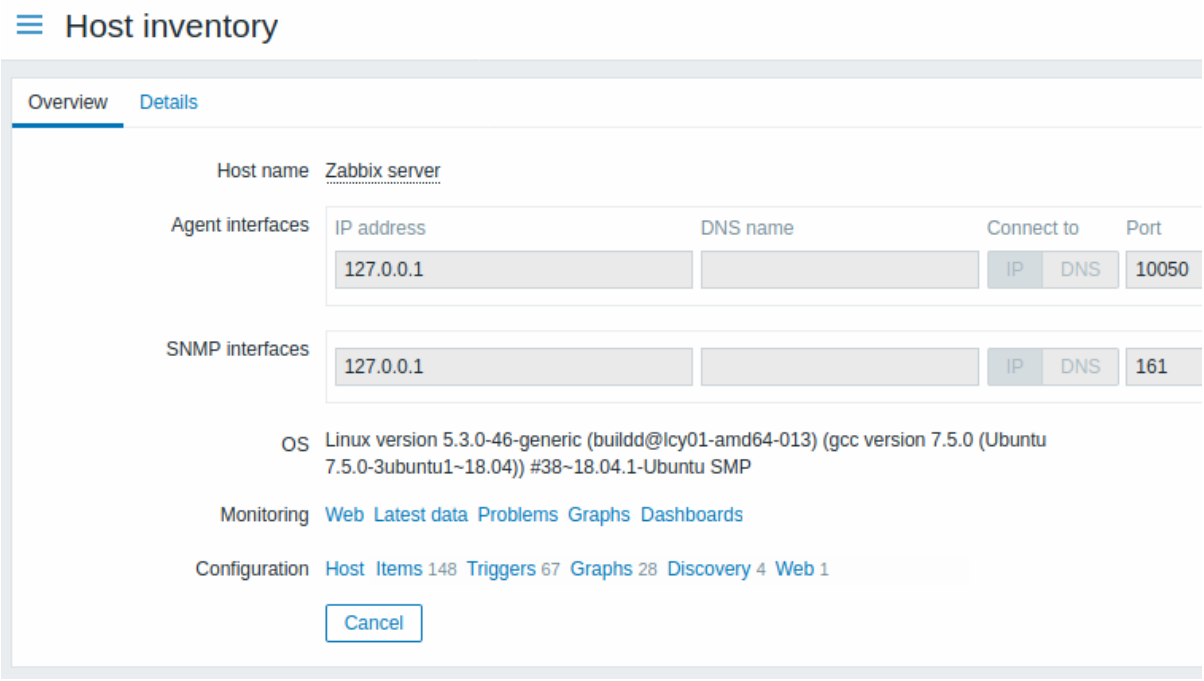
Для узлов сети, добавляемых с помощью действий по сетевым обнаружениям или авторегистрации, имеется возможность указать операцию *Задать режим инвентарных данных узлов сети (Set host inventory mode)*, выбрав ручной или автоматический режим. Эта операция переопределит настройку *Режим инвентаризации узлов сети по умолчанию*.

Обзор инвентаризации

Сведения обо всех существующих данных инвентаризации доступны в меню *Инвентаризация*.

В *Инвентаризация* → *Обзор* можно получить количество узлов сети по различным полям инвентаризации.

В *Инвентаризация* → *Узлы сети* можно увидеть все узлы сети, для которых есть информация об инвентаризации. Щелчок по имени узла сети откроет сведения об инвентаризации в форме.




Вкладка **Обзор** показывает:

| Parameter | Description |
|---|---|
| *Host name* | Имя узла сети.<br>Щелчок по имени открывает меню со скриптами, определенными для узла сети.<br>Имя узла сети отображается с оранжевым значком, если узел сети находится на обслуживании. |
| *Visible name* | Отображаемое имя узла сети (если задано). |
| *Host (Agent, SNMP, JMX, IPMI)<br>interfaces* | Этот блок содержит сведения об интерфейсах, настроенных для узла сети. |
| *OS* | Поле инвентаризации операционной системы узла сети (если задано). |
| *Hardware* | Поле инвентаризации оборудования узла сети (если задано). |

| Parameter | Description |
|---|---|
| *Software* | Поле инвентаризации программного обеспечения узла сети (если задано). |
| *Description* | Описание узла сети. |
| *Monitoring* | Ссылки на разделы мониторинга с данными для этого узла сети: *Web*, *Latest data*, *Problems*, *Graphs*, *Dashboards*. |
| *Configuration* | Ссылки на разделы конфигурации для этого узла сети: *Host*, *Items*, *Triggers*, *Graphs*, *Discovery*, *Web*.<br>Количество настроенных сущностей указано после каждой ссылки. |

Вкладка **Детали** показывает все заполненные поля инвентаризации (не пустые).

Макросы инвентаризации

Для использования в уведомлениях доступны макросы инвентаризации узла сети {INVENTORY.*}, например:

”На сервере в {INVENTORY.LOCATION1} возникла проблема, ответственное лицо — {INVENTORY.CONTACT1}, номер телефона {INVENTORY.POC.PRIMARY.PHONE.A1}.”

Подробнее см. страницу Поддерживаемые макросы.

**5 Массовое обновление**

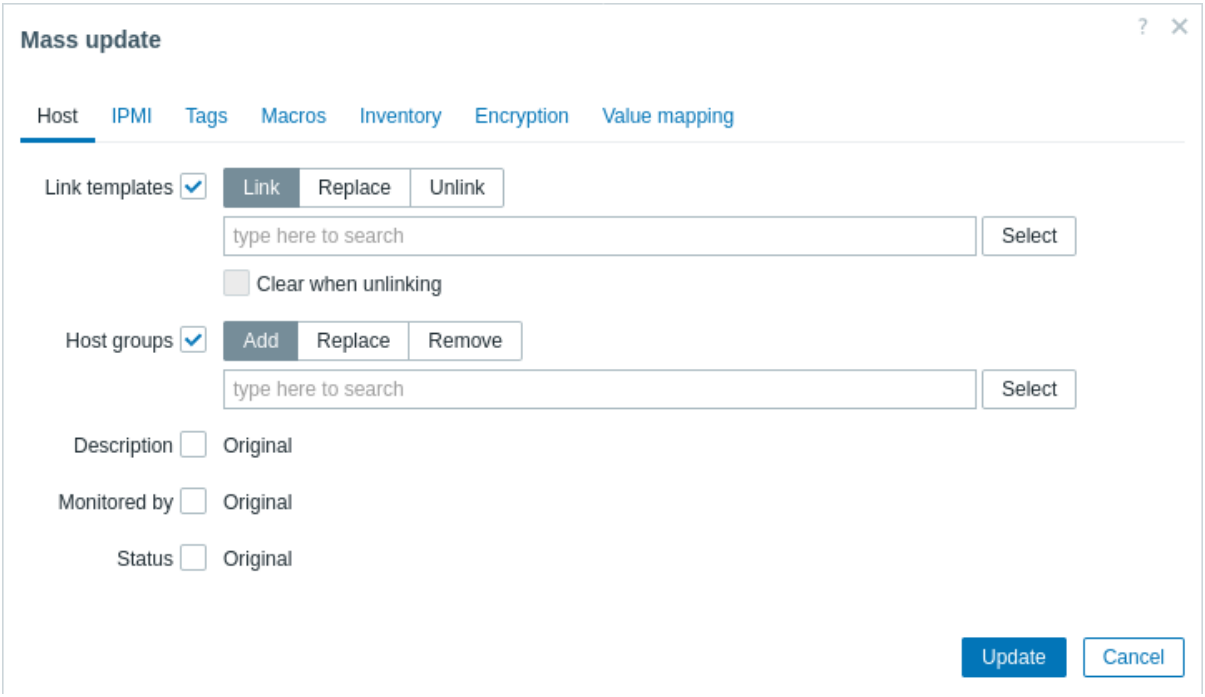
Обзор

Порой вы можете захотеть изменить какие-нибудь атрибуты для нескольких узлов сети одновременно. Вместо того, чтобы открывать каждый узел сети для редактирования, вы можете воспользоваться функцией массового обновления.

Использование массового обновления

Чтобы массово обновить некоторые узлы сети, выполните следующие действия:

- Отметьте флажки перед узлами сети, которые нужно обновить, в списке узлов сети
- Нажмите *Массовое обновление* под списком
- Перейдите на вкладку с нужными атрибутами (*Узел сети*, *IPMI*, *Теги*, *Макросы*, *Инвентаризация*, *Шифрование* или *Соответствие значений*)
- Отметьте флажки у любых атрибутов, которые нужно обновить, и введите для них новое значение




При выборе соответствующей кнопки для обновления привязки **шаблонов** доступны следующие параметры:

- *Link* - укажите, какие дополнительные шаблоны нужно привязать
- *Replace* - укажите, какие шаблоны нужно привязать, при этом отвязав все шаблоны, которые были привязаны к узлам сети ранее
- *Unlink* - укажите, какие шаблоны нужно отвязать

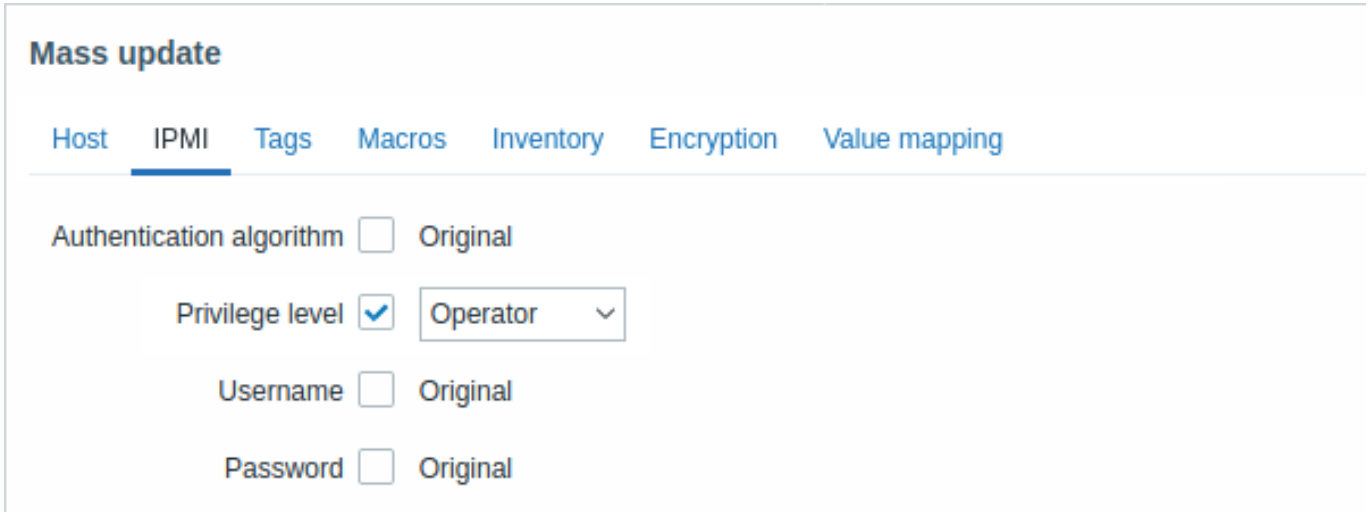
Чтобы указать шаблоны для привязки/отвязки, начните вводить имя шаблона в поле автодополнения, пока не появится выпадающий список с подходящими шаблонами. Просто прокрутите список вниз и выберите нужный шаблон.

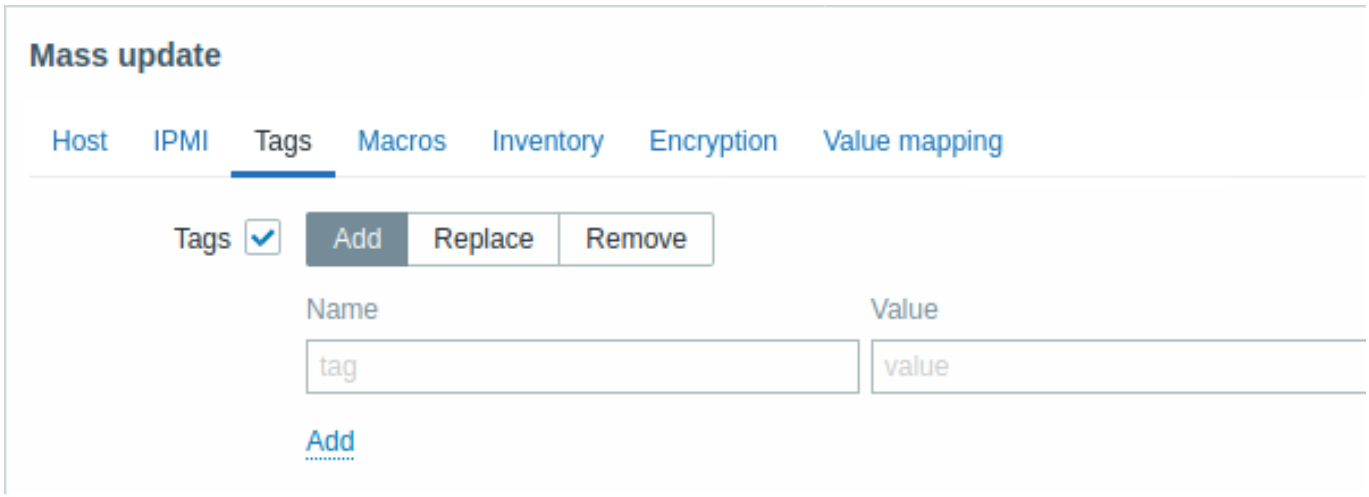
Параметр *Clear when unlinking* позволит не только отвязать ранее привязанные шаблоны, но и удалить все унаследованные от них элементы (элементы данных, триггеры и т. д.).

При выборе соответствующей кнопки для обновления **группы узлов сети** доступны следующие параметры:

- *Add* - позволяет указать дополнительные группы узлов сети из уже существующих или ввести совершенно новые группы узлов сети для узлов сети
- *Replace* - удалит узел сети из всех существующих групп узлов сети и заменит их указанной в этом поле группой или группами (существующими или новыми)
- *Remove* - удалит указанные группы узлов сети из узлов сети

Эти поля поддерживают автодополнение: при вводе в них появляется выпадающий список подходящих групп узлов сети. Если группа узлов сети новая, она также отображается в списке и помечается *(new)* после строки. Просто прокрутите вниз и выберите ее.

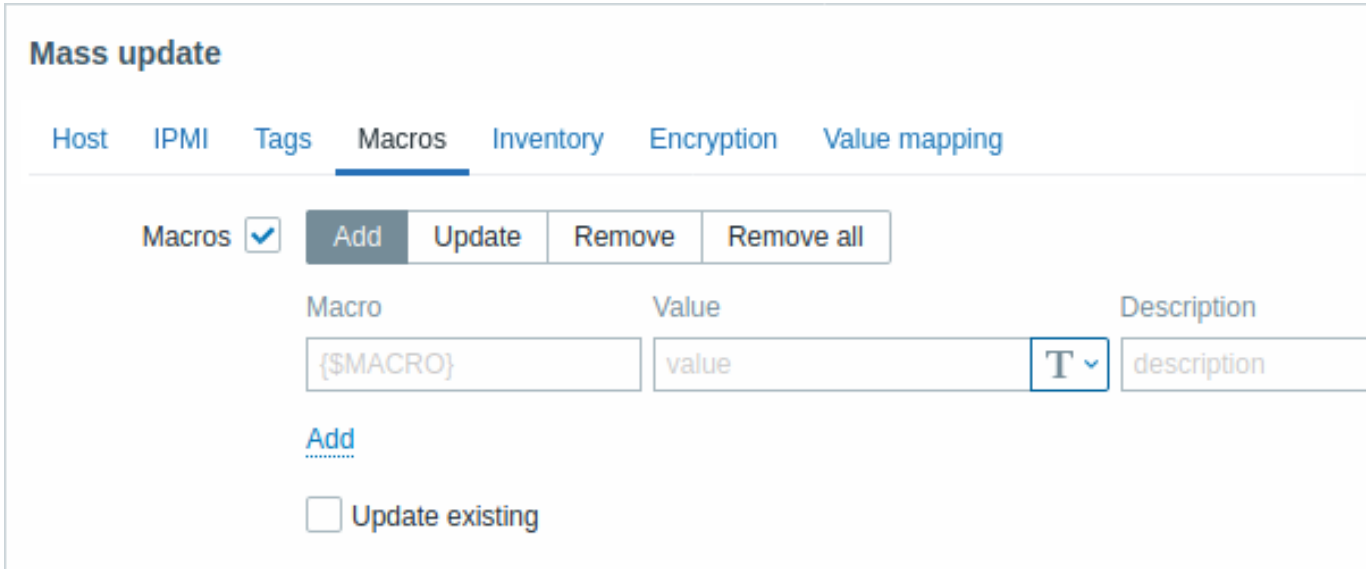



При выборе соответствующей кнопки для обновления тегов доступны следующие параметры:

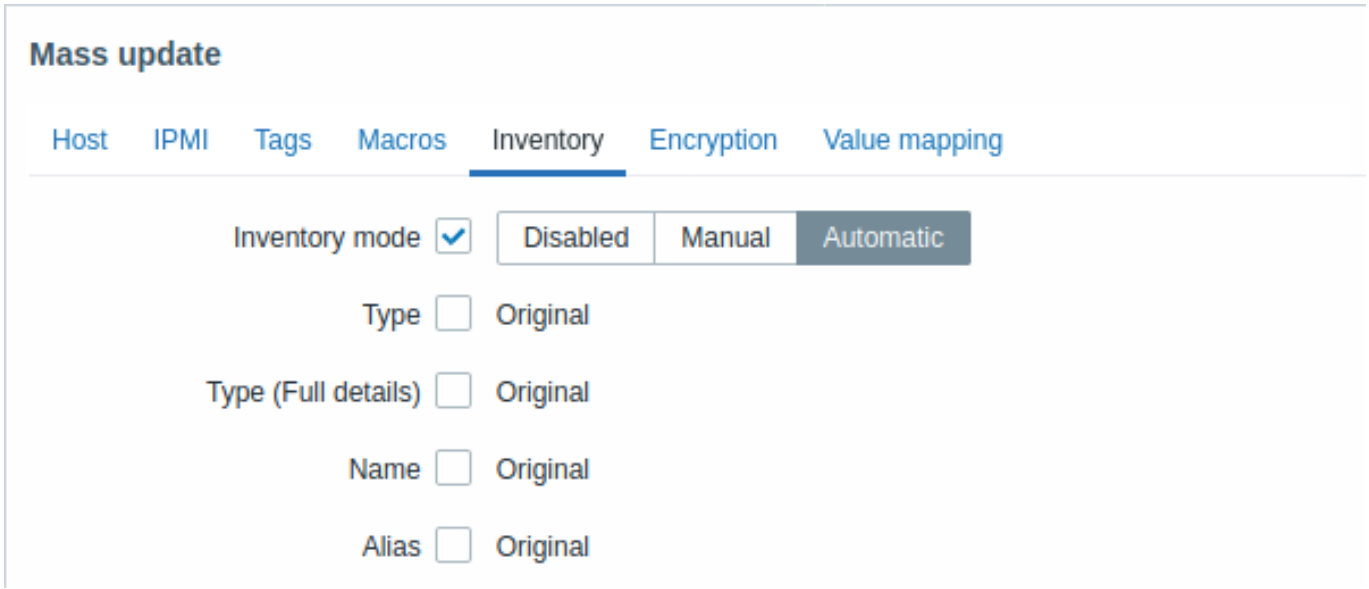
- *Add* - позволяет добавить новые теги для узлов сети;
- *Replace* - удалит все существующие теги с узла сети и заменит их указанными ниже;
- *Remove* - удалит указанные теги из узлов сети.

В тегах поддерживаются пользовательские макросы, макросы {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} и {HOST.ID} теги. Обратите внимание, что теги с одинаковым именем, но разными значениями не считаются ”дубликатами” и могут быть добавлены к одному и тому же узлу сети.

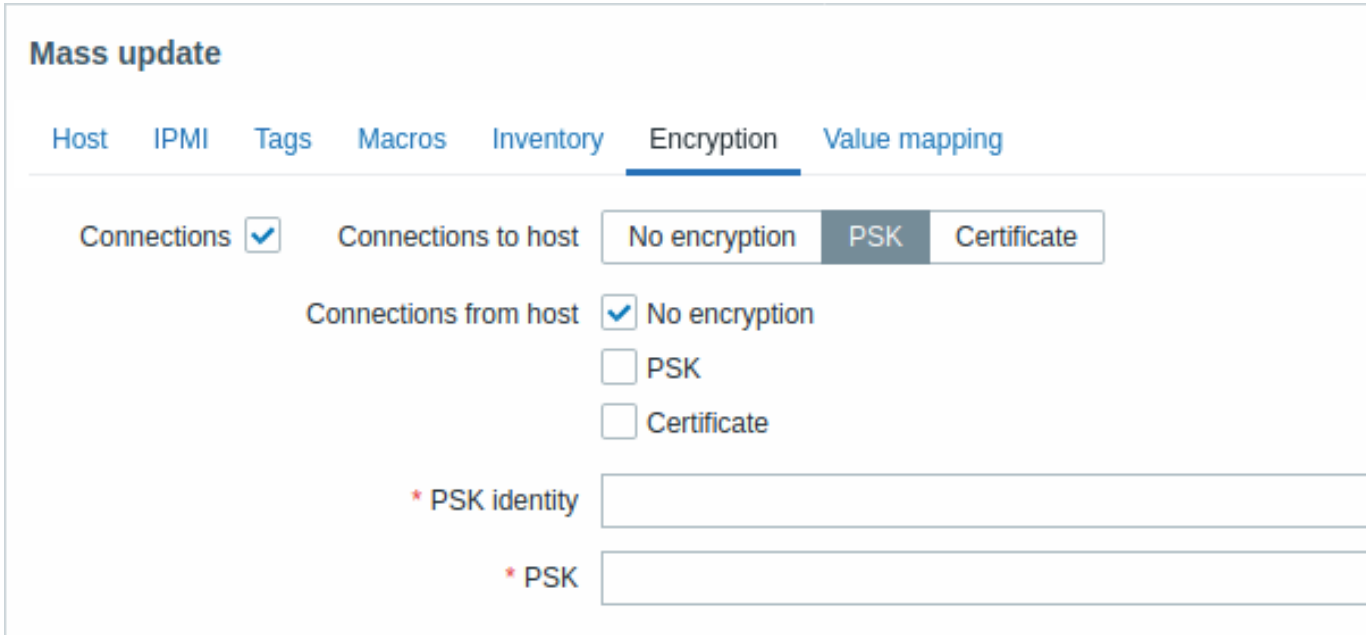


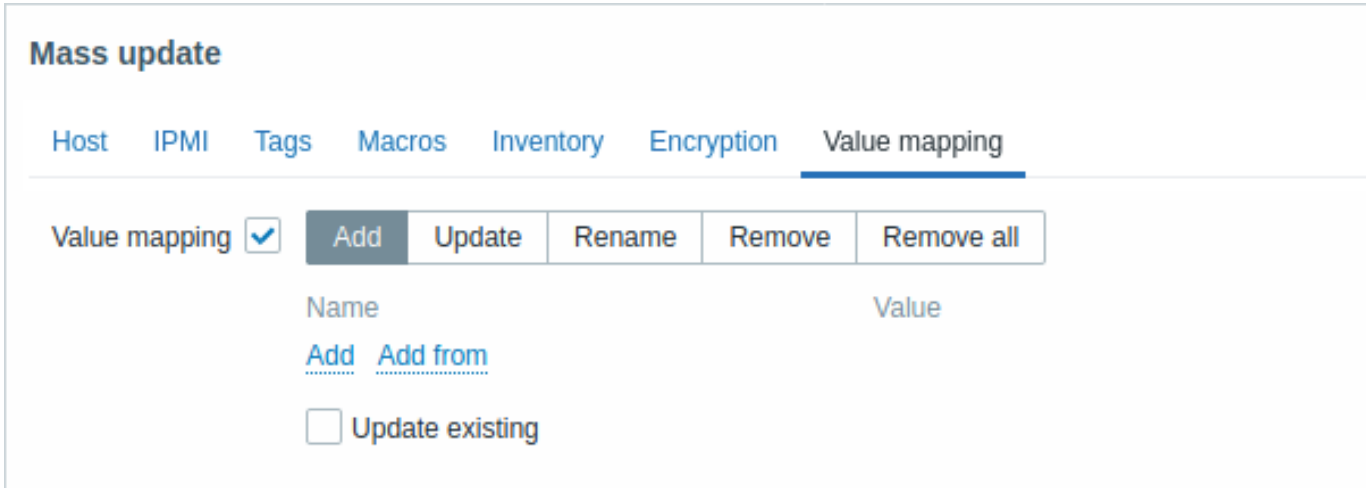

При выборе соответствующей кнопки для обновления макросов доступны следующие параметры:

- *Add* - позволяет указать дополнительные пользовательские макросы для узлов сети. Если установлен флажок *Update existing*, значение, тип и описание для указанного имени макроса будут обновлены. Если флажок снят, то если макрос с таким именем уже существует на узле сети, он не будет обновлен.
- *Update* - заменит значения, типы и описания макросов, указанных в этом списке. Если установлен флажок *Add missing*, макрос, который ранее не существовал на узле сети, будет добавлен как новый макрос. Если флажок снят, будут обновлены только макросы, которые уже существуют на узле сети.
- *Remove* - удалит указанные макросы с узлов сети. Если установлен флажок *Except selected*, будут удалены все макросы, кроме указанных в списке. Если флажок снят, будут удалены только макросы, указанные в списке.
- *Remove all* - удалит все пользовательские макросы с узлов сети. Если флажок *I confirm to remove all macros* не установлен, откроется новое всплывающее окно с запросом подтверждения удаления всех макросов.




Чтобы можно было массово обновлять поля инвентаризации, для параметра *Inventory mode* должен быть установлен режим 'Manual' или 'Automatic'.




Для обновления соответствий значений доступны кнопки со следующими параметрами:

- *Add* - добавляет соответствия значений к узлам сети. Если отметить *Update existing*, будут обновлены все свойства соответствия значений с этим именем. В противном случае, если соответствие значений с таким именем уже существует, оно не будет обновлено.
- *Update* - обновляет существующие соответствия значений. Если отметить *Add missing*, соответствие значений, которое ранее не существовало на узле сети, будет добавлено как новое соответствие значений. В противном случае будут обновлены только соответствия значений, которые уже существуют на узле сети.
- *Rename* - задает новое имя для существующего соответствия значений
- *Remove* - удаляет указанные соответствия значений с узлов сети. Если отметить *Except selected*, будут удалены все соответствия значений **кроме** указанных.
- *Remove all* - удаляет все соответствия значений с узлов сети. Если флажок *I confirm to remove all value maps* не установлен, откроется новое всплывающее окно с запросом подтверждения удаления.

Когда все необходимые изменения будут внесены, нажмите *Update*. Атрибуты будут соответствующим образом обновлены для всех выбранных узлов сети.

**2. Элементы данных**

Обзор

Элемент данных является отдельной метрикой.

Элементы данных используются для сбора данных.

Когда вы настроили узел сети, для получения реальных данных вам нужно добавить элементы данных. Один из способов быстро добавить большое количество элементов данных, — это присоединить к узлу сети один из предопределённых

шаблонов. Однако, для оптимизации производительности системы, вам может потребоваться выполнить тонкую настройку шаблонов, чтобы у вас было ровно столько элементов данных и они собирались бы настолько часто, насколько это действительно необходимо.

Чтобы указать, какого рода данные будут собираться с узла сети, используйте ключ элемента данных. Например, элемент данных с именем ключа **system.cpu.load** будет собирать данные о загрузке процессора, тогда как элемент данных с именем ключа **net.if.in** будет собирать информацию о входящем трафике.

Дополнительные параметры могут быть указаны в квадратных скобках после имени ключа. Например, system.cpu.load**[avg5]** будет возвращать загрузку процессора за последние 5 минут, а net.if.in**[eth0]** покажет входящий трафик на интерфейсе eth0.

**Note:**
Для просмотра всех поддерживаемых типов элементов данных и ключей элементов данных смотрите отдельные разделы по типам элементов данных.

Перейти к созданию и настройке элемента данных.

### 1. Создание элемента данных

Обзор

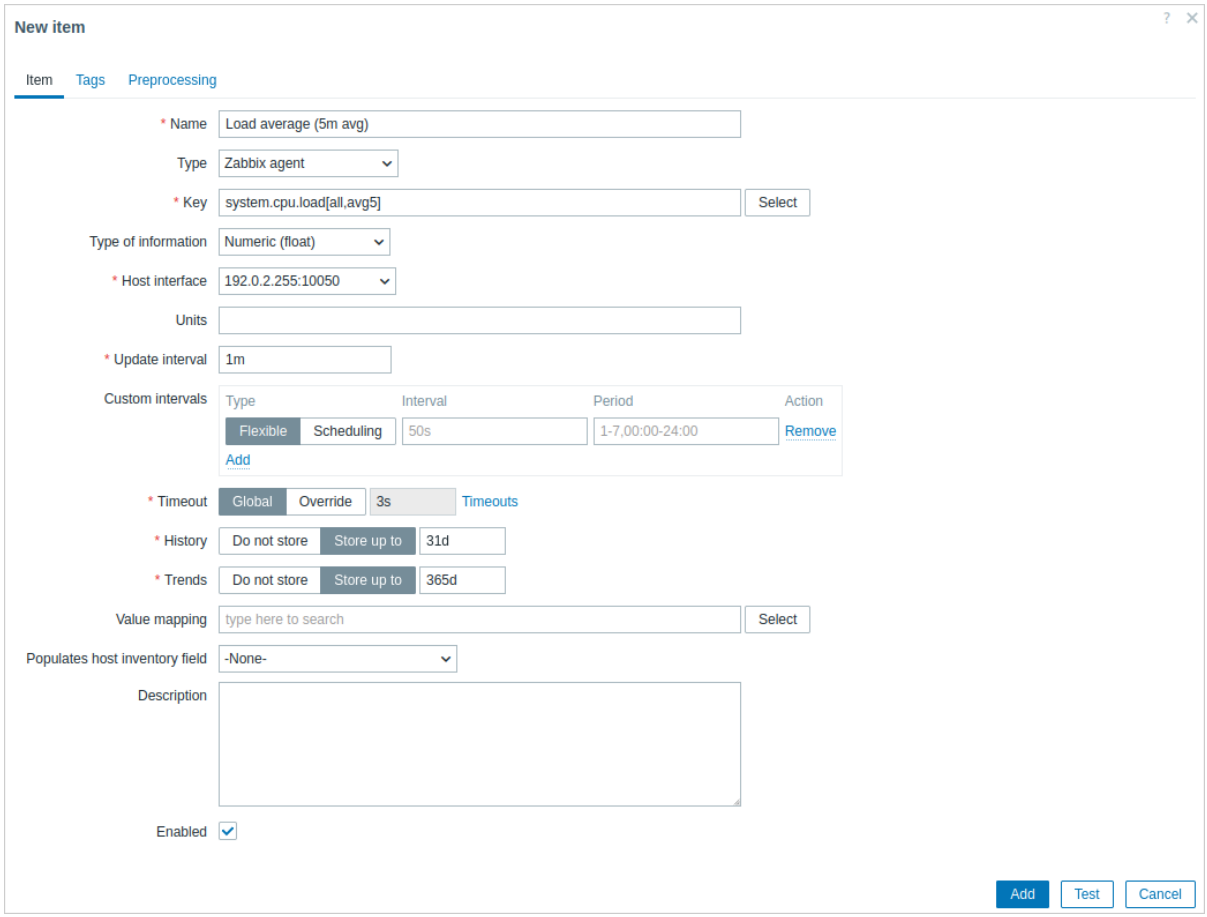
Чтобы создать элемент данных в веб-интерфейсе Zabbix, выполните следующие действия:

- Перейдите в: *Сбор данных* > *Узлы сети*
- Нажмите *Элементы данных* в строке узла сети
- Нажмите *Создать элемент данных* в правом верхнем углу экрана
- Укажите параметры элемента данных в форме

Также можно создать элемент данных, открыв существующий, нажав кнопку *Клонировать*, а затем сохранив его под другим именем.

Конфигурация

Вкладка **Элемент данных** содержит общие атрибуты элемента данных.




Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Name* | Имя элемента данных.<br>Поддерживаются пользовательские макросы. |
| *Type* | Тип элемента данных. См. отдельные разделы по типам элементов данных. |
| *Key* | Ключ элемента данных (до 2048 символов).<br>Поддерживаемые ключи элементов данных можно найти в отдельных разделах по типам элементов данных.<br>Ключ должен быть уникальным в пределах одного узла сети.<br>Если тип ключа - 'Zabbix agent', 'Zabbix agent (active)' или 'Simple check', значение ключа должно поддерживаться Zabbix agent или Zabbix server.<br>См. также: правильный формат ключа. |
| *Type of information* | Тип данных, как он хранится в базе данных после выполнения преобразований, если они есть.<br>**Numeric (unsigned)** - 64-битное беззнаковое целое число.<br>Обратите внимание, что значение с плавающей точкой, если оно получено для целочисленного элемента данных, будет усечено до дробной части.<br>**Numeric (float)** - 64-битное число с плавающей точкой.<br>Этот тип обеспечивает точность примерно до 15 цифр и диапазон примерно от -1.79E+308 до 1.79E+308 (за исключением PostgreSQL 11 и более ранних версий).<br>Также поддерживается получение значений в научной нотации. Например, 1.23E+7, 1e308, 1.1E-4.<br>**Character** - короткие текстовые данные.<br>**Log** - длинные текстовые данные с необязательными свойствами, связанными с журналом (timestamp, source, severity, logeventid).<br>**Text** - длинные текстовые данные. См. также ограничения для текстовых данных.<br>**Binary** - двоичное число (поддерживается только для зависимых элементов данных).<br>Элементы данных, возвращающие значения с двоичным типом данных, не поддерживаются в формулах вычисляемых элементов данных или выражениях триггеров.<br>Для ключей элементов данных, которые возвращают данные только в одном определенном формате, соответствующий тип информации выбирается автоматически. |
| *Host interface* | Выберите интерфейс узла сети. Это поле доступно при редактировании элемента данных на уровне узла сети. |
| *Units* | Если задан символ единицы измерения, Zabbix применяет постобработку к полученному значению элемента данных и отображает его с указанным суффиксом единицы.<br>Поддерживаются символы единиц с особым форматированием (и примеры преобразования полученного значения элемента данных → отображаемое значение):<br>**B** - байты (1024 → 1 KB)<br>**Bps** - байты в секунду (1024 → 1 KBps)<br>**s** - секунды, отображаются с использованием до трех самых больших ненулевых единиц времени (881764 → 10d 4h 56m)<br>**uptime** - прошедшее время в формате hh:mm:ss или N days, hh:mm:ss (881764 → 10 days, 04:56:04)<br>**unixtime** - Unix timestamp, форматируется как yyyy.mm.dd hh:mm:ss (881764 → 1970-01-11 04:56:04 AM); для корректного форматирования полученное значение элемента данных должно быть *Numeric (unsigned)*.<br>Для других единиц (таких как Hz, W и т. д.), если полученное значение превышает 1000, оно делится на 1000 и отображается с соответствующим префиксом (5000 → 5 KHz, 881764 → 881.76 KW).<br>Если перед символом единицы указан префикс !, преобразование единиц и добавление префикса отключаются (1024 !B → 1024 B, 61 !s → 61 s).<br>Дополнительные примеры и сведения о символах единиц и преобразовании единиц см. в разделе Суффиксы значений элементов данных. |

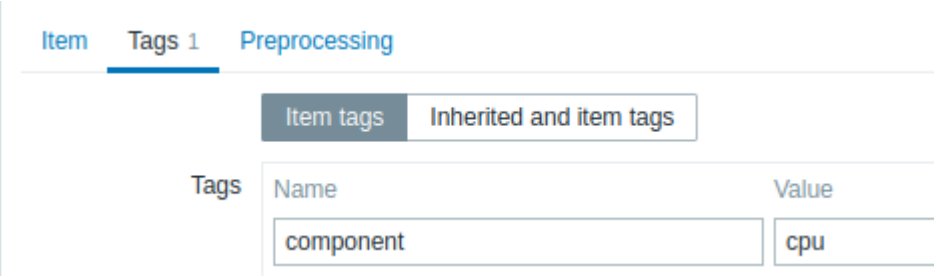
| Parameter | Description |
|---|---|
| *Update interval* | Получать новое значение для этого элемента данных каждые N секунд. Максимально допустимый интервал обновления - 86400 секунд (1 день).<br>Поддерживаются суффиксы времени, например 30s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы.<br>Одним макросом должно заполняться все поле. Несколько макросов в одном поле или макросы, смешанные с текстом, не поддерживаются.<br>*Примечание*: интервал обновления можно установить в '0' только если существуют пользовательские интервалы с ненулевым значением. Если установлено '0' и существует пользовательский интервал (гибкий или по расписанию) с ненулевым значением, опрос элемента данных будет выполняться в течение длительности пользовательского интервала.<br>*Примечание*: первый опрос элемента данных после его активации или после изменения интервала обновления может произойти раньше, чем заданное значение.<br>Новые элементы данных будут проверены в течение 60 секунд после создания, если только для них не задан интервал обновления по расписанию или гибкий интервал и для *Update interval* не установлено значение 0.<br>Существующий пассивный элемент данных можно опросить немедленно, нажав кнопку *Execute now* button. |
| *Custom intervals* | Вы можете создать пользовательские правила проверки элемента данных:<br>**Flexible** - создать исключение для *Update interval* (интервал с другой частотой).<br>**Scheduling** - создать пользовательское расписание опроса.<br>Подробную информацию см. в разделе Пользовательские интервалы.<br>В поле *Interval* поддерживаются суффиксы времени, например 30s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы.<br>Одним макросом должно заполняться все поле. Несколько макросов в одном поле или макросы, смешанные с текстом, не поддерживаются. |
| *Timeout* | Установите тайм-аут проверки элемента данных (доступно для поддерживаемых типов элементов данных). Выберите параметр тайм-аута:<br>**Global** - используется тайм-аут прокси/глобальный тайм-аут (отображается в неактивном поле *Timeout*).<br>**Override** - используется пользовательский тайм-аут (задается в поле *Timeout*; допустимый диапазон: 1 - 600s). Поддерживаются суффиксы времени, например 30s, 1m, и пользовательские макросы.<br>Нажатие ссылки *Timeouts* позволяет настроить тайм-ауты прокси или глобальные тайм-ауты (если прокси не используется). Обратите внимание, что ссылка *Timeouts* видна только пользователям типа *Super admin* с правами на разделы веб-интерфейса *Administration* > *General* или *Administration* > *Proxies*. |
| *History* | Выберите один из вариантов:<br>**Do not store** - история элемента данных не хранится. Полезно для master-элементов, если историю нужно хранить только для зависимых элементов данных.<br>Этот параметр не может быть переопределен глобальными настройками housekeeper settings.<br>**Store up to** - укажите длительность хранения подробной истории в базе данных (от 1 часа до 25 лет). Более старые данные будут удаляться housekeeper. Хранится в секундах. Поддерживаются суффиксы времени, например 2h, 1d. Поддерживаются пользовательские макросы.<br>Значение *Store up to* можно глобально переопределить в *Administration* > *Housekeeping*.<br>Если существует глобальная настройка переопределения, отображается оранжевый значок информации . Если навести на него курсор, отображается предупреждение, например *Overridden by global housekeeper settings (1d)*.<br>Рекомендуется хранить записанные значения как можно меньшее время, чтобы уменьшить размер истории значений в базе данных. Вместо хранения длинной истории значений можно хранить более длительные данные трендов.<br>См. также История и тренды. |

| Parameter | Description |
|---|---|
| *Trends* | Выберите один из вариантов:<br>**Do not store** - тренды не хранятся.<br>Этот параметр не может быть переопределен глобальными настройками housekeeper settings.<br>**Store up to** - укажите длительность хранения агрегированной истории (почасовые min, max, avg, count) в базе данных (от 1 дня до 25 лет). Более старые данные будут удаляться housekeeper. Хранится в секундах.<br>Поддерживаются суффиксы времени, например 24h, 1d. Поддерживаются пользовательские макросы.<br>Значение *Store up to* можно глобально переопределить в *Administration* > *Housekeeping*.<br>Если существует глобальная настройка переопределения, отображается оранжевый значок информации. Если навести на него курсор, отображается предупреждение, например *Overridden by global housekeeper settings (7d)*.<br>*Примечание:* хранение трендов недоступно для нечисловых данных - character, log и text.<br>См. также История и тренды. |
| *Value mapping* | Применить сопоставление значений к этому элементу данных. Сопоставление значений не изменяет полученные значения, оно используется только для отображения данных.<br>Оно работает с элементами данных *Numeric(unsigned)*, *Numeric(float)* и *Character*.<br>Например, ”Состояния служб Windows”. |
| *Log time format* | Доступно только для элементов данных типа **Log**. Поддерживаемые заполнители:<br>**y**: *Год (1970-2038)*.<br>**M**: *Месяц (01-12)*.<br>**d**: *День (01-31)*.<br>**h**: *Час (00-23)*.<br>**m**: *Минута (00-59)*.<br>**s**: *Секунда (00-59)*.<br>Если оставить поле пустым, метка времени будет установлена в 0 по Unix time, что соответствует 1 января 1970 года.<br>Например, рассмотрим следующую строку из файла журнала Zabbix agent:<br>” 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).”<br>Она начинается с шести символьных позиций для PID, затем идут дата, время и остальная часть сообщения.<br>Формат времени журнала для этой строки будет ”ppppp:yyyyMMdd:hhmmss”.<br>Обратите внимание, что символы ”p” и ”:” являются заполнителями и могут быть любыми символами, кроме ”yMdhms”. |
| *Populates host inventory field* | Вы можете выбрать поле инвентаризации узла сети, которое будет заполнено значением элемента данных. Это будет работать, если для узла сети включено автоматическое заполнение инвентаризации.<br>Это поле недоступно, если для *Type of information* установлено значение ’Log’. |
| *Description* | Введите описание элемента данных. Поддерживаются пользовательские макросы. |
| *Enabled* | Установите флажок, чтобы включить элемент данных, после чего он будет обрабатываться.<br>Когда вы отключаете элемент данных, он немедленно удаляется из кэша истории (за исключением последнего значения, которое сохраняется для журналов). |
| *Latest data* | Нажмите ссылку, чтобы просмотреть последние данные для элемента данных.<br>Эта ссылка доступна только при редактировании уже существующего элемента данных. |

**Note:**
Поля, специфичные для типа элемента данных, описаны на соответствующих страницах.

**Note:**
При редактировании существующего элемента данных уровня шаблона на уровне узла сети некоторые поля доступны только для чтения. Вы можете использовать ссылку в заголовке формы и перейти на уровень шаблона, чтобы редактировать их там, помня, что изменения на уровне шаблона изменят элемент данных для всех узлов сети, к которым привязан шаблон.

Вкладка **Теги** позволяет определить теги на уровне элемента данных.




Предобработка значений элементов данных

Вкладка **Предобработка (Preprocessing)** позволяет задать правила преобразования полученных значений.

Тестирование элемента данных

**Attention:**
Чтобы выполнить тестирование элемента данных, убедитесь, что системное время на сервере и прокси синхронизировано. Если время на сервере отстает, при тестировании элемента данных может быть возвращено сообщение об ошибке "The task has been expired." Однако разные часовые пояса на сервере и прокси не повлияют на результат тестирования.

Можно протестировать элемент данных и, если он настроен правильно, получить в ответ реальное значение.

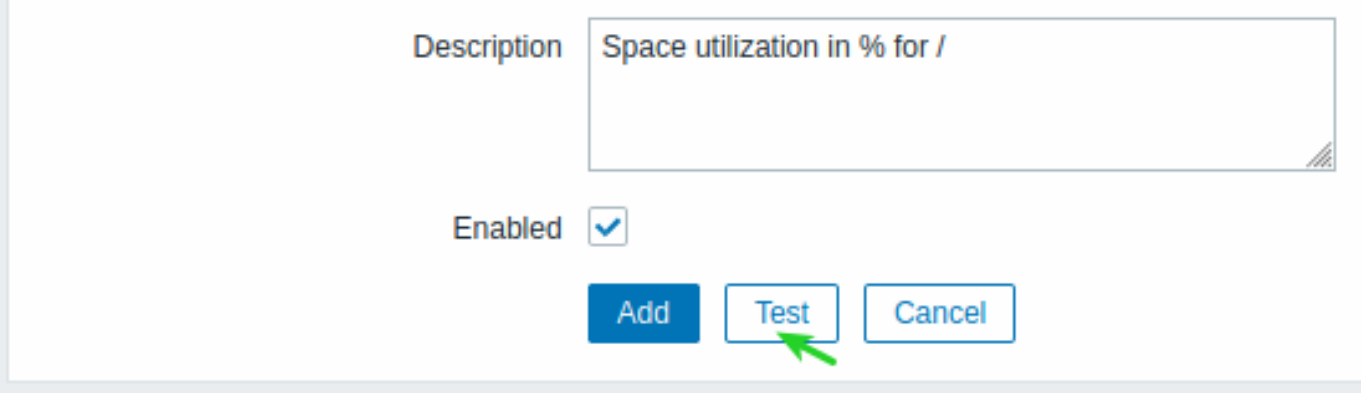
Тестирование можно выполнять даже до сохранения элемента данных.

Тестирование доступно для элементов данных узла сети и шаблона, прототипов элементов данных и правил обнаружения низкого уровня. Тестирование недоступно для активных элементов данных.

Тестирование элемента данных доступно для следующих пассивных типов элементов данных:

- 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

Чтобы протестировать элемент данных, нажмите кнопку *Test* в нижней части формы настройки элемента данных. Обратите внимание, что кнопка *Test* будет недоступна для элементов данных, которые нельзя протестировать (например, активные проверки, исключенные simple checks).

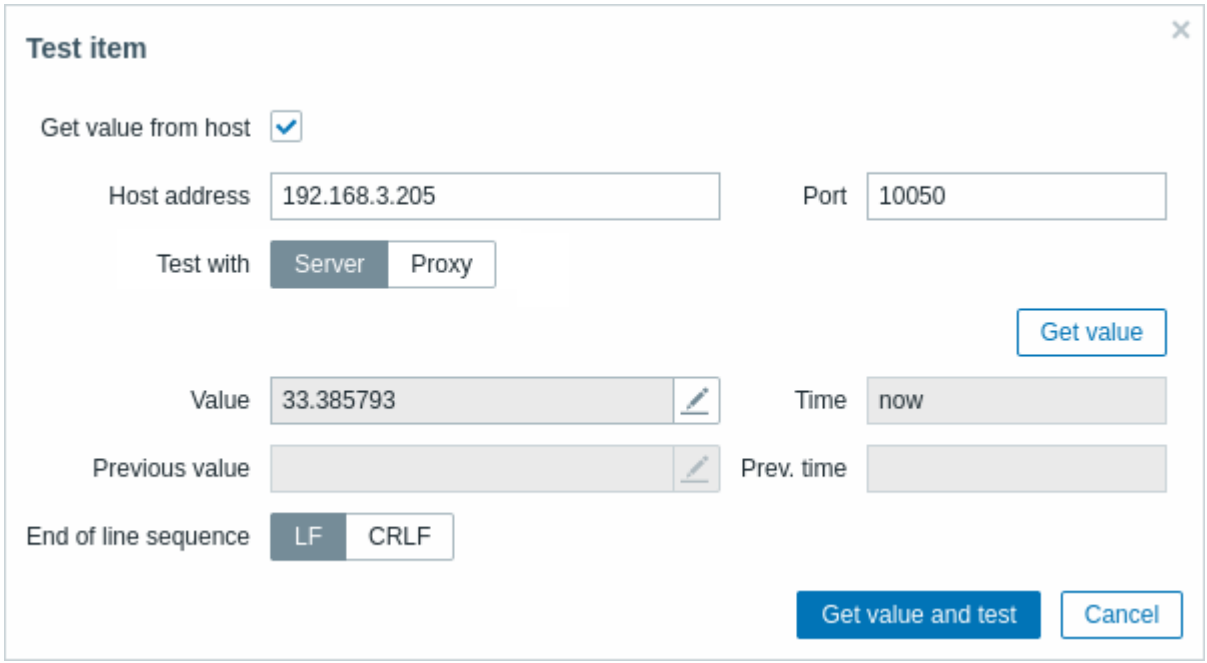



Форма тестирования элемента данных содержит поля для обязательных параметров узла сети (адрес узла сети, порт, тестирование с сервером/прокси (имя прокси)) и специфические сведения об элементе данных (например, SNMPv2 community или учетные данные безопасности SNMPv3). Эти поля учитывают контекст:

- Значения предварительно заполняются, когда это возможно, то есть для элементов данных, требующих агента, путем получения информации из выбранного интерфейса агента узла сети.
- Значения необходимо вводить вручную для элементов данных шаблона.
- Значения макросов в виде обычного текста разрешаются.
- Поля, где значение (или часть значения) является секретом или макросом Vault, остаются пустыми и должны быть введены вручную. Если какой-либо параметр элемента данных содержит значение секретного макроса, отображается следующее предупреждение: ”Item contains user-defined macros with secret values. Values of these macros should be entered manually.”
- Поля отключаются, если они не нужны в контексте типа элемента данных (например, поле адреса узла сети и поле прокси отключены для вычисляемых элементов данных)

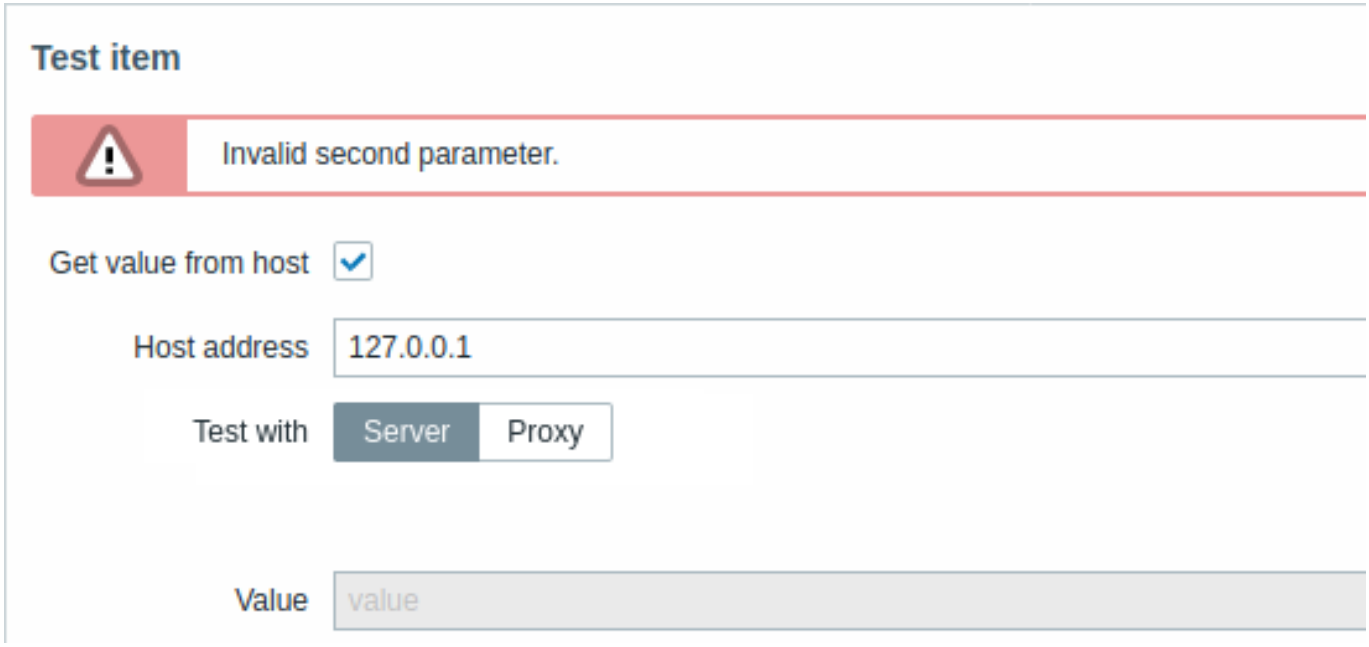
Чтобы протестировать элемент данных, нажмите *Get value*. Если значение успешно получено, оно заполнит поле *Value*, переместив текущее значение (если оно есть) в поле *Previous value*, а также вычислит поле *Prev. time*, то есть разницу во времени между двумя значениями (нажатиями), и попытается определить последовательность EOL и переключиться на CRLF, если в полученном значении обнаружено ”\n\r”.

Нажмите *Get value and test*, чтобы протестировать предварительную обработку.




Значения, полученные с узла сети, и результаты тестирования обрезаются до максимального размера 512 KB при отправке на веб-интерфейс. Если результат обрезан, отображается значок предупреждения. Описание предупреждения отображается при наведении курсора мыши. Обратите внимание, что данные объемом более 512 KB по-прежнему полностью обрабатываются сервером Zabbix.

Если конфигурация задана неверно, отображается сообщение об ошибке с описанием возможной причины.




Успешно полученное значение с узла сети также можно использовать для тестирования шагов предварительной обработки.

Кнопки диалога

Кнопки в нижней части диалога позволяют выполнить несколько видов операций.

| | |
|---|---|
| Add (Добавить) | Добавление элемента данных. Эта кнопка доступна только для новых элементов данных. |
| Update (Обновить) | Обновление свойств элемента данных. |
| Clone (Клонировать) | Создание другого элемента данных на основе свойств текущего элемента данных. |
| Execute now (Выполнить сейчас) | Выполнение немедленной проверки нового значения элемента данных. Поддерживается только для **пассивных** проверок (смотрите более подробную информацию).<br>*Обратите внимание*, что когда выполняется немедленная проверка значения, кэш конфигурации не обновляется, поэтому значение не будет отражать совсем недавние изменения настроек элемента данных. |
| Test (Тест) | Проверка корректности настройки элемента данных путём попытки получения значения. |
| Clear history and trends (Очистить историю и динамику изменений) | Удаление истории и динамики изменений элемента данных. |
| Delete (Удалить) | Удаление элемента данных. |
| Cancel (Отмена) | Отмена изменения свойств элемента данных. |

Ограничения на текстовые данные

Ограничения на текстовые данные зависят от используемой СУБД. Перед сохранением текстовых значений в базе данных они усекаются в соответствии с ограничением типа значения в базе данных:

| Database | Type of information | | |
|---|---|---|---|
| | **Character** | **Log** | **Text** |
| *MySQL* | 255 символов | 65536 байт | 65536 байт |
| *PostgreSQL* | 255 символов | 65536 символов | 65536 символов |
| *SQLite (only Zabbix proxy)* | 255 символов | 65536 символов | 65536 символов |

Ограничение пользовательского скрипта

Доступная длина пользовательского скрипта зависит от используемой базы данных:

| Database | Limit in characters | Limit in bytes |
|---|---|---|
| *MySQL* | 65535 | 65535 |
| *PostgreSQL* | 65535 | not limited |
| *SQLite (only Zabbix proxy)* | 65535 | not limited |

Таймаут элемента данных

Таймаут элемента данных определяет, как долго Zabbix должен ждать, прежде чем прервать проверку как неудачную.

Когда достигается значение таймаута, проверка прерывается, даже если получение данных еще не завершено. Если данные получены частично, элемент данных становится неподдерживаемым, и в журнале записывается сообщение об ошибке (например, если данные успешно собраны только для одного из нескольких OID в SNMP-проверке).

Для многих типов элементов данных можно задать **гибкие** таймауты элемента данных:

- для отдельного элемента данных
- для типа элемента данных (на уровне прокси)
- для типа элемента данных (на глобальном уровне)

Пользовательский таймаут *для отдельного элемента данных* полезен, если нужно указать более длительный таймаут для конкретного элемента данных, сохранив при этом низкие таймауты для остальных элементов данных.

**Приоритет таймаутов**

1. Таймаут отдельного элемента данных имеет приоритет над любыми другими таймаутами.
2. Таймауты на уровне прокси имеют приоритет над глобальными таймаутами.

Таймаут из конфигурации Zabbix сервер или прокси **не** влияет на проверки, для которых заданы гибкие таймауты.

Поддержка гибкого тайм-аута

Гибкие тайм-ауты элементов данных поддерживаются для следующих типов элементов данных:

- агент Zabbix (как пассивные, так и активные проверки)
- SNMP agent (кроме устаревших SNMP-проверок[1])
- Простая проверка (кроме `icmpping*` и элементов данных VMware[2])
- SSH agent
- Telnet agent
- Внешняя проверка
- Мониторинг базы данных
- Скрипт
- HTTP agent
- Браузер

[1] Для устаревших SNMP-проверок (опрос одного OID) применяются настройки тайм-аута из сервер или прокси.<br> [2] Для элементов данных `icmpping*` значение тайм-аута задается непосредственно в ключе элемента данных. Элементы мониторинга VMware используют собственный параметр VMwareTimeout из конфигурации сервер или прокси.

Неподдерживаемые элементы данных

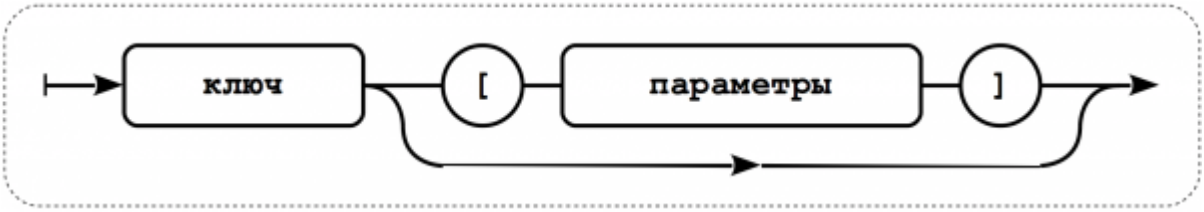
О неподдерживаемых элементах данных сообщается как об имеющих состояние *Не поддерживается*, и они по-прежнему перепроверяются со стандартным *интервалом обновления*.

Элемент данных становится неподдерживаемым, если его значение не удалось получить по какой-либо причине (например, ошибка подключения, нет настроенных поллеров для обработки элемента данных). Кроме того, элементы данных, которые не получают исторические данные, остаются в статусе *Не поддерживается*. Это ожидаемое поведение — элемент данных переходит в статус *Нормальный* только при получении новых, корректных исторических данных.

Как и проблемы, неподдерживаемые элементы данных переоцениваются только при получении новых данных — даже если для этого элемента данных больше нет исторических данных. Другими словами, элементы данных и триггеры изменяют своё состояние только при получении новых данных. Однако если выражение триггера включает функции даты и времени и/или nodata(), триггер будет пересчитываться каждые 30 секунд (подробности смотрите в разделе время расчёта триггеров).

**1. Формат ключа элемента данных**

Следуйте правилам синтаксиса в формате ключа элемента данных, включая параметры ключа. На рисунках изображен поддерживаемый синтаксис. Допустимые элементы и символы в каждой позиции можно определить, следуя стрелкам — если до какого-то блока можно добраться по линии, то он разрешён, если нет — то не разрешён.




Для построения корректного ключа элемента данных нужно начать с указания имени ключа; затем есть выбор в зависимости от того, имеет ли ключ параметры или нет, — как проиллюстрировано двумя линиями, которым можно следовать.

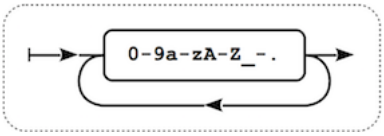
Имя ключа

Имя ключа имеет ограниченный диапазон разрешённых символов, которые просто следуют друг за другом. Разрешённые символы:

0-9a-zA-Z_-.

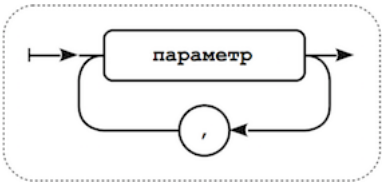
Что означает:

- все цифры;
- все латинские буквы в нижнем регистре;
- все латинские буквы в верхнем регистре;
- подчёркивание;
- минус;
- точка.




Параметры ключа

Ключ элемента данных может принимать несколько параметров, которые должны быть разделены запятой.



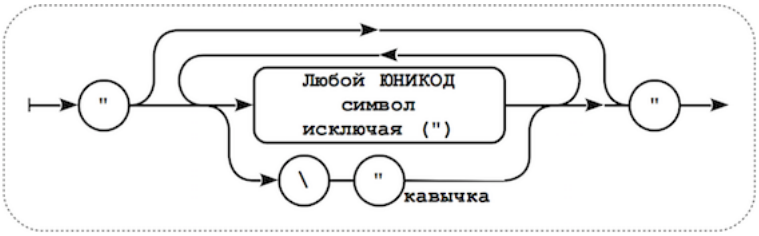

Каждый параметр ключа может быть одним из: заключённая в кавычки строка, строка без кавычек, массив.

Параметр можно оставить пустым, и тогда будет использоваться значение по умолчанию. В этом случае Вы должны добавить соответствующее количество запятых, если какой-то из дальнейших параметров все же нужно задать.

Например, ключ элемента данных **icmpping[„200„500]** будет указывать на то, что период проверки ping 200 миллисекунд, время ожидания — 500 миллисекунд, а все остальные параметры будут заполнены значениями по умолчанию.

Параметр - строка, заключённая в кавычки

Если параметр ключа — это строка, заключённая в кавычки, тогда разрешён любой символ в Юникод. Если строка параметра ключа содержит кавычки, этот параметр должен быть заключён в кавычки, а каждая кавычка, являющаяся частью строки параметра, должна экранироваться символом обратной косой черты (\). Если строка параметра ключа содержит запятую, то этот параметр должен быть заключён в кавычки.

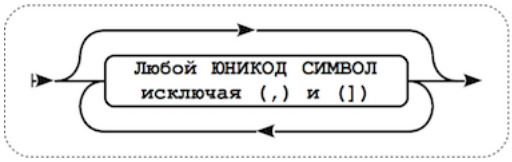



**Warning:**
Чтобы заключить параметры ключа элемента данных в кавычки, используйте только двойные кавычки. Одинарные кавычки не поддерживаются.

**Attention:**
Многоуровневые массивы параметров, например `[a,[b,[c,d]],e]`, не разрешены.

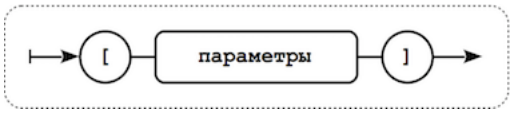
Параметр - строка, не заключённая в кавычки

Если параметр ключа — это строка без кавычек, тогда разрешён любой символ в Юникод, за исключением запятой и правой квадратной скобки (]). Параметр, который не заключён в кавычки, не может начинаться с левой квадратной скобки ([).




Параметр - массив

Если параметр ключа — это массив, тогда он должен быть заключён в квадратные скобки, в которых каждый индивидуальный параметр следует один за другим, согласно правилам и синтаксису.




### 2. Пользовательские интервалы

Обзор

Имеется возможность создания пользовательских правил относительно времени, когда элемент данных будет опрашиваться. Для этого имеется два способа: *Переменные интервалы*, позволяющий переопределить интервал обновления по умолчанию, и *По расписанию*, посредством которого элемент данных может быть опрошен в конкретное время или последовательность моментов времени.

Гибкие интервалы

Гибкие интервалы позволяют переопределять интервал обновления по умолчанию для определенных временных периодов. Гибкий интервал задается с помощью *Interval* и *Period*, где:

- *Interval* - интервал обновления для указанного временного периода. Поддерживаются суффиксы времени, например: 30s, 1m, 2h, 1d.
- *Period* - временной период, когда гибкий интервал активен (см. временные периоды для подробного описания формата *Period*)

Если несколько гибких интервалов перекрываются, для перекрывающегося периода используется наименьшее значение *Interval*. Обратите внимание, что если наименьшее значение перекрывающихся гибких интервалов равно '0', опрос выполняться не будет. Вне гибких интервалов используется интервал обновления по умолчанию.

Обратите внимание, что если гибкий интервал равен длине периода, элемент данных будет проверен ровно один раз. Если гибкий интервал больше периода, элемент данных может быть проверен один раз или не будет проверен вообще (поэтому такая конфигурация не рекомендуется). Если гибкий интервал меньше периода, элемент данных будет проверен как минимум один раз.

Если для гибкого интервала задано значение '0', элемент данных не опрашивается в течение периода гибкого интервала и возобновляет опрос в соответствии с интервалом *Update interval* по умолчанию после завершения периода. Примеры:

| Interval | Period | Description |
|---|---|---|
| 10 | 1-5,09:00-18:00 | Элемент данных будет проверяться каждые 10 секунд в рабочее время. |
| 0 | 1-7,00:00-7:00 | Элемент данных не будет проверяться ночью. |
| 0 | 7-7,00:00-24:00 | Элемент данных не будет проверяться по воскресеньям. |
| 60 | 1-7,12:00-12:01 | Элемент данных будет проверяться в 12:00 каждый день. Обратите внимание, что это использовалось как обходной путь для проверок по расписанию, и для таких проверок рекомендуется использовать интервалы планирования. |

Интервалы по расписанию

Интервалы по расписанию используются для опроса элементов данных в заданное время. Тогда как переменные интервалы разрабатывались для переопределения интервала обновления элементов данных по умолчанию, интервалы по расписанию используются для обозначения независимого расписания опроса, которое выполняется параллельно.

Интервал по расписанию задается как: `md< >wd< >h< >m< >s< >` где:

- **md** — дни месяца
- **wd** — дни недели
- **h** — часы
- **m** — минуты
- **s** — секунды

`< >` используется для того, чтобы указать значения по своему префиксу (дни, часы, минуты, секунды) и задается как: `[< >[-< >]][/< >][,< >]` где:

- `< >` и `< >` определяют диапазон соответствующих значений (включительно). Если `< >` не указывается, тогда фильтр совпадает с диапазоном `< >` — `< >`. Если `< >` также не указан, тогда фильтр совпадает со всеми возможными значениями.
- `< >` определяет пропуски количества значений через диапазон. По умолчанию значение `< >` равно 1, что означает, что все значения указанного диапазона совпадают.

Хотя добавление фильтра опционально, по крайней мере один фильтр должен использоваться. Фильтр должен определять либо диапазон, либо значение *<шага>*.

Пустой фильтр соответствует либо «0», если фильтр более низкого уровня не определён, либо, в противном случае, всем возможным значениям. Например, если фильтр часа опущен, то будет соответствовать только «0» в случае, если фильтры минут и секунд также опущены; в противном случае пустой фильтр часа будет соответствовать всем значениям часа.

Допустимые значения `< >` и `< >` по их соответствующему префиксу фильтра:

| Префикс | Описание | *<от>* | *<до>* |
|---|---|---|---|
| md | Дни месяца | 1—31 | 1—31 |
| wd | Дни недели | 1—7 | 1—7 |
| h | Часы | 0—23 | 0—23 |
| m | Минуты | 0—59 | 0—59 |
| s | Секунды | 0—59 | 0—59 |

Значение `< >` должно быть меньше или равно значению `< >`. Значение `< >` должно быть больше или равно 1 и меньше или равно `< > - < >`.

К однозначным цифрам значений дней месяца, часов, минут и секунд можно добавлять префикс 0. Например, `md01-31` и `h/02` являются допустимыми интервалами, но `md01-031` и `wd01-07` — нет.

В веб-интерфейсе Zabbix несколько интервалов по расписанию вводятся в отдельных строках. В Zabbix API они складываются в одну строку с разделителем в виде точки с запятой (`;`).

Если время совпадает по нескольким интервалам, опрос выполнится только раз. Например, `wd1h9;h9` в понедельник в 9 утра выполнится только один раз.

Примеры:

| Интервал | Когда выполняется |
|---|---|
| m0-59 | каждую минуту |
| h9-17/2 | каждые 2 часа, начиная с 9:00 (9:00, 11:00 ...) |
| m0,30 или m/30 | каждый час в чч:00 и чч:30 |
| m0,5,10,15,20,25,30,35,40,45,50,55 или m/5 | каждые пять минут |
| wd1-5h9 | каждый день с понедельника по Пятницу в 9:00 |
| wd1-5h9-18 | каждый день с понедельник по Пятницу в 9:00,10:00,...,18:00 |
| h9,10,11 или h9-11 | каждый день в 9:00, 10:00 и 11:00 |
| md1h9m30 | каждое первое число каждого месяца в 9:30 |
| md1wd1h9m30 | каждое первое число каждого месяца в 9:30, если этот день понедельник |
| h9m/30 | каждый день в 9:00, 9:30 |
| h9m0-59/30 | каждый день в 9:00, 9:30 |
| h9,10m/30 | каждый день в 9:00, 9:30, 10:00, 10:30 |
| h9-10m30 | каждый день в 9:30, 10:30 |
| h9m10-40/30 | каждый день в 9:10, 9:40 |
| h9,10m10-40/30 | каждый день в 9:10, 9:40, 10:10, 10:40 |
| h9-10m10-40/30 | каждый день в 9:10, 9:40, 10:10, 10:40 |
| h9m10-40 | каждый день в 9:10, 9:11, 9:12, ... 9:40 |
| h9m10-40/1 | каждый день в 9:10, 9:11, 9:12, ... 9:40 |
| h9-12,15 | каждый день в 9:00, 10:00, 11:00, 12:00, 15:00 |
| h9-12,15m0 | каждый день в 9:00, 10:00, 11:00, 12:00, 15:00 |
| h9-12,15m0s30 | каждый день в 9:00:30, 10:00:30, 11:00:30, 12:00:30, 15:00:30 |
| h9-12s30 | каждый день в 9:00:30, 9:01:30, 9:02:30 ... 12:58:30, 12:59:30 |
| h9m/30;h10 (*особый синтаксис API*) | каждый день в 9:00, 9:30, 10:00 |
| h9m/30<br>h10 (*добавьте это отдельной строкой в веб-интерфейсе*) | каждый день в 9:00, 9:30, 10:00 |

Выравнивание часовых поясов для прокси и агента

Обратите внимание, что Zabbix прокси и агент, когда обрабатывают интервалы по расписанию, используют свои местные часовые пояса.

По этой причине, когда интервалы по расписанию применяются к элементам данных, которые отслеживаются через прокси, либо к активным элементам данных агента, то рекомендуется настроить часовой пояс соответствующих прокси либо агента таким же, как и на сервере Zabbix, в противном случае очередь может неправильно сообщать о задержках элементов данных.

Часовой пояс для Zabbix прокси или агента можно настроить при помощи переменной окружения `TZ` в файле юнита `systemd`:

```
[Service]
...
Environment="TZ=Europe/Amsterdam"
```

#### 2. Предобработка значений элементов данных

Обзор

Предварительная обработка позволяет применять преобразования к полученным значениям элементов данных перед сохранением их в базе данных. Эти преобразования/шаги предобработки выполняются Zabbix сервером или прокси (если элементы данных отслеживаются через прокси).

Этот функционал поддерживает множество вариантов использования, таких как:

- преобразование байтов в биты (например, умножая значения сетевого трафика на «8»);

- расчёт посекундной статистики постепенно возрастающих значений;
- применение регулярных выражений для извлечения или изменения значений;
- выполнение пользовательских скриптов для обработки значений;
- отбрасывание неизменившихся значений для оптимизации хранения базы данных.

Для элемента данных можно настроить один или несколько шагов предварительной обработки. Эти шаги выполняются в том порядке, в котором они настроены.

**Attention:**
Если шаг предварительной обработки завершается неудачей, элемент данных станет неподдерживаемым. Этого можно избежать с помощью пользовательской обработки ошибок с использованием параметра *Другое при ошибке* (доступной для большинства преобразований), что позволяет вам отбрасывать значения или устанавливать пользовательские значения. <br><br> Для элементов данных типа «журнал» метаданные журнала (без значения) всегда будут сбрасывать неподдерживаемое состояние элемента данных и снова делать элемент данных поддерживаемым. Это происходит, даже если первоначальная ошибка произошла после получения значения журнала от агента.

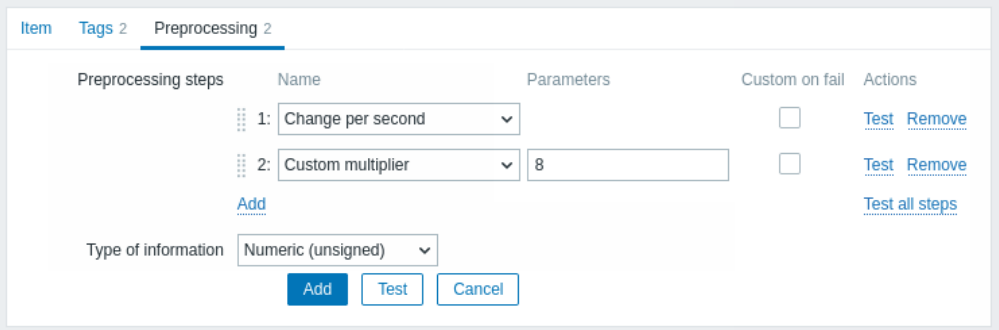
Все значения, передаваемые в предварительную обработку, изначально рассматриваются как строки. Преобразование в требуемый тип значения (как определено в конфигурации элемента данных) выполняется в конце конвейера предобработки. Однако, определённые шаги предварительной обработки могут инициировать более ранние преобразования, если это необходимо. Подробную техническую информацию смотрите в разделе Подробности предварительной обработки.

Чтобы убедиться, что ваша конфигурация предварительной обработки работает так, как ожидается, можно её протестировать.

Смотрите также: Примеры предварительной обработки

Конфигурация

Шаги предварительной обработки задаются на вкладке **Предварительная обработка** формы конфигурации элемента данных.




Нажмите *Добавить*, чтобы выбрать поддерживаемое преобразование.

Поле *Тип информации* отображается в нижней части вкладки, когда задан хотя бы один шаг предварительной обработки. При необходимости можно изменить тип информации, не покидая вкладку *Предварительная обработка*. См. Создание элемента данных для подробного описания параметров.

Поддерживаемые преобразования

Все поддерживаемые преобразования перечислены ниже. Нажмите на название преобразования, чтобы увидеть его полные сведения.

| Name | Description | Type |
|---|---|---|
| Regular expression | Сопоставляет значение с регулярным выражением и заменяет его требуемым результатом. | Text |
| Replace | Находит строку поиска и заменяет ее другой строкой (или ничем). | |
| Trim | Удаляет указанные символы с начала и конца значения. | |
| Right trim | Удаляет указанные символы с конца значения. | |
| Left trim | Удаляет указанные символы с начала значения. | |
| XML XPath | Извлекает значение или фрагмент из данных XML с помощью функциональности XPath. | Structured data |
| JSON Path | Извлекает значение или фрагмент из данных JSON с помощью функциональности JSONPath. | |
| CSV to JSON | Преобразует данные CSV-файла в формат JSON. | |

| Name | Description | Type |
|---|---|---|
| XML to JSON | Преобразует данные в формате XML в JSON. | |
| SNMP walk value | Извлекает значение по указанному имени OID/MIB и применяет параметры форматирования. | SNMP |
| SNMP walk to JSON | Преобразует значения SNMP в JSON. | |
| SNMP get value | Применяет параметры форматирования к значению SNMP get. | |
| Custom multiplier | Умножает значение на указанное целое или число с плавающей точкой. | Arithmetic |
| Simple change | Вычисляет разницу между текущим и предыдущим значением. | Change |
| Change per second | Вычисляет скорость изменения значения (разницу между текущим и предыдущим значением) в секунду. | |
| Boolean to decimal | Преобразует значение из булевого формата в десятичный. | Numeral systems |
| Octal to decimal | Преобразует значение из восьмеричного формата в десятичный. | |
| Hexadecimal to decimal | Преобразует значение из шестнадцатеричного формата в десятичный. | |
| JavaScript | Введите код JavaScript. | Custom scripts |
| In range | Определяет диапазон, в котором должно находиться значение. | Validation |
| Matches regular expression | Указывает регулярное выражение, которому должно соответствовать значение. | |
| Does not match regular expression | Указывает регулярное выражение, которому значение не должно соответствовать. | |
| Check for error in JSON | Проверяет наличие сообщения об ошибке уровня приложения, расположенного по JSONPath. | |
| Check for error in XML | Проверяет наличие сообщения об ошибке уровня приложения, расположенного по XPath. | |
| Check for error using a regular expression | Проверяет наличие сообщения об ошибке уровня приложения с помощью регулярного выражения. | |
| Check for not supported value | Проверяет, удалось ли получить значение элемента данных. | |
| Discard unchanged | Отбрасывает значение, если оно не изменилось. | Throttling |
| Discard unchanged with heartbeat | Отбрасывает значение, если оно не изменилось в течение заданного периода времени. | |
| Prometheus pattern | Использует следующий запрос для извлечения требуемых данных из метрик Prometheus. | Prometheus |
| Prometheus to JSON | Преобразует требуемые метрики Prometheus в JSON. | |

Обратите внимание, что для шагов предварительной обработки *Change* и *Throttling* Zabbix должен запоминать последнее значение, чтобы вычислить/сравнить новое значение требуемым образом. Эти предыдущие значения обрабатываются менеджером предварительной обработки. Если сервер или прокси Zabbix перезапущен либо в шаги предварительной обработки внесены какие-либо изменения, последнее значение соответствующего элемента данных сбрасывается, в результате чего:

- для шагов *Simple change*, *Change per second* - следующее значение будет проигнорировано, поскольку отсутствует предыдущее значение, от которого можно вычислить изменение;
- для шагов *Discard unchanged*, *Discard unchanged with heartbeat* - следующее значение никогда не будет отброшено, даже если оно должно было быть отброшено согласно правилам отбрасывания.

Регулярное выражение (Regular expression)

Сопоставить значение с регулярным выражением и заменить в соответствии с требуемым выводом.

Параметры:

- **шаблон (pattern)** — регулярное выражение;<br>
- **вывод (output)** — шаблон форматирования вывода. Управляющая последовательность \N (где N=1...9) заменяется N-ой совпадающей группой. Управляющая последовательность \0 заменяется совпавшим текстом.

Комментарии:

- При ошибке соответствия входящего значения элемент данных станет неподдерживаемым;<br>
- Регулярное выражение поддерживает извлечение до 10 скобочных групп в \N последовательности;<br>
- Если выбрана опция *Другое при ошибке*, то можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного

завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.<br>
- Пожалуйста, обратитесь к разделу регулярные выражения для ознакомления с некоторыми существующими примерами.

Замена (Replace)

Найти искомую строку и заменить её на другую (или на пустую строку).

Параметры:

- **строка поиска (search string)** — строка для поиска и замены, с учётом регистра (обязательно);<br>
- **замена (replacement)** — строка для замены строки поиска. Строка замены также может быть пустой, что позволяет эффективно удалять строку поиска при ее обнаружении.

Комментарии:

- Будут заменены все вхождения строки поиска;
- Можно использовать управляющие последовательности для поиска или замены переводов строк, возврата каретки, табуляции и пробелов «\n \r \t \s»; обратную косую черту можно экранировать как «\\», а управляющие последовательности можно экранировать как «\\n»;
- Экранирование переводов строк, возвратов каретки и табуляций выполняется автоматически при низкоуровневом обнаружении.

Обрезать (Trim)

Удалить указанные символы с начала и конца значения.

Обрезать справа (Right trim)

Удалить указанные символы с конца значения.

Обрезать слева (Left trim)

Удалить указанные символы с начала значения.

XML XPath

Извлечь значение или фрагмент из данных XML с использованием функционала XPath.

Комментарии:

- Чтобы эта опция работала, Zabbix сервер (или Zabbix прокси) должен быть скомпилирован с поддержкой libxml;<br>
- Пространства имён не поддерживаются;<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Примеры:

```
number(/document/item/value) #     «10»   <document><item><value>10</value></item></document>
number(/document/item/@attribute) #     «10»   <document><item attribute="10"></item></document>
/document/item #     «<item><value>10</value></item>»   <document><item><value>10</value></item></document
```

JSON Path

Извлечь значение или фрагмент из данных JSON с использованием функционала JSONPath.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

CSV в JSON

Преобразует данные CSV-файла в формат JSON.

Дополнительную информацию см. в разделе: Предобработка CSV в JSON.

XML в JSON

Сконвертировать данные в формате XML в JSON.

Для получения дополнительной информации смотрите: Правила сериализации.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Значение SNMP walk (SNMP walk value)

Извлечь значение по указанному OID или имени MIB и применить параметры форматирования:<br>

- **Без изменений (Unchanged)** — вернуть шестнадцатеричную строку как неэкранированную шестнадцатеричную строку (*обратите внимание*, что подсказки отображения всё ещё применяются);<br>
- **UTF-8 из Hex-STRING (UTF-8 from hex-STRING)** — преобразовать шестнадцатеричную строку в строку UTF-8;<br>
- **MAC из Hex-STRING (MAC from hex-STRING)** — проверить шестнадцатеричную строку как MAC-адрес и вернуть правильную строку MAC-адреса (где `' '` заменяются на `':'`);<br>
- **Целое число из BITS (Integer from BITS)** — преобразовать первые 8 байтов битовой строки, выраженной как последовательность шестнадцатеричных символов (например, «1A 2B 3C 4D»), в 64-битное беззнаковое целое число. В битовых строках длиннее 8 байтов последующие байты будут игнорироваться.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

SNMP walk в JSON

Преобразовать значения SNMP в JSON.

Укажите имя поля в JSON и соответствующий путь SNMP OID. Значения полей будут заполнены значениями в указанном пути SNMP OID.

Комментарии:

- Доступны параметры форматирования значений, аналогичные шагу *Значение SNMP walk*;<br>
- Можно использовать этот шаг предварительной обработки для обнаружения SNMP OID;<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

SNMP get значение (SNMP get value)

Применить параметры форматирования к SNMP get значение:<br>

- **UTF-8 из Hex-STRING (UTF-8 from Hex-STRING)** — преобразовать hex-строку в строку UTF-8;<br>
- **MAC из Hex-STRING (MAC from Hex-STRING)** — проверить hex-строку как MAC-адрес и вернуть правильную строку MAC-адреса (где `' '` заменяются на `':'`);<br>
- **Целое число из BITS (Integer from BITS)** — преобразовать первые 8 байт битовой строки, выраженной как последовательность шестнадцатеричных символов (например, «1A 2B 3C 4D»), в 64-битное беззнаковое целое число. В битовых строках длиннее 8 байт последующие байты будут игнорироваться

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Пользовательский множитель (Custom multiplier)

Умножить значение на указанное целое число или значение с плавающей точкой.

Комментарии:

- Используйте эту опцию для преобразования значений, полученных в КБ, Мбит/сек, и т.д. в Б, бит/сек. В противном случае Zabbix не сможет корректно подставить префиксы (K, M, G и т.д.).<br>
- Обратите внимание, если тип элемента информации *Числовой (целое положительное)*, входящие значения с дробной частью будут обрезаны (т. е. «0,9» станет «0») до применения пользовательского множителя;<br>
- Если вы используете пользовательский множитель или сохраняете значение как *Изменение в секунду* для элементов данных с типом информации *Числовой (без знака)*, и полученное вычисленное значение фактически является числом с плавающей точкой, вычисленное значение все равно принимается как правильное путём обрезания десятичной части и сохранения значения как целого числа;<br>

- Поддерживаются: научное представление, например, 1e+70; пользовательские макросы и LLD макросы; строки, которые включают в себя макросы, например: `{#MACRO}e+10`, `{$MACRO1}e+{$MACRO2}`. Эти макросы должны раскрываться в целое число или число с плавающей точкой.
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Простое изменение (Simple change)

Вычислить разницу между текущим и предыдущим значениями.

Комментарии:

- Этот шаг может быть полезен для измерения постоянно растущих значений;<br>
- Вычисляется **значение**-**пред_значение**, где *значение* — текущее значение; *пред_значение* — предыдущее полученное значение;<br>
- На один элемент данных разрешена только одна операция изменения («Простое изменение» или «Изменение в секунду»);
- Если текущее значение меньше предыдущего, тогда Zabbix отбрасывает это изменение (ничего не записывает) и ждёт следующего значения;<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Изменение в секунду

Вычисляет скорость изменения значения (разницу между текущим и предыдущим значением) в секунду.

Комментарии:

- Этот шаг полезен для вычисления скорости изменения в секунду для постоянно растущего значения;<br>
- Поскольку это вычисление может давать числа с плавающей точкой, рекомендуется установить для поля *Тип информации* значение *Numeric (float)*, даже если входные необработанные значения являются целыми числами. Это особенно важно для небольших чисел, где имеет значение дробная часть. Если значения с плавающей точкой велики и могут превысить длину поля *float*, в результате чего может быть потеряно все значение, на самом деле рекомендуется использовать *Numeric (unsigned)* и тем самым обрезать только дробную часть;<br>
- Вычисляется как (**value**-**prev_value**)/(**time**-**prev_time**), где *value* - текущее значение; *prev_value* - ранее полученное значение; *time* - текущая временная метка; *prev_time* - временная метка предыдущего значения;<br>
- Для элемента данных допускается только одна операция изменения (”Simple change” или ”Change per second”);
- Если текущее значение меньше предыдущего, Zabbix отбрасывает эту разницу (ничего не сохраняет) и ожидает следующее значение. Это помогает корректно работать, например, при переполнении 32-битных счетчиков SNMP.<br>
- Если вы отметите флажок *Custom on fail*, можно указать собственные параметры обработки ошибок: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. В случае сбоя шага предварительной обработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или задать указанное значение.

Логический в десятичный (Boolean to decimal)

Преобразовать значение из логического формата в десятичный.

Комментарии:

- Текстовое представление значения переводится в значения 0 или 1. Таким образом, «TRUE» сохранится как 1, а «FALSE» сохранится как 0. Во всех сравнениях регистр символов не учитывается. В данное время распознаются значения: для *TRUE* — true, t, yes, y, on, up, running, enabled, available, ok, master; для *FALSE* — false, f, no, n, off, down, unused, disabled, unavailable, err, slave. Кроме того, любое ненулевое числовое значение считается TRUE, нулевое считается FALSE.<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Восьмеричное в десятичное (Octal to decimal)

Преобразовать значение из восьмеричного формата в десятичный.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Шестнадцатеричное в десятичное (Hexadecimal to decimal)

Преобразовать значение из шестнадцатеричного формата в десятичный.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

JavaScript

Введите код JavaScript в модальном редакторе, который открывается при нажатии на поле параметра или на значок карандаша рядом с ним.

> **Attention:**
> Не используйте необъявленные присваивания в предварительной обработке JavaScript. Используйте `var` для объявления локальных переменных.

Комментарии:

- Доступная длина JavaScript зависит от используемой базы данных;<br>
- Для получения дополнительной информации смотрите: Предобработка Javascript.

По диапазону (In range)

Определить диапазон, в котором должно находиться значение, указав минимальное/максимальное значения (включительно).

Комментарии:

- Допускаются числовые значения (включая любое количество цифр, опционально десятичную часть и экспоненциальную часть, отрицательные значения);<br>
- Минимальное значение должно быть меньше максимального;<br>
- Должно существовать хотя бы одно значение;<br>
- Можно использовать пользовательские макросы и макросы низкоуровневого обнаружения;<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Соответствие регулярному выражению (Matches regular expression)

Указать регулярное выражение, которому должно соответствовать значение.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Не совпадает с регулярным выражением (Does not match regular expression)

Указать регулярное выражение, которому значение не должно соответствовать.

Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Проверка на ошибку в JSON (Check for error in JSON)

Проверить, нет ли сообщения об ошибке на уровне приложения в JSONPath. Обработка будет остановлена в случае положительного результата (сообщение присутствует и непустое); в противном случае обработка будет продолжена со значением, подготовленным до этого этапа предварительной обработки.

Комментарии:

- Эти внешние сервисные ошибки сообщаются пользователю напрямую, без добавления информации о шаге предварительной обработки;<br>
- Об ошибке не будет сообщено в случае неудачного анализа некорректного JSON;<br>

- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Проверка на ошибку в XML (Check for error in XML)

Проверить, нет ли сообщения об ошибке на уровне приложения в XPath. Обработка будет остановлена в случае положительного результата (сообщение присутствует и непустое); в противном случае обработка будет продолжена со значением, подготовленным до этого этапа предварительной обработки.

Комментарии:

- Эти внешние сервисные ошибки сообщаются пользователю напрямую, без добавления информации о шаге предварительной обработки;<br>
- Об ошибке не будет сообщено в случае неудачного анализа некорректного XML;<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Проверка на ошибку с использованием регулярного выражения (Check for error using a regular expression)

Проверить, нет ли сообщения об ошибке на уровне приложения с помощью регулярного выражения. Обработка будет остановлена в случае положительного результата (сообщение присутствует и непустое); в противном случае обработка будет продолжена со значением, подготовленным до этого этапа предварительной обработки.

Параметры:

- **шаблон (pattern)** — регулярное выражение;<br>
- **вывод (output)** — шаблон форматирования вывода. Управляющая последовательность \N (где N = 1…9) заменяется N-ной совпадающей группой. Управляющая последовательность \0 заменяется совпавшим текстом.

Комментарии:

- Эти внешние сервисные ошибки сообщаются пользователю напрямую, без добавления информации о шаге предварительной обработки;<br>
- Если выбрана опция *Другое при ошибке*, можно указать пользовательские варианты обработки ошибок: либо отбросить значение, либо задать нужное значение, либо задать нужное сообщение об ошибке. В случае неудачного завершения шага предобработки элемент данных не станет неподдерживаемым, если выбран вариант отбросить значение или выставить в указанное значение.

Проверка на неподдерживаемое значение (Check for not supported value)

Проверить факт того, что не удалось извлечь значение элемента данных. Укажите, как следует обрабатывать ошибку, на основе проверки возвращённого сообщения об ошибке.

Параметры:

- **область (scope)** — выберите область обработки ошибок:<br>*любая ошибка (any error)* — любая ошибка;<br>*ошибка соответствует (error matches)* — только ошибка, которая соответствует регулярному выражению, указанному в *шаблоне*;<br>*ошибка не соответствует (error does not match)* — только ошибка, которая не соответствует регулярному выражению, указанному в *шаблоне*<br>
- **шаблон (pattern)** — регулярное выражение для сопоставления ошибки. Если в параметре *область* выбрано *любая ошибка*, это поле не отображается. Если отображается, это поле является обязательным.<br>

Комментарии:

- Обычно отсутствие/неудача при получении значения приводит к тому, что элемент данных становится неподдерживаемым. Этот шаг предварительной обработки позволяет изменить это поведение. Если отметить флажок *Другое при ошибке* (для этого шага предварительной обработки он всегда отмечен и неактивен), то можно указать пользовательские параметры обработки ошибок: либо отбросить значение, либо задать указанное значение, либо установить указанное сообщение об ошибке. В случае неудачного шага предварительной обработки элемент данных не станет неподдерживаемым, если выбрана опция отбросить значение или установить указанное значение.
- Этот шаг предобработки проверяет только факт того, что не удалось извлечь значение элемента данных. Он не проверяет, например, соответствует ли тип полученного значения (скажем, строка) типу информации элемента данных (скажем, числовой); подробности смотрите Примерах предобработки. Если есть несоответствие типов, элемент данных всё равно может стать неподдерживаемым после выполнения всех шагов предобработки. Для проверки несоответствия типов можно использовать, например, шаг предобработки *Пользовательский множитель*; смотрите Примеры предварительной обработки.

- В полях *Задать значением* или *Задать ошибку в* поддерживаются скобочные группы регулярных выражений. Используйте управляющую последовательность \N (где N=1…9), чтобы извлечь N-ю совпавшую группу, используйте управляющую последовательность \0 для извлечения всего совпавшего текста.
- Эти шаги всегда выполняются первыми шагами предварительной обработки и размещаются над всеми остальными после сохранения изменений в элементе данных.
- Поддерживаются несколько шагов *Проверка на неподдерживаемое значение* в указанном порядке. Шаг для *любой ошибки* будет автоматически помещён последним шагом в этой группе.

Отбрасывать не изменившееся (Discard unchanged)

Отбросить значение, если оно не изменилось.

Комментарии:

- Если значение отбрасывается, оно не сохраняется в базе данных и Zabbix сервер не будет знать, что значение было получено. Выражения триггеров не будут пересчитываться, в результате не будут созданы/закрыты проблемы по связанным триггерам. Функции триггеров будут работать только на основе данных, которые фактически сохраняются в базу данных. Поскольку динамика изменений строится на основе данных в базе данных, если в течение часа не сохраняются значения, то для этого часа также не будет данных динамики изменений.<br>
- Для элемента данных можно указать только один параметр троттлинга.

Отбрасывать не изменившееся с периодическим контролем (Discard unchanged with heartbeat)

Отбросить значение, если оно не изменилось в течение заданного периода (в секундах).

Комментарии:

- Поддерживаются положительные целые значения для секунд (минимально — 1 секунда);<br>
- Можно использовать суффиксы времени (такие как 30s, 1m, 2h, 1d);<br>
- Можно использовать пользовательские макросы и макросы низкоуровневого обнаружения;<br>
- Если значение отбрасывается, оно не сохраняется в базе данных и Zabbix сервер не будет знать, что это значение было получено. Выражения триггеров не будут пересчитываться, в результате не будут созданы/закрыты проблемы по связанным триггерам. Функции триггеров будут работать только на основе данных, которые фактически сохраняются в базе данных. Поскольку динамика изменения строится на основе данных в базе данных, если в течение часа не сохраняются значения, то для этого часа также не будет данных о динамике изменений.<br>
- Для элемента данных можно указать только один параметр троттлинга.

Шаблон Prometheus

Используйте следующий запрос для извлечения необходимых данных из метрик Prometheus.

Дополнительные сведения см. в разделе Проверки Prometheus.

Prometheus в JSON

Преобразует необходимые метрики Prometheus в JSON.

Подробнее см. в разделе Проверки Prometheus.

Поддержка макросов

Пользовательские макросы и пользовательские макросы с контекстом поддерживаются в:

- параметрах шагов предварительной обработки, включая код JavaScript;
- параметрах обработки пользовательских ошибок (поля *Задать значением* и *Задать ошибку в*).

**Note:**
Контекст макроса игнорируется, когда макрос заменяется своим значением. Значение макроса подставляется в код как есть, без возможности добавления дополнительного экранирования до помещения значения в код JavaScript. Обратите внимание, что в некоторых случаях это может привести к ошибкам JavaScript.

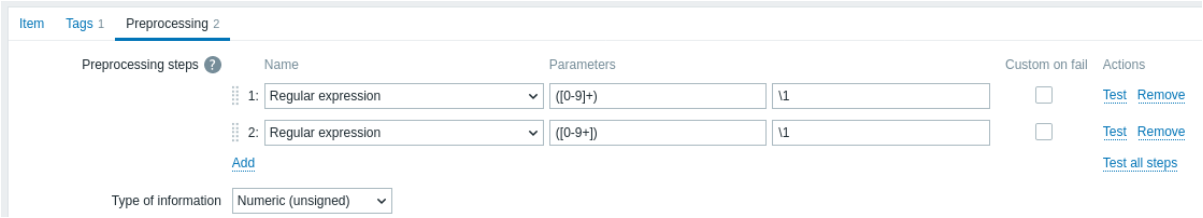
Тестирование

Смотрите тестирование предварительной обработки.

**1. Тестирование предварительной обработки**

Тестирование

Тестирование шагов предварительной обработки полезно, чтобы убедиться, что сложные конвейеры предварительной обработки дают ожидаемые результаты, не дожидаясь получения и предварительной обработки значения элемента данных.

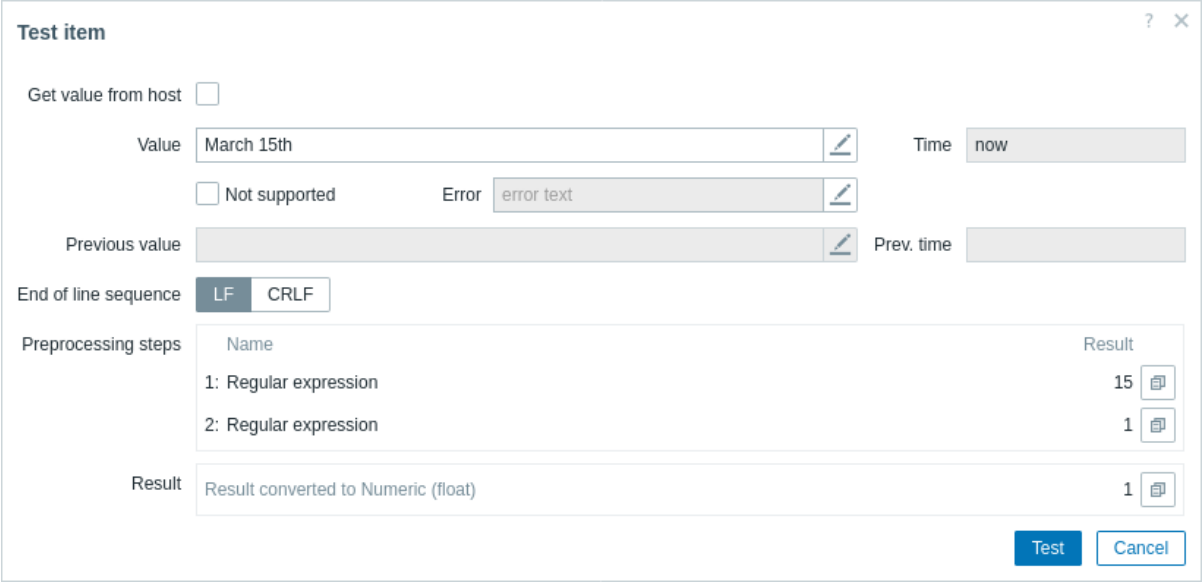



Можно выполнить тестирование:

- на основе гипотетического значения
- на основе реального значения с узла сети

Каждый шаг предварительной обработки можно тестировать отдельно, а также все шаги можно протестировать вместе. При нажатии кнопки *Test* или *Test all steps* соответственно в блоке Actions открывается окно тестирования.

Проверка гипотетического значения




| Parameter | Description |
|---|---|
| *Get value from host* | Если вы хотите проверить гипотетическое значение, оставьте этот флажок снятым.<br>См. также: Проверка реального значения. |
| *Value* | Введите значение для проверки.<br>Щелчок в поле параметра или по кнопке просмотра/редактирования откроет окно текстовой области для ввода значения или блока кода. |
| *Not supported* | Установите этот флажок, чтобы проверить неподдерживаемое значение.<br>Этот параметр полезен для проверки шага предварительной обработки *Check for not supported value*. |
| *Error* | Введите текст ошибки.<br>Это поле доступно, когда флажок *Get value from host* снят, а *Not supported* установлен.<br>Если флажок *Get value from host* установлен, в это поле подставляется фактическое сообщение об ошибке (только для чтения) с узла сети. |
| *Time* | Отображается время входного значения: `now` (только для чтения). |
| *Previous value* | Введите предыдущее входное значение для сравнения.<br>Только для шагов предварительной обработки *Change* и *Throttling*. |
| *Previous time* | Введите время предыдущего входного значения для сравнения.<br>Только для шагов предварительной обработки *Change* и *Throttling*.<br>Значение по умолчанию основано на значении поля 'Update interval' элемента данных (если указано '1m', то в это поле подставляется `now-1m`). Если ничего не указано или у пользователя нет доступа к узлу сети, по умолчанию используется `now-30s`. |
| *Macros* | Если используются какие-либо макросы, они перечисляются вместе со своими значениями. Значения можно редактировать для целей тестирования, но изменения будут сохранены только в рамках контекста тестирования. |
| *End of line sequence* | Выберите последовательность конца строки для многострочных входных значений:<br>**LF** - последовательность LF (перевод строки)<br>**CRLF** - последовательность CRLF (возврат каретки + перевод строки). |

| Parameter | Description |
|---|---|
| *Preprocessing steps* | Перечислены шаги предварительной обработки; результат тестирования отображается для каждого шага после нажатия кнопки *Test*.<br>Результаты тестирования обрезаются до максимального размера 512 KB при отправке на веб-интерфейс. Результаты тестирования можно копировать (не более обрезанных 512 KB). Если результат обрезан, отображается значок предупреждения. Описание предупреждения показывается при наведении курсора. Обратите внимание, что данные объемом более 512 KB по-прежнему полностью обрабатываются сервером Zabbix.<br>Если при тестировании шаг завершился с ошибкой, отображается значок ошибки. Описание ошибки показывается при наведении курсора.<br>Если для шага указан параметр "Custom on fail" и это действие выполняется, сразу после строки шага тестирования предварительной обработки появляется новая строка, показывающая, какое действие было выполнено и какой результат оно дало (ошибка или значение). |
| *Result* | Итоговый результат тестирования шагов предварительной обработки отображается во всех случаях, когда все шаги тестируются вместе (при нажатии кнопки *Test all steps*). Также отображается тип преобразования к типу значения элемента данных, например `Result converted to Numeric (unsigned)`.<br>Результаты тестирования обрезаются до максимального размера 512 KB при отправке на веб-интерфейс. Итоговый результат можно копировать (не более обрезанных 512 KB). Если результат обрезан, отображается значок предупреждения. Описание предупреждения показывается при наведении курсора. Обратите внимание, что данные объемом более 512 KB по-прежнему полностью обрабатываются сервером Zabbix. |

Нажмите *Test*, чтобы увидеть результат после каждого шага предварительной обработки.

Тестовые значения сохраняются между сеансами тестирования для отдельных шагов или для всех шагов, что позволяет пользователю изменять шаги предварительной обработки или конфигурацию элемента данных, а затем возвращаться в окно тестирования без необходимости повторно вводить информацию. Однако при обновлении страницы значения теряются.

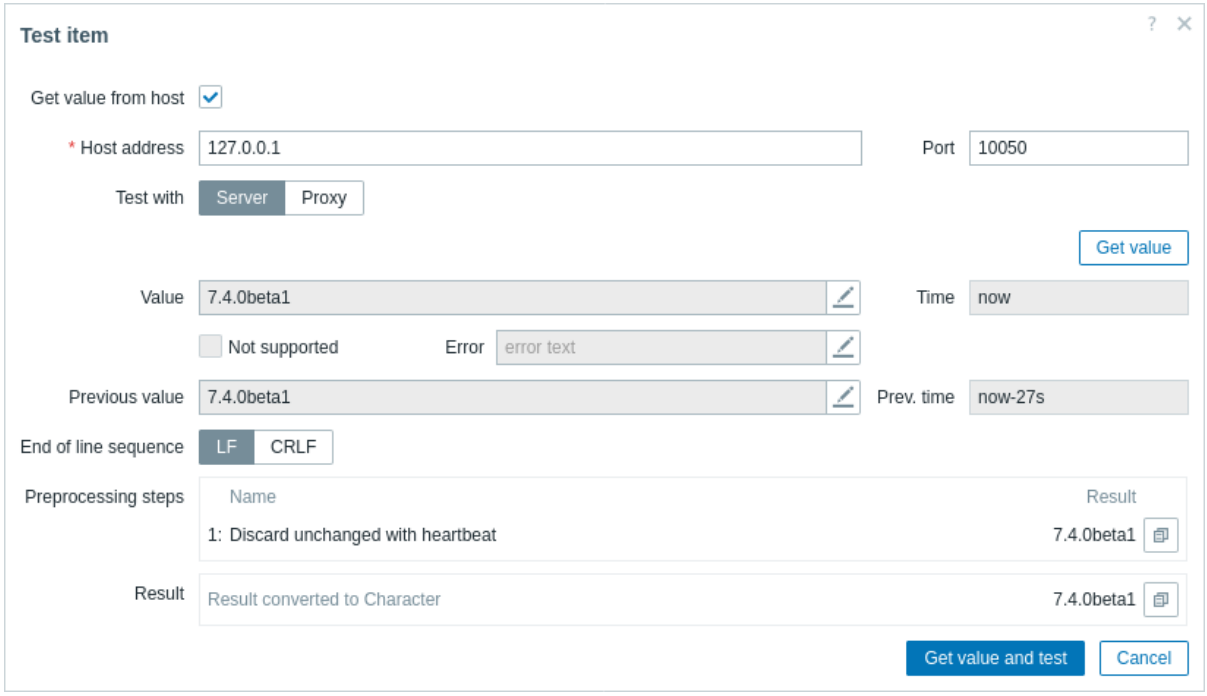
Тестирование выполняется сервером Zabbix. Веб-интерфейс отправляет соответствующий запрос на сервер и ожидает результат. Запрос содержит входное значение и шаги предварительной обработки (с раскрытыми пользовательскими макросами). Для шагов *Change* и *Throttling* можно указать необязательные предыдущее значение и время. Сервер возвращает результаты для каждого шага предварительной обработки.

Все технические ошибки или ошибки проверки входных данных отображаются в поле ошибок в верхней части окна тестирования.

Проверка реального значения

Чтобы проверить предварительную обработку на реальном значении:

- Установите флажок *Get value from host*
- Введите или проверьте параметры узла сети (адрес узла, порт, имя прокси/без прокси) и специфичные для элемента данные (например, SNMPv2 community или учетные данные безопасности SNMPv3). Эти поля учитывают контекст:
  - Значения заполняются заранее, когда это возможно, то есть для элементов, требующих агента, путем получения информации из выбранного интерфейса агента узла сети
  - Значения необходимо заполнять вручную для элементов шаблона
  - Значения макросов в виде обычного текста разрешаются
  - Если значение поля (или часть значения) является секретом или макросом Vault, поле будет пустым и его нужно заполнить вручную. Если какой-либо параметр элемента содержит значение секретного макроса, отображается следующее предупреждение: "Item contains user-defined macros with secret values. Values of these macros should be entered manually."
  - Поля отключаются, если они не нужны в контексте типа элемента (например, поля адреса узла и прокси отключены для вычисляемых элементов)
- Нажмите *Get value and test*, чтобы проверить предварительную обработку




Если в форме настройки элемента вы указали сопоставление значений (‘Show value’), диалог проверки элемента покажет еще одну строку после итогового результата с названием ‘Result with value map applied’.

Параметры, специфичные для получения реального значения с узла сети:

| Parameter | Description |
|---|---|
| *Get value from host* | Установите этот флажок, чтобы получить реальное значение с узла сети. |
| *Host address* | Введите адрес узла сети.<br>Это поле автоматически заполняется адресом интерфейса узла сети элемента. |
| *Port* | Введите порт узла сети.<br>Это поле автоматически заполняется портом интерфейса узла сети элемента. |
| *Additional fields for SNMP interfaces<br>(SNMP version, SNMP community, Context name, etc.)* | См. Настройка мониторинга SNMP для получения дополнительных сведений о настройке интерфейса SNMP (v1, v2 и v3).<br>Эти поля автоматически заполняются из интерфейса узла сети элемента. |
| *Proxy* | Укажите прокси, если узел сети отслеживается через прокси.<br>Это поле автоматически заполняется прокси узла сети (если он задан). |
| *Value* | Значение, полученное с узла сети.<br>Щелчок в поле параметра или по кнопке просмотра/редактирования откроет окно текстовой области со значением или блоком кода.<br>Значения обрезаются до максимального размера 512 КБ и только во веб-интерфейсе. Если результат обрезан, отображается значок предупреждения. Описание предупреждения показывается при наведении курсора. Обратите внимание, что данные объемом более 512 КБ по-прежнему полностью обрабатываются сервером Zabbix. |

Для остальных параметров см. Проверка гипотетического значения выше.

**2. Дополнительные сведения о предобработке**

Обзор

В этом разделе приведены сведения о предварительной обработке значений элементов данных. Предварительная обработка значений элементов данных позволяет определять и выполнять правила преобразования для полученных значений элементов данных.

Предварительная обработка управляется процессом менеджера предварительной обработки вместе с рабочими процессами предварительной обработки, которые выполняют этапы предварительной обработки. Все значения с предварительной обработкой (до Zabbix 7.4.1 — все значения), полученные от разных сборщиков данных, проходят

через менеджер предварительной обработки перед добавлением в кэш истории. Для связи между сборщиками данных (опросчиками, ловушками и т. д.) и процессом предварительной обработки используется IPC-связь на основе сокетов. Этапы предварительной обработки выполняет либо сервер Zabbix, либо прокси Zabbix (для элементов данных, контролируемых прокси).

Обработка значений элементов данных

Для визуализации потока данных от источника данных к базе данных Zabbix мы можем использовать следующую упрощённую диаграмму:

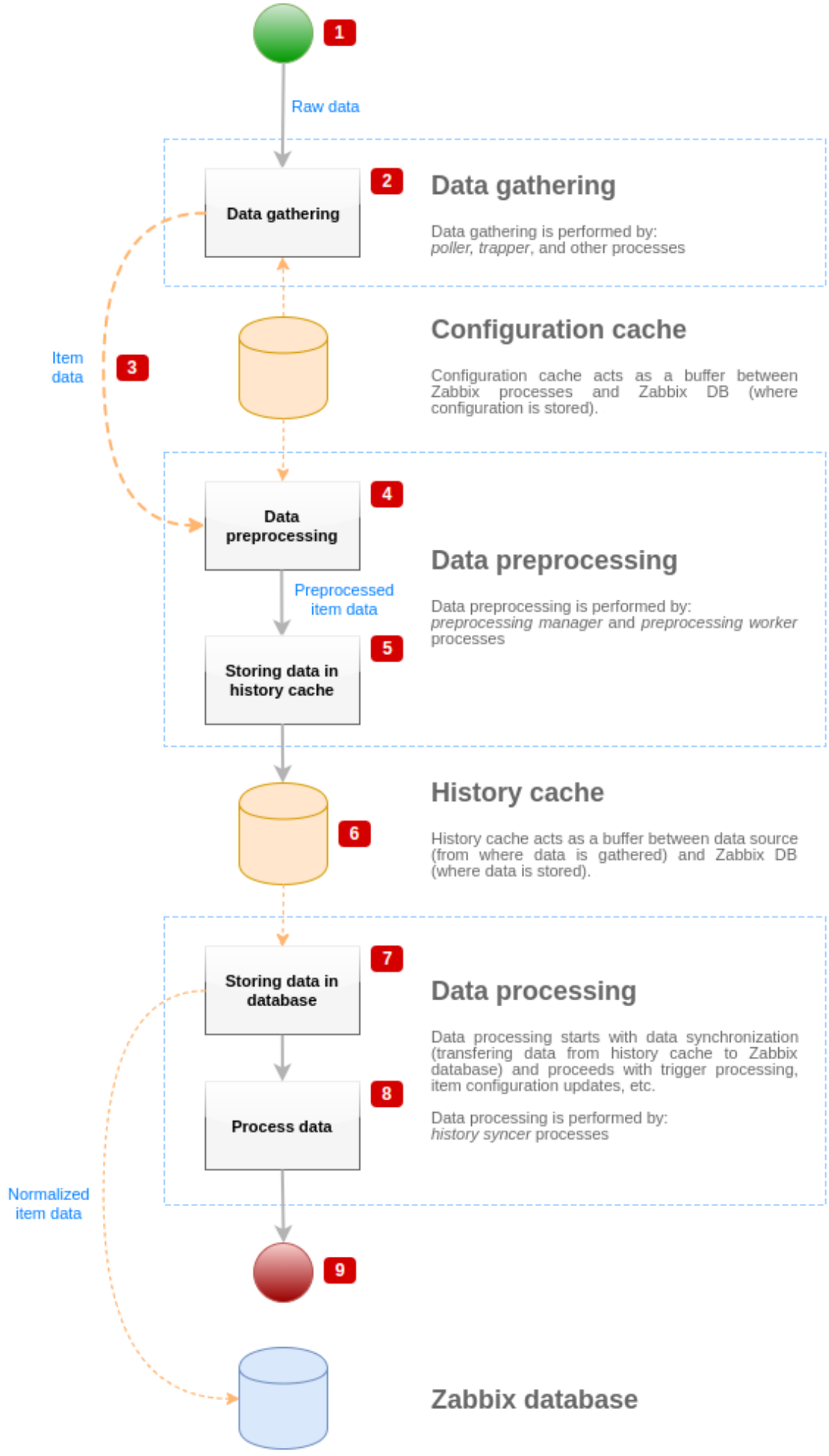

1
Raw data
Data gathering
2
Data gathering
Data gathering is performed by:
poller, trapper, and other processes
Item data
3
Configuration cache
Configuration cache acts as a buffer between Zabbix processes and Zabbix DB (where configuration is stored).
Data preprocessing
4
Data preprocessing
Data preprocessing 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

Диаграмма выше в **упрощённой** форме показывает только процессы, объекты и действия, которые связаны с предварительной обработкой значений элементов данных. Диаграмма не отображает изменения направлений при различных условиях, обработку ошибок или циклы. Локальный кэш данных в менеджере предобработки тоже не показан, так как он не влияет напрямую на поток данных. Цель этой диаграммы — показать процессы, вовлечённые в обработку значений элементов данных, а также способ их взаимодействия.

- Сбор данных начинается с сырых данных от источника данных. В этот момент данные содержат только ID, штамп времени и значение (также может быть несколько значений).
- Не важно, какой используется тип сборщика данных, идея одинакова для активных, пассивных проверок, для траппер элементов данных и т.д., поскольку меняется только формат данных и инициатор передачи (либо сборщик данных ожидает соединение и данные, либо сборщик данных инициирует соединение и запрашивает данные). Сырые данные проверяются, конфигурация элемента данных извлекается из кэша конфигурации (в данные добавляются конфигурационные данные).
- Механизм IPC на основе сокета используется для передачи данных от сборщиков данных к менеджеру предобработки. В этот момент сборщик данных продолжает сбор данных без ожидания ответа от менеджера предварительной обработки.
- Выполняется предварительная обработка данных. Это включает в себя выполнение шагов предварительной обработки и обработку зависимых элементов данных.

**Note:**
В процессе выполнения предварительной обработки элемент данных может изменить свое состояние на НЕ ПОДДЕРЖИВАЕТСЯ, если какой-либо шаг предварительной обработки завершается с ошибкой.

- Данные истории из локального кэша данных менеджера предобработки сбрасываются в кэш истории.
- Тут поток данных останавливается до следующей синхронизации кэша истории (когда процесс history syncer выполняет синхронизацию данных).
- Процесс синхронизации начинается с нормализации данных перед сохранением в базу данных Zabbix. Нормализация данных выполняет преобразования в желаемый тип элемента данных (тип, заданный в конфигурации элемента данных), включая обрезку текстовых данных на основе предопределённых размеров для этих типов (HISTORY_STR_VALUE_LEN для строк, HISTORY_TEXT_VALUE_LEN для текста и HISTORY_LOG_VALUE_LEN для журнал (лог) значений). Данные будут отправлены в базу данных Zabbix после завершения нормализации.

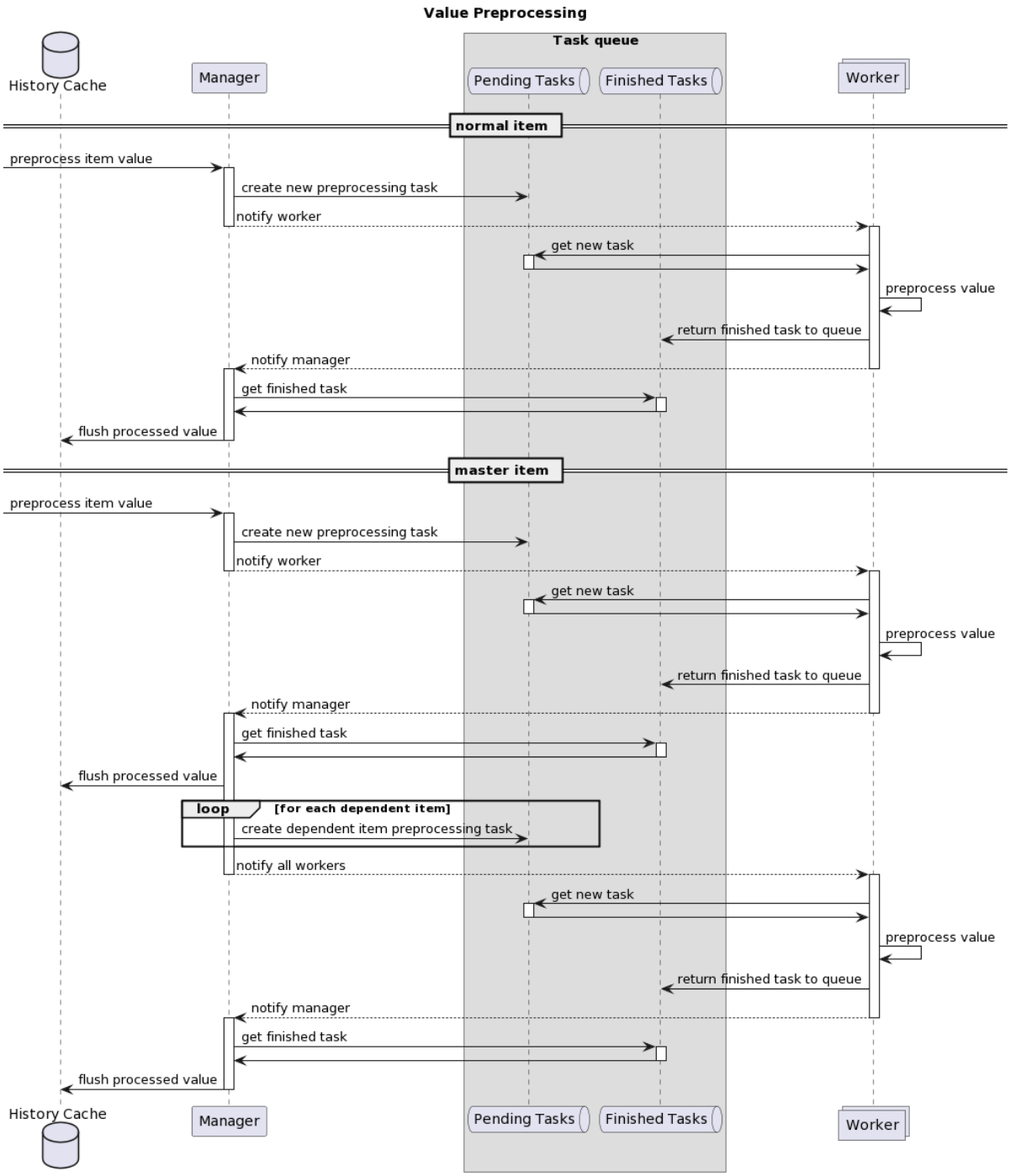
**Note:**
Элемент данных может изменить своё состояние на НЕ ПОДДЕРЖИВАЕТСЯ, если процесс нормализации завершится с ошибкой (например, когда текстовое значение не удалось преобразовать в число).

- Выполняется обработка собранных данных — проверяются триггеры, обновляется конфигурация элементов данных, если элемент данных становится НЕПОДДЕРЖИВАЕМЫМ, и тому подобное.
- Этот момент считается завершением потока данных с точки зрения обработки значений элементов данных.

Предобработка значения элемента данных

Предобработка данных выполняется в следующих шагах:

- Если у элемента данных нет ни предобработки, ни зависимых элементов, его значение либо добавляется в кэш истории, либо отправляется менеджеру LLD. В противном случае значение элемента данных передается менеджеру предобработки с использованием механизма IPC на основе UNIX socket (до Zabbix 7.4.1 все значения проходили через менеджер предобработки до добавления в кэш истории или отправки менеджеру LLD).
- Создается задача предобработки, она добавляется в очередь, и рабочие процессы предобработки уведомляются о новой задаче.
- На этом этапе поток данных останавливается, пока не появится хотя бы один свободный (то есть не выполняющий никаких задач) рабочий процесс предобработки.
- Когда рабочий процесс предобработки становится доступен, он берет следующую задачу из очереди.
- После завершения предобработки (как при неудачном, так и при успешном выполнении шагов предобработки) предварительно обработанное значение добавляется в очередь завершенных задач, и менеджер уведомляется о новой завершенной задаче.
- Менеджер предобработки преобразует результат в нужный формат (определяемый типом значения элемента данных) и либо добавляет его в кэш истории, либо отправляет менеджеру LLD.
- Если для обрабатываемого элемента данных есть зависимые элементы, то они добавляются в очередь предобработки с предварительно обработанным значением основного элемента данных. Зависимые элементы помещаются в очередь в обход обычных запросов на предобработку значений, но только для основных элементов данных, у которых значение задано и которые не находятся в состоянии NOT SUPPORTED.




Обратите внимание, что на диаграмме предобработка основного элемента данных немного упрощена за счет пропуска кэширования предобработки.

Очередь предварительной обработки

Очередь предварительной обработки организована следующим образом:

- список ожидающих задач:
  - задачи, созданные непосредственно из запросов предварительной обработки значений в порядке их получения
- список немедленных задач (обрабатываемых до ожидающих задач):
  - задачи тестирования (созданы в ответ на запросы тестирования элементов данных/предварительной обработки веб-интерфейсом)
  - задачи зависимых элементов данных
  - последовательные задачи (задачи, которые должны выполняться в строгом порядке):
  - имеющие шаги предварительной обработки с использованием последнего значения:
    * изменение
    * троттлинг
    * JavaScript (кэширование байт-кода)

  - кэширование предварительной обработки зависимых элементов данных
- список завершённых задач

Кэширование предварительной обработки

Кэширование предварительной обработки было введено для повышения производительности предварительной обработки для нескольких зависимых элементов данных, имеющих схожие шаги предварительной обработки (что является распространённым результатом LLD).

Кэширование выполняется путём предварительной обработки одного зависимого элемента данных и повторного использования некоторых внутренних данных предварительной обработки для остальных зависимых элементов данных. Кэширование предварительной обработки поддерживается только для первого шага предварительной обработки следующих типов:

- шаблон Prometheus (индексирует входные данные по метрикам)
- JSONPath (разбирает данные в дерево объектов и индексирует первое выражение `[?(@.path == "      ")]`)

Рабочие процессы предварительной обработки

Файл конфигурации Zabbix сервера позволяет пользователям указать количество потоков рабочих процессов предварительной обработки («preprocessing worker»). Параметр конфигурации StartPreprocessors следует использовать, чтобы задать количество экземпляров рабочих процессов предварительной обработки, которое должно как минимум соответствовать количеству доступных ядер CPU.

Если задачи предварительной обработки не привязаны к CPU и включают частые сетевые запросы, рекомендуется настроить дополнительные рабочие процессы. Оптимальное количество рабочих процессов предварительной обработки может определяться многими факторами, включая количество «предобрабатываемых» элементов данных (элементов данных, требующих выполнения каких-либо шагов предварительной обработки), количество процессов сбора данных, среднее количество шагов для предварительной обработки элементов данных и так далее. Недостаточное количество рабочих процессов может привести к высокому потреблению памяти. Для устранения неполадок, связанных с чрезмерным использованием памяти в вашей инсталляции Zabbix, смотрите Профилирование чрезмерного использования памяти с помощью tcmalloc.

Но если предположить, что тяжелые операции предварительной обработки (вроде разбора больших кусков XML/JSON) отсутствуют, количество процессов «preprocessing worker» может совпадать с количеством сборщиков данных. Таким образом, в большинстве случаев для полученных данных будет по крайней мере один незанятый рабочий процесс предварительной обработки (за исключением случаев, когда данные от сборщиков поступают массово).

**Warning:**
Слишком большое количество процессов сбора данных (поллеров, поллеров недоступных устройств, ODBC и HTTP поллеров, Java поллеров, пингеров, трапперов, прокси поллеров) вместе с менеджером IPMI, SNMP траппером и рабочими процессами предобработки могут исчерпать ограничение количества файловых дескрипторов на процесс для менеджера предварительной обработки. <br><br> Исчерпание лимита дескрипторов файлов на процесс приведёт к останову Zabbix сервера (обычно вскоре после запуска, но иногда это может занять более длительное время). Чтобы избежать подобной ситуации, проанализируйте файл конфигурации Zabbix сервера, чтобы оптимизировать количество одновременных проверок и процессов. Кроме того, при необходимости путём проверки и настройки системных ограничений убедитесь, что лимит дескрипторов файлов установлен достаточно высоким.

Поток обработки значения

Предварительная обработка значения элемента данных выполняется в несколько шагов (или фаз) несколькими процессами. Это может привести к тому, что:

- Зависимый элемент данных может получить значения, в то время как основной элемент данных — нет. Такое может произойти при следующем сценарии:
  - У основного элемента данных тип значения `UINT` (можно использовать элемент данных траппер), у зависимого элемента данных тип значения `TEXT`.
  - Шаги предварительной обработки не требуются ни для основного, ни для зависимого элементов данных.
  - Основному элементу данных необходимо передать текстовое значение (например, «abc»).
  - Поскольку нет шагов предварительной обработки для выполнения, менеджер предварительной обработки проверяет, что основной элемент данных не находится в НЕПОДДЕРЖИВАЕМОМ состоянии и что имеется значение (оба условия выполняются), и помещает в очередь зависимый элемент данных с таким же значением, что и основной элемент данных (поскольку шаги предварительной обработки отсутствуют).
  - Когда основной и зависимый элементы данных переходят в фазу синхронизации истории, основной элемент данных становится НЕПОДДЕРЖИВАЕМЫМ из-за ошибки конвертации значения (текстовые данные невозможно преобразовать в целое положительное число).

В результате зависимый элемент данных получает значение, тогда как основной элемент данных меняет свое состояние на НЕ ПОДДЕРЖИВАЕТСЯ.

- Зависимый элемент данных получает значение, которое отсутствует в истории основного элемента данных. Данный случай очень похож на предыдущий, за исключением типа основного элемента данных. Например, если у основного элемента данных используется тип CHAR, то значение основного элемента данных будет усечено на стадии синхронизации истории, в то время как зависимые элементы данных получат свои значения с изначального (не усечённого) значения основного элемента данных.

### 3. Примеры предобработки

Обзор

В этом разделе представлены примеры использования шагов предварительной обработки для решения некоторых практических задач.

Фильтрация записей журнала событий VMware

В этом примере используется шаг предварительной обработки Совпадение регулярному выражению (Matches regular expression) для фильтрации ненужных событий из журнала событий VMware.

1. Убедитесь, что на работающем узле сети гипервизора VMware элемент данных vmware.eventlog присутствует и работает должным образом. Обратите внимание, что элемент данных журнала событий уже может присутствовать у гипервизора, если при создании узла сети был присоединён шаблон VMware.

2. На узле сети гипервизора VMware создайте зависимый элемент данных с типом *Журнал (лог) (Log)* и выберите элемент данных журнала событий в качестве основного элемента данных.

3. На вкладке *Предобработка (Preprocessing)* зависимого элемента данных нажмите *Добавить (Add)*, чтобы создать шаг предобработки, и выберите из выпадающего списка *Совпадение регулярному выражению (Matches regular expression)*. Затем укажите один из следующих параметров:

- Для фильтрации всех событий журнала:

```
.* logged in .*
```

- Для фильтрации строк, содержащих имена пользователей после «User»:

```
\bUser\s+\K\S+
```

**Attention:**
Если нет соответствия регулярному выражению, то зависимый элемент данных станет неподдерживаемым с соответствующим сообщением об ошибке. Чтобы избежать этого, отметьте флажок *Другое при ошибке (Custom on fail)* и выберите опцию — например, отбросить значение либо выставить пользовательское значение. Обратите внимание, что отброшенные значения в базе данных не сохраняются; в результате триггеры не вычисляются и данные о динамике изменений не генерируются.

В качестве альтернативы вы можете использовать шаг предварительной обработки Регулярное выражение для извлечения соответствующих групп и управления выводом:

- Чтобы извлечь и вывести из журнала событий полное событие, содержащее строку «logged in», укажите следующие параметры:

```
(Pattern): .*logged in.*
(Output): \0
```

- Чтобы извлечь и вывести имена пользователей после «User»:

```
(Pattern): User (.*?)(?=\ )
(Output): \1
```

Проверка типа полученного значения

В этом примере используется шаг предварительной обработки Custom multiplier, чтобы проверить, является ли тип полученного значения элемента данных числовым.

На вкладке *Предварительная обработка* элемента данных выберите шаг предварительной обработки *Custom multiplier* и укажите следующий параметр (умножает полученное значение на 1):

```
1
```

**Attention:**
Если предварительная обработка завершается сбоем (например, входные данные не являются числовыми), элемент данных становится неподдерживаемым с соответствующим сообщением об ошибке. Чтобы избежать этого, установите флажок *Custom on fail* и выберите вариант, например, отклонение значения или задание пользовательского значения. Обратите внимание, что отклоненные значения не сохраняются в базе данных; в результате триггеры не вычисляются и данные трендов не создаются.

Проверка на неподдерживаемое значение

В этом примере используется шаг предварительной обработки Проверка на неподдерживаемое значение (Check for not supported value) для проверки того, что не удалось извлечь значение элемента данных.

Когда процесс поллера Zabbix сервера/прокси пытается собрать значение элемента данных, он может:

- Вернуть корректный результат.
- Вернуть результат, который изначально кажется корректным, но может стать неподдерживаемым позже (например, из-за несоответствия типов значения после предобработки).
- Вернуть ошибку получения значения, в результате чего элемент станет неподдерживаемым. Распространённые причины включают:
  - Неизвестный ключ элемента данных (для Zabbix агента, Простой проверки или внутренних элементов данных Zabbix)
  - Неизвестный OID (SNMP агент), неизвестный датчик (IPMI агент) или отсутствие метрики JMX (JMX агент)
  - Невозможность прочитать файл трапов (SNMP trap)
  - Скрипт не найден (Внешняя проверка)
  - Нет такого URL (HTTP агент, Браузер)
  - Неуспешный вход (SSH агент, TELNET агент)
  - Недопустимый синтаксис формулы (Вычисляемый), ошибка синтаксиса JavaScript (Скрипт) или некорректный SQL (Монитор баз данных)

Чтобы обнаружить и обработать ошибки сбора значений элементов данных, вы можете использовать шаг предварительной обработки *Проверка на неподдерживаемое значение (Check for not supported value)*. Обратите внимание, что этот шаг всегда выполняется первым и обнаруживает только ошибки, которые возникают до начала предварительной обработки.

На вкладке *Предобработка (Preprocessing)* элемента данных выберите шаг предобработки *Проверка на неподдерживаемое значение (Check for not supported value)* и укажите один из следующих параметров:

- Для любых ошибок:

```
:          (any error)
```

- Для ошибок, содержащих «cannot connect»:

```
:              (error matches)
: (?i)cannot connect
```

Затем используйте параметр *Другое при ошибке (Custom on fail)*, чтобы отбросить значение (в данном случае ошибку), задать пользовательское значение или вернуть пользовательское сообщение об ошибке. Обратите внимание, что отброшенные значения в базе данных не сохраняются; в результате триггеры не вычисляются и данные о динамике изменений не генерируются.

#### 4. Поддержка функционала JSONPath

Обзор

В этом разделе описывается поддерживаемый функционал JSONPath в шагах предварительной обработки значений элементов данных.

JSONPath состоит из сегментов, разделенных точками. Сегмент может быть иметь форму либо простого слова, представляющего имя значения JSON, либо подстановочного символа (*), либо более замысловатой конструкции, заключённой в квадратные скобки. Точка перед сегментом скобок необязательна и может быть опущена.

| Пример JSONPath | Описание |
|---|---|
| `$.object.name` | Возвращает содержимое object.name. |
| `$.object['name']` | Возвращает содержимое object.name. |
| `$.object.['name']` | Возвращает содержимое object.name. |
| `$["object"]['name']` | Возвращает содержимое object.name. |
| `$.['object'].["name"]` | Возвращает содержимое object.name. |

| Пример JSONPath | Описание |
|---|---|
| `$.object.history.length()` | Возвращает количество элементов массива object.history. |
| `$[?(@.name == 'Object')].price.first()` | Возвращает значение свойства `price` первого объекта по имени «Object». |
| `$[?(@.name == 'Object')].history.first().length()` | Возвращает количество элементов массива history первого объекта с именем «Object». |
| `$[?(@.price > 10)].length()` | Возвращает количество объектов с price больше чем 10. |

Смотрите также: Экранирование специальных символов из значений LLD макросов в JSONPath.

Поддерживаемые сегменты

| Сегмент | Описание |
|---|---|
| `< >` | Соответствие свойству объекта по имени. |
| `*` | Соответствие всем свойствам объекта. |
| `['< >']` | Соответствие свойству объекта по имени. |
| `['< >', '< >', ...]` | Соответствие свойству объекта по любому из перечисленных имён. |
| `[< >]` | Соответствие элементу массива по его индексу. |
| `[< >, < >, ...]` | Соответствие элементу массива по любому из перечисленных индексов. |
| `[*]` | Соответствие всем свойствам объекта или элементам массива. |
| `[< >:< >]` | Соответствие элементам массива по заданному диапазону:<br>**<начало>** — первый индекс соответствия (включительно). Если не указан, соответствует всем элементам с самого начала. В случае отрицательного значения указывает начальное смещение от конца массива.<br>**<конец>** — последний индекс соответствия (не включая). Если не указан, соответствует всем элементам массива до самого конца. В случае отрицательного значения указывает начальное смещение от конца массива. |
| `[?(< >)]` | Соответствие объектов / элементов массива с применением выражения фильтра. |

Чтобы найти соответствующий сегмент, игнорируя его происхождение (отсоединённый сегмент), его необходимо указывать с префиксом '..', например `$..name` или `$..['name']` вернёт значения всех свойств «name».

Соответствующие имена элементов можно извлечь, добавив суффикс ~ к JSONPath. Он возвращает имя соответствующего объекта или индекс в строковом формате соответствующего элемента массива. Формат вывода следует тем же самым правилам, что и остальные запросы JSONPath — результаты определённого пути возвращаются «как есть», а результаты неопределённого пути возвращаются в виде массива. Однако, нет особого смысла извлекать имя соответствующего элемента определённого пути — оно и так уже известно.

Выражение фильтра

Выражение фильтра является арифметическим выражением в инфиксной нотации.

Поддерживаемые операнды:

| Операнд | Описание |
|---|---|
| `"< >"`<br>`'<text>'` | Текстовая константа.<br><br>Пример:<br>`'value: \\'1\\''`<br>`"value: '1'"` |
| `< >` | Числовая константа, поддерживая научное представление.<br><br>Пример: 123 |
| `<jsonpath $>` | Значение, указанное в JSONPath, начиная с корневого узла входного документа; поддерживаются только определённые пути.<br><br>Пример: `$.object.name` |

| Операнд | Описание |
|---|---|
| <jsonpath @> | Значение, указанное в JSONPath, начиная с текущего объекта/элемента; поддерживаются только определённые пути.<br><br>Пример: `@.name` |

Поддерживаемые операторы:

| Оператор | Тип | Описание | Результат |
|---|---|---|---|
| - | бинарный | Вычитание | Число |
| + | бинарный | Сложение | Число |
| / | бинарный | Деление | Число |
| * | бинарный | Умножение | Число |
| == | бинарный | Равно | Логический (1 или 0) |
| != | бинарный | Не равно | Логический (1 или 0) |
| | бинарный | Меньше чем | Логический (1 или 0) |
| <= | бинарный | Меньше чем или равно | Логический (1 или 0) |
| > | бинарный | Больше чем | Логический (1 или 0) |
| >= | бинарный | Больше чем или равно | Логический (1 или 0) |
| =~ | бинарный | Соответствие регулярному выражению | Логический (1 или 0) |
| ! | унарный | Логическое НЕ | Логический (1 или 0) |
| \|\| | бинарный | Логическое ИЛИ | Логический (1 или 0) |
| && | бинарный | Логическое И | Логический (1 или 0) |

Функции

Функции можно использовать в конце JSONPath. Несколько функций можно включать в цепочку, если предыдущая функция возвращает значение, которое принимается следующей функцией.

Поддерживаемые функции:

| Функция | Описание | Вход | Выход |
|---|---|---|---|
| avg | Среднее значение из чисел во входящем массиве | Массив чисел | Число |
| min | Минимальное значение из чисел во входящем массиве | Массив чисел | Число |
| max | Максимальное значение из чисел во входящем массиве | Массив чисел | Число |
| sum | Сумма чисел во входящем массиве | Массив чисел | Число |
| length | Количество элементов во входящем массиве | Массив | Число |
| first | Первый элемент массива | Массив | Конструкция JSON (объект, массив, значение), в зависимости от содержимого входящего массива |

Функциями агрегации JSONPath принимаются числовые значения в кавычках. Это означает, что значения преобразуются из строкового типа в числовой, если требуется агрегирование.

Несовместимые входные данные повлекут ошибку в функции.

Результирующее значение

JSONPath можно разделить на определённые и неопределённые пути. Определённый путь может вернуть только null или одно совпадение. Неопределённый путь может вернуть несколько совпадений: JSONPath с отсоединёнными несколькими именами / списком индексов, фрагментами массива или сегментами выражения. Однако, когда используется функция, JSONPath становится определённым, так как функции всегда имеют одиночное результирующее значение.

Определённый путь возвращает объект/массив/значение, на которое он ссылается. Напротив, неопределённый путь возвращает массив соответствующих объектов/массивов/значений.

**Attention:**
Порядок свойств в результатах запроса JSONPath может не соответствовать порядку свойств исходного JSON'а из-за внутренних методов оптимизации. Например, JSONPath `$.books[1]["author", "title"]` может вернуть `["title", "author"]`. Если сохранение исходного порядка свойств имеет важное значение, следует рассмотреть альтернативные методы обработки пост-запроса.

Правила форматирования пути

Пробельные символы (пробелы и символы табуляции) можно свободно использовать в сегментах с представлением в виде квадратных скобок и в выражениях, например: `$[ 'a' ][ 0 ][ ?( $.b == 'c' ) ][ : -1 ].first( )`.

Строки должны быть заключены в одинарные (') или двойные (") кавычки. Внутри строк одинарные или двойные кавычки (в зависимости от того, в какие кавычки заключена строка) и обратные косые черты (\) экранируются при помощи символа обратной косой черты (\).

Пример


```
{
  "books": [
    {
      "category": "reference",
      "author": "Nigel Rees",
      "title": "Sayings of the Century",
      "price": 8.95,
      "id": 1
    },
    {
      "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Sword of Honour",
      "price": 12.99,
      "id": 2
    },
    {
      "category": "fiction",
      "author": "Herman Melville",
      "title": "Moby Dick",
      "isbn": "0-553-21311-3",
      "price": 8.99,
      "id": 3
    },
    {
      "category": "fiction",
      "author": "J. R. R. Tolkien",
      "title": "The Lord of the Rings",
      "isbn": "0-395-19395-8",
      "price": 22.99,
      "id": 4
    }
  ],
  "services": {
    "delivery": {
      "servicegroup": 1000,
      "description": "Next day delivery in local town",
      "active": true,
      "price": 5
    },
    "bookbinding": {
      "servicegroup": 1001,
      "description": "Printing and assembling book in A5 format",
      "active": true,
      "price": 154.99
    },
    "restoration": {
```

```
      "servicegroup": 1002,
      "description": "Various restoration methods",
      "active": false,
      "methods": [
        {
          "description": "Chemical cleaning",
          "price": 46
        },
        {
          "description": "Pressing pages damaged by moisture",
          "price": 24.5
        },
        {
          "description": "Rebinding torn book",
          "price": 99.49
        }
      ]
    }
  },
  "filters": {
    "price": 10,
    "category": "fiction",
    "no filters": "no \"filters\""
  },
  "closed message": "Store is closed",
  "tags": [
    "a",
    "b",
    "c",
    "d",
    "e"
  ]
}
```

| JSONPath | Тип | Результат |
|---|---|---|
| `$.filters.price` | определённый | 10 |
| `$.filters.category` | определённый | fiction |
| `$.filters['no filters']` | определённый | no "filters" |
| `$.filters` | определённый | {<br>"price": 10,<br>"category": "fiction",<br>"no filters": "no \"filters\""<br>} |
| `$.books[1].title` | определённый | Sword of Honour |
| `$.books[-1].author` | определённый | J. R. R. Tolkien |
| `$.books.length()` | определённый | 4 |
| `$.tags[:]` | неопределённый | ["a", "b", "c", "d", "e" ] |
| `$.tags[2:]` | неопределённый | ["c", "d", "e" ] |
| `$.tags[:3]` | неопределённый | ["a", "b", "c"] |
| `$.tags[1:4]` | неопределённый | ["b", "c", "d"] |
| `$.tags[-2:]` | неопределённый | ["d", "e"] |
| `$.tags[:-3]` | неопределённый | ["a", "b"] |
| `$.tags[:-3].length()` | определённый | 2 |
| `$.books[0, 2].title` | неопределённый | ["Moby Dick", "Sayings of the Century"] |
| `$.books[1]['author', "title"]` | неопределённый | ["Sword of Honour", "Evelyn Waugh"] |
| `$..id` | неопределённый | [1, 2, 3, 4] |
| `$.services..price` | неопределённый | [14.99, 5, 46, 24.5, 99.49] |

| JSONPath | Тип | Результат |
|---|---|---|
| `$.books[?(@.id == 4 - 0.4 * 5)].title` | неопределённый | [”Sword of Honour”]<br><br>Примечание: Этот запрос показывает, что в запросах можно использовать математические операции. Конечно, этот запрос можно упростить до `$.books[?(@.id == 2)].title` |
| `$.books[?(@.id == 2 \|\| @.id == 4)].title` | неопределённый | [”Sword of Honour”, ”The Lord of the Rings”] |
| `$.books[?(!(@.id == 2))].title` | неопределённый | [”Sayings of the Century”, ”Moby Dick”, ”The Lord of the Rings”] |
| `$.books[?(@.id != 2)].title` | неопределённый | [”Sayings of the Century”, ”Moby Dick”, ”The Lord of the Rings”] |
| `$.books[?(@.title =~ " of ")].title` | неопределённый | [”Sayings of the Century”, ”Sword of Honour”, ”The Lord of the Rings”] |
| `$.books[?(@.price > 12.99)].title` | неопределённый | [”The Lord of the Rings”] |
| `$.books[?(@.author > "Herman Melville")].title` | неопределённый | [”Sayings of the Century”, ”The Lord of the Rings”] |
| `$.books[?(@.price > $.filters.price)].title` | неопределённый | [”Sword of Honour”, ”The Lord of the Rings”] |
| `$.books[?(@.category == $.filters.category)].title` | неопределённый | [”Sword of Honour”,”Moby Dick”,”The Lord of the Rings”] |
| `$.books[?(@.category == "fiction" && @.price < 10)].title` | неопределённый | [”Moby Dick”] |
| `$..[?(@.id)]` | неопределённый | {<br>”price”: 8.95,<br>”id”: 1,<br>”category”: ”reference”,<br>”author”: ”Nigel Rees”,<br>”title”: ”Sayings of the Century”<br>},<br>{<br>”price”: 12.99,<br>”id”: 2,<br>”category”: ”fiction”,<br>”author”: ”Evelyn Waugh”,<br>”title”: ”Sword of Honour”<br>},<br>{<br>”price”: 8.99,<br>”id”: 3,<br>”category”: ”fiction”,<br>”author”: ”Herman Melville”,<br>”title”: ”Moby Dick”,<br>”isbn”: ”0-553-21311-3”<br>},<br>{<br>”price”: 22.99,<br>”id”: 4,<br>”category”: ”fiction”,<br>”author”: ”J. R. R. Tolkien”,<br>”title”: ”The Lord of the Rings”,<br>”isbn”: ”0-395-19395-8”<br>}<br>] |
| `$.services..[?(@.price > 50)].description` | неопределённый | [”Printing and assembling book in A5 format”, ”Rebinding torn book”] |
| `$..id.length()` | определённый | 4 |
| `$.books[?(@.id == 2)].title.first()` | определённый | Sword of Honour |

| JSONPath | Тип | Результат |
|---|---|---|
| `$..tags.first().length()` | определённый | 5<br><br>Примечание: `$..tags` неопределённый путь, поэтому он вернёт массив соответствующих элементов, т.е. `[["a", "b", "c", "d", "e" ]]`; `first()` вернёт первый элемент, т.е., `["a", "b", "c", "d", "e"]`; `length()` вычислит его размер, т.е.5. |
| `$.books[*].price.min()` | определённый | 8.95 |
| `$..price.max()` | определённый | 154.99 |
| `$.books[?(@.category == "fiction")].price.avg()` | определённый | 14.99 |
| `$.books[?(@.category == $.filters.xyz)].title` | неопределённый | Примечание: Запрос без совпадения вернёт NULL по определённому и неопределённому путям. |
| `$.services[?(@.active=="true")].servicegroup` | определённый | [1001,1000]<br><br>Примечание: В сравнениях логических значений необходимо использовать текстовые константы. |
| `$.services[?(@.active=="false")].servicegroup` | определённый | [1002]<br><br>Примечание: В сравнениях логических значений необходимо использовать текстовые константы. |
| `$.services[?(@.servicegroup=="1002")].firstname` | определённый | Simon |

1. Экранирование спецсимволов из значений LLD макросов в JSONPath

Когда в предварительной обработке JSONPath используются макросы низкоуровневого обнаружения и их значения раскрываются, применяются следующие правила экранирования спецсимволов:

- для экранирования учитываются только символы обратной косой черты (\) и двойной кавычки (”);
- если раскрытое значение макроса содержит эти символы, каждый из них экранируется при помощи обратной косой черты;
- если эти символы уже экранированы при помощи обратной косой черты, такая запись не считается экранированием и как косая черта, так и последующие спецсимволы будут экранированы еще раз.

Например:

| JSONPath | Значение LLD макроса | После подстановки |
|---|---|---|
| $.[?(@.value == ”{#MACRO}”)] | special ”value” | $.[?(@.value == ”special \”value\””)] |
| | c:\temp | $.[?(@.value == ”c:\\temp”)] |
| | a\\b | $.[?(@.value == ”a\\\\b”)] |

Макрос, который может иметь спецсимволы, при использовании в выражении необходимо заключить в двойные кавычки:

| JSONPath | Значение LLD макроса | После подстановки | Результат |
|---|---|---|---|
| $.[?(@.value == ”{#MACRO}”)] | special ”value” | $.[?(@.value == ”special \”value\””)] | OK |
| $.[?(@.value == {#MACRO})] | | $.[?(@.value == special \”value\”)] | **Плохое JSONPath выражение** |

Макрос который может иметь спецсимволы, при использовании в пути необходимо заключить в квадратные скобки **и** двойные кавычки:

| JSONPath | Значение LLD макроса | После подстановки | Результат |
|---|---|---|---|
| $.[”{#MACRO}”].value | c:\temp | $.[”c:\\temp”].value | OK |
| $.{#MACRO}.value | | $.c:\\temp.value | **Плохое JSONPath выражение** |

### 5. Предобработка Javascript

Обзор

В этом разделе приведены сведения о предварительной обработке при помощи JavaScript.

> **Attention:**
> Не используйте необъявленные присваивания в предварительной обработке JavaScript. Используйте `var` для объявления локальных переменных.

Предобработка Javascript

Предобработка JavaScript выполняется путём вызова функции JavaScript с единственным параметром «value» и заданным пользователем телом функции. Результатом шага предобработки является значение, которое возвращается из этой функции, например, для выполнения конвертации из градусов Фаренгейта в градусы Цельсия пользователю необходимо ввести:

```
return (value - 32)  * 5 / 9
```

Это будет обёрнуто сервером в функцию JavaScript :

```
function (value)
{
   return (value - 32) * 5 / 9
}
```

Входной параметр «value» всегда передаётся как строка. Возвращаемое значение автоматически приводится к строке через метод toString() (если он завершится с ошибкой, тогда эта ошибка возвращается в виде строкового значения), с несколькими исключениями:

- возврат неопределённого значения приведёт к ошибке
- возврат значения null приведет к тому, что входящее значение будет отброшено, очень похоже на предварительную обработку «Отбросить значение» на действие «Другое при ошибке».

Ошибки можно вернуть при помощи выбрасывания значений/объектов в качестве исключений (обычно это строки или объекты ошибок).

Например:

```
if (value == 0)
    throw "Zero input value"
return 1/value
```

У каждого скрипта есть 10-секундный тайм-аут — время ожидания выполнения (в зависимости от скрипта, срабатывание тайм-аута может занять более длительное время); превышение времени ожидания вернёт ошибку. Также применяется ограничение heap памяти в 512 мегабайт.

Байт-код шага предобработки JavaScript кэшируется и используется повторно, когда шаг применяется в следующий раз. Любые изменения в шагах предварительной обработки элемента данных вызовут сброс закэшированного скрипта и затем его перекомпиляцию.

Последовательные ошибки во время выполнения (3 раза подряд) вызовут повторную инициализацию механизма для уменьшения вероятности того, что один скрипт сломает всю среду выполнения для следующих скриптов (это действие заносится в журнал с режимом отладки DebugLevel 4 и выше).

Предобработка JavaScript реализована при помощи механизма JavaScript Duktape ([en] https://duktape.org/).

Смотрите также: Дополнительные объекты и глобальные функции JavaScript

Использование макросов в скриптах

В коде JavaScript можно использовать пользовательские макросы (а также макросы низкоуровневого обнаружения в контексте низкоуровневого обнаружения). Если скрипт содержит макросы, эти макросы раскрываются сервером/прокси до выполнения конкретных шагов предварительной обработки. Обратите внимание, что при тестировании шагов предварительной обработки в веб-интерфейсе значения макросов не извлекаются автоматически и их нужно указывать вручную.

**Note:**
Когда макрос заменяется своим значением, контекст игнорируется. Значение макроса вставляется в код как есть, невозможно добавить дополнительное экранирование до помещения значения в код JavaScript. Поэтому будьте аккуратны, так как в некоторых случаях это может вызвать ошибки в JavaScript.

В примере ниже, если полученное значение превышает значение макроса {$THRESHOLD}, то вместо самого значения возвращается значение порога (если присутствует):

```
var threshold = '{$THRESHOLD}';
return (!isNaN(threshold) && value > threshold) ? threshold : value;
```

Примеры

Следующие примеры иллюстрируют, как можно использовать предобработку JavaScript.

Каждый пример содержит краткое описание, тело функции для параметров предварительной обработки JavaScript и результат шага предварительной обработки — значение, возвращаемое функцией.

Пример 1: Преобразование числа (из научной нотации в целое число)

Преобразовать число «2.62128e+07» из научной нотации в целое число.

```
return (Number(value))
```

Значение, возвращаемое функцией: 26212800.

Пример 2: Преобразование числа (двоичного в десятичное)

Преобразовать двоичное число «11010010» в десятичное.

```
return(parseInt(value,2))
```

Значение, возвращаемое функцией: 210.

Пример 3: Округление числа

Округлить число «18.2345» до двух цифр.

```
return(Math.round(value* 100) / 100)
```

Значение, возвращаемое функцией: 18.23.

Пример 4: Подсчёт букв в строке

Подсчитать количество букв в строке «Zabbix».

```
return (value.length)
```

Значение, возвращаемое функцией: 6.

Пример 5: Получить оставшееся время

Получить время (в секундах), оставшееся до момента истечения срока действия сертификата (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);
```

Значение, возвращаемое функцией: 44380233.

Пример 6: Удаление свойств JSON

Изменить структуру данных JSON, удалив все свойства с ключом "data_size" или "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)
```

Значение, принимаемое функцией:

```
[
    {
        "table_name":"history",
        "data_size":"326.05",
        "index_size":"174.34"
    },
    {
        "table_name":"history_log",
        "data_size":"6.02",
        "index_size":"3.45"
    }
]
```

Значение, возвращаемое функцией:

```
[
    {
        "table_name":"history"
    },
    {
        "table_name":"history_log"
    }
]
```

Пример 7: Преобразование статуса Apache в JSON

Преобразовать значение, полученное от элемента данных агента Zabbix web.page.get (например, web.page.get[http://127.0.0.1:80/serv
status?auto]), в объект JSON.

```
//              Apache   JSON

//
var lines = value.split('\n');

//               «output»
var output = {};

//          «workers»
var workers = {
    '_': 0, 'S': 0, 'R': 0, 'W': 0,
    'K': 0, 'D': 0, 'C': 0, 'L': 0,
    'G': 0, 'I': 0, '.': 0
};

//                   «lines»       «output»          (      -     )
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]);
    }
}

//
output.ServerUptimeSeconds = output.ServerUptimeSeconds || output.Uptime;
output.ServerVersion = output.ServerVersion || output.Server;

//           «Scoreboard»                 worker-
if (typeof output.Scoreboard === 'string') {
    for (var i = 0; i < output.Scoreboard.length; i++) {
        var char = output.Scoreboard[i];
```

```
        workers[char]++;
    }
}

//            worker-        «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['.']
};

//          JSON
return JSON.stringify(output);
```

Значение, принятое функцией:

```
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__.....................................................................................
```

Значение, возвращаемое функцией:

```
{
    "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
    }
}
```


1. Дополнительные объекты Javascript

Обзор

Этот раздел описывает дополнения Zabbix к языку JavaScript, который реализован при помощи Duktape, и поддерживаемые глобальные функции JavaScript.

**Attention:**
Не используйте необъявленные присваивания в предварительной обработке JavaScript. Используйте `var` для объявления локальных переменных.

Встроенные объекты

Zabbix

Объект Zabbix даёт возможность взаимодействия с внутренним функционалом Zabbix.

| Метод | Описание |
|---|---|
| log( _ , ) | Записывает <сообщение> в журнал Zabbix с использованием уровня журнала <уровень_журнала> (смотрите параметр DebugLevel в файле конфигурации). |

Пример:

```
Zabbix.log(3, "this is a log entry written with 'Warning' log level")
```

Вы можете использовать следующие псевдонимы:

| Псевдоним | Псевдоним к |
|---|---|
| console.log(объект) | Zabbix.log(4, JSON.stringify(объект)) |
| console.warn(объект) | Zabbix.log(3, JSON.stringify(объект)) |
| console.error(объект) | Zabbix.log(2, JSON.stringify(объект)) |

**Attention:**
Суммарный размер всех записываемых в журнал сообщений ограничен величиной 8 мегабайт на выполняемый скрипт.

| Метод | Описание |
|---|---|
| sleep( ) | Задержка выполнения JavaScript на миллисекунд. |

Пример (задержка выполнения на 15 секунд):

```
Zabbix.sleep(15000)
```

HttpRequest

Этот объект инкапсулирует обработчик cURL, который позволяет выполнять простые HTTP запросы. Ошибки сообщаются как исключения.

**Attention:**
Инициализация нескольких объектов `HttpRequest` ограничена количеством 10 штук на выполняемый скрипт.

| Метод | Описание |
|---|---|
| addHeader( ) | Добавление поля HTTP заголовка. Это поле используется для всех последующих вызовов, до момента очистки при помощи метода `clearHeader()`.<br>Суммарная длина полей заголовков, которые могут быть добавлены к одному объекту `HttpRequest`, ограничена величиной 128 килобайт (включая специальные символы и имена заголовков). |
| clearHeader() | Очистка HTTP заголовка. Если заданные поля заголовков отсутствуют, `HttpRequest` задаст Content-Type значением application/json, если публикуемые данные отформатированы в виде JSON; в противном случае — text/plain. |
| connect(url) | Отправка на URL HTTP запроса CONNECT и получение ответа. |
| customRequest( , url, ) | Позволяет указать первым параметром любой HTTP метод. Отправляет на URL запрос указанным методом с опциональной полезной нагрузкой в параметре *данные* и возвращает ответ. |
| delete(url, ) | Отправка на URL HTTP запроса DELETE с опциональной полезной нагрузкой в параметре *данные* и получение ответа. |
| getHeaders(< _ _ | Возвращает объект полученных полей HTTP заголовков.<br>Параметр _ _ можно задать значением «true» (например, `getHeaders(true)`), «false» или неопределённым. Если задано значением «true», полученные значения полей HTTP заголовков вернутся в виде массивов; это следует использовать для получения значений полей нескольких заголовков с одинаковыми именами.<br>Если не задано или значение «false», полученные значения полей HTTP заголовков возвращаются в виде строк. |
| get(url, ) | Отправка на URL HTTP запроса GET с опциональной полезной нагрузкой в параметре *данные* и получение ответа. |

| Метод | Описание |
|---|---|
| head(url) | Отправка на URL HTTP запроса HEAD и получение ответа. |
| options(url) | Отправка на URL HTTP запроса OPTIONS и получение ответа. |
| patch(url, ) | Отправка на URL HTTP запроса PATCH с опциональной полезной нагрузкой в параметре *данные* и получение ответа. |
| put(url, ) | Отправка на URL HTTP запроса PUT с опциональной полезной нагрузкой в параметре *данные* и получение ответа. |
| post(url, ) | Отправка на URL HTTP запроса POST с опциональной полезной нагрузкой в параметре *данные* и получение ответа. |
| getStatus() | Получение кода состояния последнего HTTP запроса. |
| setProxy( ) | Задаёт HTTP прокси значением «прокси». Если этот параметр пустой, то никакой прокси не используется. |
| setHttpAuth( _ , _ , ) | Включение методов HTTP аутентификации (HTTPAUTH_BASIC, HTTPAUTH_DIGEST, HTTPAUTH_NEGOTIATE, HTTPAUTH_NTLM, HTTPAUTH_NONE) в параметре «битовая_маска». Флаг HTTPAUTH_NONE позволяет отключить HTTP аутентификацию.<br>Например:<br>`request.setHttpAuth(HTTPAUTH_NTLM \| HTTPAUTH_BASIC, username, password)`<br>`request.setHttpAuth(HTTPAUTH_NONE)` |
| trace(url, ) | Отправка на URL HTTP запроса TRACE с опциональной полезной нагрузкой в параметре *данные* и получение ответа. |

Пример:

```
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

Объект XML позволяет обрабатывать XML данные в элементах данных и в предварительной обработке низкоуровневых обнаружений, а также в вебхуках.

> **Attention:**
> Чтобы использовать объект XML, сервер/прокси должны быть скомпилированы с поддержкой libxml2.

| Метод | Описание |
|---|---|
| `XML.query( , )` | Получение содержимого узла с использованием XPath. Возвращает null, если узел не найден.<br>**выражение** — выражение XPath;<br>**данные** — данные XML в виде строки. |
| `XML.toJson( )` | Преобразование данных в XML формате в JSON. |
| `XML.fromJson( )` | Преобразование данных в JSON формате в XML. |

Пример:

*Входящие данные:*

```
<menu>
    <food type = "breakfast">
        <name>Chocolate</name>
        <price>$5.95</price>
        <description></description>
        <calories>650</calories>
    </food>
</menu>
```

*Вывод:*

```
{
    "menu": {
        "food": {
            "@type": "breakfast",
            "name": "Chocolate",
            "price": "$5.95",
            "description": null,
            "calories": "650"
        }
    }
}
```

Правила сериализации

Преобразование XML в JSON обрабатывается в соответствии со следующими правилами (для преобразования JSON в XML применяются обратные правила):

1. Атрибуты XML преобразуются в ключи, имеющие в своих именах префикс '@'.

Пример:

*Входящие данные:*

```
<xml foo="FOO">
  <bar>
    <baz>BAZ</baz>
  </bar>
</xml>
```

*Вывод:*

```
{
  "xml": {
    "@foo": "FOO",
    "bar": {
      "baz": "BAZ"
```

```
    }
  }
}
```

2. Элементы, которые закрываются в самих себе (<foo/>), будут преобразованы как имеющие значение «null».

Пример:

*Входящие данные:*

```
<xml>
  <foo/>
</xml>
```

*Вывод:*

```
{
  "xml": {
    "foo": null
  }
}
```

3. Пустые атрибуты (со значением ””) будут преобразованы как имеющие значение пустой строки (”).

Пример:

*Входящие данные:*

```
<xml>
  <foo bar="" />
</xml>
```

*Вывод:*

```
{
  "xml": {
    "foo": {
      "@bar": ""
    }
  }
}
```

4. Несколько дочерних узлов с идентичными именами элементов будут преобразованы в один ключ, который имеет массив значений в качестве своего значения.

Пример:

*Входящие данные:*

```
<xml>
  <foo>BAR</foo>
  <foo>BAZ</foo>
  <foo>QUX</foo>
</xml>
```

*Вывод:*

```
{
  "xml": {
    "foo": ["BAR", "BAZ", "QUX"]
  }
}
```

5. Если текстовый элемент не имеет атрибутов и дочерних элементов, он будет преобразован в строку.

Пример:

*Входящие данные:*

```
<xml>
    <foo>BAZ</foo>
</xml>
```

*Вывод:*

```
{
  "xml": {
    "foo": "BAZ"
   }
}
```

6. Если текстовый элемент не имеет дочерних элементов, но имеет атрибуты, то текстовое содержимое будет преобразовано в элемент с ключом «#text» и содержимым в виде значения; атрибуты будут преобразованы в соответствии с правилом сериализации 1.

Пример:

*Входящие данные:*

```
<xml>
  <foo bar="BAR">
    BAZ
  </foo>
</xml>
```

*Вывод:*

```
{
  "xml": {
    "foo": {
      "@bar": "BAR",
      "#text": "BAZ"
    }
  }
}
```

Глобальные функции JavaScript

В Duktape реализованы дополнительные глобальные функции JavaScript:

- `btoa(data)` - кодирует данные в строку base64.
- `atob(base64_string)` - декодирует строку base64 в буфер Uint8Array.

```
try {
    b64 = btoa("test string");
    buffer = atob(b64);

    //          ,                                   .
    decoded = String.fromCharCode.apply(this, [].slice.call(buffer));
}
catch (error) {
    return {'error.name' : error.name, 'error.message' : error.message};
}
```

- `md5(data)` - вычисляет MD5-хэш данных.
- `sha256(data)` - вычисляет SHA256-хэш данных.
- `hmac('<hash type>',key,data)` - возвращает HMAC-хэш в виде строки в шестнадцатеричном формате; в качестве `hash type` поддерживаются `md5` и `sha256`; параметры `key` и `data` поддерживают двоичные данные.

  Примеры:

  - `hmac('md5',key,data)`
  - `hmac('sha256',key,data)`
- `sign(hash,key,data)` - возвращает вычисленную подпись (RSA-подпись с SHA-256) в виде строки, где:<br> **hash** - допускается только `sha256`, в противном случае будет выдана ошибка.<br> **key** - закрытый ключ. Он должен соответствовать стандарту PKCS#1 или PKCS#8. Ключ может быть передан в разных формах:<br>

  - с пробелами вместо символов новой строки
  - с экранированными или неэкранированными `\n` вместо символов новой строки
  - без символов новой строки, в виде однострочной строки
  - в виде строки в формате JSON

Ключ также можно загрузить из пользовательского макроса/секретного макроса/хранилища.

**data** - данные, которые будут подписаны. Это может быть строка (также поддерживаются двоичные данные) или буфер (Uint8Array/ArrayBuffer).

Пример:

```
- sign('sha256',key,data)
```

Для вычисления подписей используется OpenSSL или GnuTLS. Если Zabbix был собран без какой-либо из этих библиотек шифрования, будет выдана ошибка (`missing OpenSSL or GnuTLS library`).

2. Объекты JavaScript элемента данных «Браузер»

Обзор

В этом разделе описываются расширения Zabbix к языку JavaScript, реализованные с помощью Duktape для использования в скрипте браузерного элемента данных. Эти расширения дополняют объекты JavaScript, описанные на странице Дополнительные объекты JavaScript.

Browser

Объект `Browser` управляет сеансами WebDriver, инициализируя сеанс при создании и завершая его при уничтожении. Один скрипт может поддерживать до четырех объектов `Browser`.

Чтобы создать объект `Browser`, используйте синтаксис `new Browser(options)`. Параметр `options` (*JSON object*) задает параметры браузера и обычно представляет собой результат метода параметров WebDriver (например, `Browser.chromeOptions()`).

С объектом `Browser` поддерживаются следующие методы.

| Method | Description |
|---|---|
| `navigate(url)` | Перейти по указанному URL.<br><br>Parameters:<br>`url` - (string) URL, по которому нужно перейти. |
| `getUrl()` | Возвращает строку с URL открытой страницы. |
| `getPageSource()` | Возвращает строку с исходным кодом открытой страницы. |
| `findElement(strategy, selector)` | Возвращает объект `Element` с одним элементом на открытой странице (или возвращает `null`, если ни один элемент не соответствует `strategy` и `selector`).<br><br>Parameters:<br>`strategy` - (string, css selector/link text/partial link text/tag name/xpath) Location strategy;<br>`selector` - (string) Селектор элемента с использованием указанной стратегии поиска. |
| `findElements(strategy, target)` | Возвращает массив объектов `Element` с несколькими элементами на открытой странице (или возвращает пустой массив, если ни один элемент не соответствует стратегии поиска и target).<br><br>Parameters:<br>`strategy` - (string, css selector/link text/partial link text/tag name/xpath) Location strategy;<br>`target` - (string) Селектор элемента с использованием указанной стратегии поиска. |
| `getCookies()` | Возвращает массив объектов `Cookie`. |
| `addCookie(cookie)` | Установить cookies.<br><br>Parameters:<br>`cookie` - (object `Cookie`) Cookie, которую нужно установить. |
| `getScreenshot()` | Возвращает строку, закодированную в base64, представляющую изображение области просмотра браузера. |

| Method | Description |
|---|---|
| `setScreenSize(x,y)` | Установить размер области просмотра браузера.<br><br>Parameters:<br>`x` - (string) Ширина в пикселях;<br>`y` - (string) Высота в пикселях. |
| `setScriptTimeout(timeout)` | Установить тайм-аут загрузки скрипта.<br><br>Parameters:<br>`timeout` - (integer) Значение тайм-аута в миллисекундах. |
| `setSessionTimeout(timeout)` | Установить тайм-аут сеанса (загрузки страницы).<br><br>Parameters:<br>`timeout` - (integer) Значение тайм-аута в миллисекундах. |
| `setElementWaitTimeout(timeout)` | Установить тайм-аут стратегии поиска элемента (неявный).<br><br>Parameters:<br>`timeout` - (integer) Значение тайм-аута в миллисекундах. |
| `collectPerfEntries(mark)` | Собрать записи производительности для получения с помощью метода `getResult()`.<br><br>Parameters:<br>`mark` - (string, optional) метка снимка производительности. |
| `getRawPerfEntries()` | Возвращает массив объектов записи производительности. |
| `getResult()` | Возвращает объект `Result` со статистикой сеанса браузера (сведения об ошибках, снимки производительности и т. д.). |
| `getError()` | Возвращает объект `BrowserError` с ошибками браузера (или возвращает `null`, если ошибок браузера нет). |
| `setError(message)` | Задать собственное сообщение об ошибке, которое будет включено в объект `Result`.<br><br>Parameters:<br>`message` - (string) Сообщение об ошибке. |
| `discardError()` | Отбросить ошибку, которая будет возвращена в объекте `Result`. |
| `getAlert()` | Возвращает объект `Alert` с предупреждениями браузера (или возвращает `null`, если предупреждений браузера нет). |
| `chromeOptions()` | Возвращает объект `chromeOptions` с предопределенными параметрами браузера Chrome. |
| `firefoxOptions()` | Возвращает объект `firefoxOptions` с предопределенными параметрами браузера Firefox. |
| `safariOptions()` | Возвращает объект `safariOptions` с предопределенными параметрами браузера Safari. |
| `edgeOptions()` | Возвращает объект `edgeOptions` с предопределенными параметрами браузера Edge. |
| `switchFrame(target)` | Переключиться на указанный фрейм.<br><br>Parameters:<br>`target` - (browser element or integer, optional) Целевой фрейм. Чтобы выбрать фрейм по элементу, передайте элемент. Чтобы выбрать фрейм по индексу, передайте число. Если оставить пустым, будет выполнено переключение на контекст просмотра верхнего уровня. |

Все методы Browser могут вызывать следующие ошибки:

- `BrowserError` - производная от объекта `Error`, которая возникает, если конструктор `Browser` завершается с ошибкой; содержит дополнительное свойство `browser` с объектом `Browser`, который вызвал эту ошибку `BrowserError`.
- `WebdriverError` - производная от `BrowserError`; содержит те же свойства, что и объект `BrowserError`, и указывает, была ли ошибка сгенерирована в ответ на ошибку в ответе WebDriver.

Element

Объект `Element` возвращается методами `findElement()`/`findElements()` объекта `Browser` и не может быть создан

напрямую.

Объект `Element` представляет элемент на веб-странице и предоставляет методы для взаимодействия с ним.

Для объекта `Element` поддерживаются следующие методы.

| Method | Description |
|---|---|
| `getAttribute(name)` | Возвращает строковое значение атрибута элемента (или `null`, если указанный атрибут не найден).<br><br>Parameters:<br>`name` - (string) Имя атрибута. |
| `getProperty(name)` | Возвращает строковое значение свойства элемента (или `null`, если указанное свойство не найдено).<br><br>Parameters:<br>`name` - (string) Имя свойства. |
| `getText()` | Возвращает строковое значение текста элемента. |
| `click()` | Щелкает по элементу. |
| `clear()` | Очищает содержимое редактируемого элемента. |
| `sendKeys(keys)` | Отправляет клавиши.<br><br>Parameters:<br>`keys` - (string) Клавиши для отправки. |

Cookie

Объект `Cookie` возвращается методом `getCookies()` объекта `Browser` и передаётся в метод `addCookie()`.

Хотя у объекта `Cookie` нет методов, он может содержать следующие свойства:

| Свойство | Тип | Описание |
|---|---|---|
| name | строка | Имя cookie. |
| value | строка | Значение cookie. |
| path | строка | Путь, для которого cookie действителен.<br>По умолчанию `"/"`, если не указан при добавлении cookie. |
| domain | строка | Домен, в котором cookie является видимым.<br>По умолчанию — домен URL активного документа текущего контекста просмотра сессии, если не указан при добавлении cookie |
| secure | логический | Указывает, является ли cookie безопасным.<br>По умолчанию `false`, если не указан при добавлении cookie. |
| httpOnly | логический | Указывает, является ли cookie HTTP-only.<br>По умолчанию `false`, если не указан при добавлении cookie. |
| expiry | целое число | Время истечения срока действия cookie (в секундах с начала эпохи Unix). Не должно быть установлено, если не указано при добавлении файла cookie. |
| sameSite | строка | Атрибут `sameSite` файла cookie, который управляет тем, должен ли файл cookie быть ограничен контекстом первой стороны (first-party) или того же сайта.<br>Может быть установлено как `"Lax"` или `"Strict"`.<br>По умолчанию `"None"`, если не указан при добавлении cookie. |

Alert

Объект `Alert` представляет оповещение (alert) веб-страницы, возвращается методом `getAlert()` объекта `Browser` и не может быть создан напрямую.

Объект `Alert` содержит свойство `text` с текстом оповещения (или `null`, если оповещения отсутствуют).

Объектом `Alert` поддерживаются следующие методы.

| Метод | Описание |
|---|---|
| `accept()` | Принять оповещение. |
| `dismiss()` | Отклонить оповещение. |

Result

Объект `Result` содержит статистику сессии и возвращается методом `getResult()` объекта `Browser`.

Обычно объект `Result` преобразуется в строку и возвращается из скрипта, а затем разбирается на значения зависимых элементов данных посредством предварительной обработки.

Хотя объект `Result` не имеет методов, он может содержать следующие свойства.

| Свойство | Тип | Описание |
|---|---|---|
| duration | строка | Длительность сессии от создания сессии до получения результата. |
| error | объект | Информация об ошибке. |
| http_status | целое число | Статус HTTP, возвращаемый WebDriver'ом (или 0, если ошибок WebDriver'а нет). |
| error_code | строка | Ошибка, возвращаемая WebDriver'ом (или пустая строка, если ошибок WebDriver'а нет). |
| message | строка | Сообщение об ошибке WebDriver'а (или пустая строка, если ошибок WebDriver'а нет). |
| performance_data | объект | Статистика производительности. |
| summary | объект | Сводка производительности. |
| navigation | объект | Сводка навигации. |
| resource | объект | Сводка ресурсов. |
| details | массив объектов | Статистика производительности после каждой операции, которая могла привести к навигации. |
| mark | строка | *(опционально)* Отметка снимка производительности, указанная с помощью метода `collectPerfEntries()`. |
| navigation | объект | Статистика навигации. |
| resource | объект | Сводка ресурсов для этого шага. |
| user | массив объектов | Массив статистики отметки / типа измерения. |
| marks | массив объектов | Отмеченные индексы моментальных снимков производительности. |
| name | строка | Имя отметки моментального снимка производительности. |
| index | целое число | Индекс моментального снимка производительности в массиве details. |

### 6. Предобработка CSV в JSON

Обзор

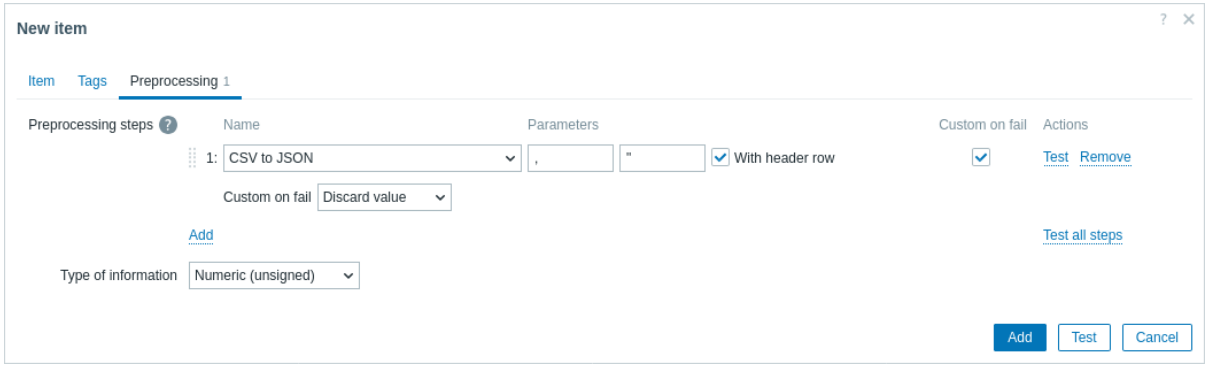
В этом шаге предварительной обработки имеется возможность преобразования данных CSV файла в формат JSON. Поддерживается в:

- элементах данных (прототипах элементов данных)
- правилах низкоуровневого обнаружения

Конфигурация

Чтобы настроить шаг предварительной обработки CSV в JSON:

- Перейдите на вкладку Предварительная обработка в конфигурации элемента данных/правила обнаружения
- Нажмите *Добавить*
- Выберите опцию *CSV to JSON*




Первый параметр позволяет задать пользовательский разделитель. Обратите внимание: если первая строка входных данных CSV начинается с "Sep=" и за ней следует один символ UTF-8, то этот символ будет использоваться в качестве разделителя, если первый параметр не задан. Если первый параметр не задан и разделитель не был получен из строки "Sep=", то в качестве разделителя используется запятая.

Второй необязательный параметр позволяет задать символ кавычек.

Если установлен флажок *With header row*, значения строки заголовка будут интерпретироваться как имена столбцов (подробнее см. Обработка заголовка).

Если установлен флажок *Custom on fail*, элемент данных не станет неподдерживаемым в случае сбоя шага предварительной обработки. Кроме того, можно задать пользовательские параметры обработки ошибок: отбросить значение, установить указанное значение или установить указанное сообщение об ошибке.

Обработка заголовка

Строка заголовка CSV файла может обрабатываться двумя разными способами:

- Если флажок *Со строкой заголовка* отмечен — значения строки заголовка интерпретируются как имена столбцов. В этом случае имена столбцов должны быть уникальны и строки с данными не должны содержать больше столбцов, чем строка заголовка;
- Если флажок *Со строкой заголовка* не отмечен — строка заголовка интерпретируется как данные. Имена столбцов генерируются автоматически (1,2,3,4...)

Пример CSV файла:

```
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:**
Знак кавычек внутри заключённого в кавычки поля во входящих данных должен быть экранирован при помощи другого символа кавычек.

**Обработка строки заголовка**

Вывод JSON, когда ожидается строка заголовка:

```
[
   {
      "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"
   }
]
```

**Обработка без наличия строки заголовка**

Вывод JSON, когда не ожидается строка заголовка:

```
[
   {
      "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"
   }
]
```

### 3. Типы элементов данных

Обзор

Типы элементов данных охватывают различные методы получения данных из вашей системы. Каждый тип элемента данных поставляется со своим набором поддерживаемых ключей элементов данных и обязательных параметров.

В настоящее время Zabbix предлагает следующие типы элементов данных:

- агент Zabbix
  - Мониторинг файлов журнала
- Простая проверка
  - Мониторинг VMware
- SNMP agent
- SNMP trap
- Внутренний Zabbix
- Zabbix trapper
- Внешняя проверка
- Монитор базы данных
- HTTP agent
  - Проверка Prometheus
- агент IPMI
- агент SSH
- агент Telnet
- агент JMX
- Вычисляемый элемент данных
  - Агрегатные вычисления
- Зависимый элемент данных
- Элемент данных скрипта
- Элемент данных браузера

Подробные сведения обо всех типах элементов данных приведены на подстраницах этого раздела. Хотя типы элементов данных предоставляют множество возможностей для сбора данных, существуют дополнительные варианты через пользовательские параметры или загружаемые модули.

Некоторые проверки выполняются только сервером Zabbix (как мониторинг без агента), в то время как другие требуют Zabbix agent или даже Zabbix Java gateway (при мониторинге JMX).

**Attention:**
Если для определенного типа элемента данных требуется определенный интерфейс (например, проверка IPMI требует интерфейс IPMI на узле сети), такой интерфейс должен существовать в определении узла сети.

В определении узла сети можно задать несколько интерфейсов: Zabbix agent, SNMP agent, JMX и IPMI. Если элемент данных может использовать более одного интерфейса, он будет искать доступные интерфейсы узла сети (в порядке: Agent→SNMP→JMX→IPMI) и связываться с первым подходящим.

Все элементы данных, которые возвращают текст (типы информации character, log, text), могут также возвращать только пробелы (где применимо), устанавливая возвращаемое значение в пустую строку (поддерживается с версии 2.0).

### 1. Zabbix агент

Обзор

В этом разделе приведены сведения о ключах элементов данных, которые для сбора данных используют общение с Zabbix агентом.

Существуют пассивные и активные проверки. При настройке элемента данных вы можете выбрать требуемый тип:

- *Zabbix агент* — для пассивных проверок
- *Zabbix агент (активный)* — для активных проверок

Обратите внимание, что все ключи элементов данных, поддерживаемые агентом Zabbix, также поддерживаются новым поколением Zabbix агента 2. Смотрите дополнительные ключи элементов данных, которые можно использовать только с агентом 2.

Поддерживаемые ключи элементов данных

Ниже перечислены ключи элементов данных, которые можно использовать с агентом Zabbix.

Ключи элементов данных перечислены без параметров и дополнительной информации. Щелкните ключ элемента данных, чтобы увидеть полные сведения.

| Item key | Description | Item group |
|---|---|---|
| kernel.maxfiles | Максимальное число открытых файлов, поддерживаемое ОС. | Kernel |
| kernel.maxproc | Максимальное число процессов, поддерживаемое ОС. | |
| kernel.openfiles | Количество файловых дескрипторов, открытых в данный момент. | |
| log | Мониторинг файла журнала. | Log monitoring |
| log.count | Количество совпавших строк в контролируемом файле журнала. | |
| logrt | Мониторинг файла журнала с ротацией. | |
| logrt.count | Количество совпавших строк в контролируемом файле журнала с ротацией. | |
| modbus.get | Чтение данных Modbus. | Modbus |
| net.dns | Проверка состояния службы DNS. | Network |
| net.dns.perf | Проверка производительности службы DNS. | |
| net.dns.record | Выполнение DNS-запроса. | |
| net.if.collisions | Количество коллизий вне окна. | |
| net.if.discovery | Список сетевых интерфейсов. | |
| net.if.in | Статистика входящего трафика на сетевом интерфейсе. | |
| net.if.out | Статистика исходящего трафика на сетевом интерфейсе. | |
| net.if.total | Сумма статистики входящего и исходящего трафика на сетевом интерфейсе. | |
| net.tcp.listen | Проверка, находится ли этот TCP-порт в состоянии LISTEN. | |
| net.tcp.port | Проверка возможности установить TCP-соединение с указанным портом. | |
| net.tcp.service | Проверка, запущена ли служба и принимает ли она TCP-соединения. | |
| net.tcp.service.perf | Проверка производительности службы TCP. | |
| net.tcp.socket.count | Возвращает количество TCP-сокетов, соответствующих параметрам. | |
| net.udp.listen | Проверка, находится ли этот UDP-порт в состоянии LISTEN. | |
| net.udp.service | Проверка, запущена ли служба и отвечает ли она на UDP-запросы. | |

| Item key | Description | Item group |
|---|---|---|
| net.udp.service.perf | Проверка производительности службы UDP. | |
| net.udp.socket.count | Возвращает количество UDP-сокетов, соответствующих параметрам. | |
| proc.cpu.util | Процент использования CPU процессом. | Processes |
| proc.get | Список процессов ОС и их параметров. | |
| proc.mem | Объем памяти, используемый процессом, в байтах. | |
| proc.num | Количество процессов. | |
| sensor | Показания аппаратного датчика. | Sensors |
| system.boottime | Время загрузки системы. | System |
| system.cpu.discovery | Список обнаруженных CPU/ядер CPU. | |
| system.cpu.intr | Прерывания устройства. | |
| system.cpu.load | Нагрузка CPU. | |
| system.cpu.num | Количество CPU. | |
| system.cpu.switches | Количество переключений контекста. | |
| system.cpu.util | Процент использования CPU. | |
| system.hostname | Имя узла сети системы. | |
| system.hw.chassis | Сведения о корпусе. | |
| system.hw.cpu | Сведения о CPU. | |
| system.hw.devices | Список устройств PCI или USB. | |
| system.hw.macaddr | Список MAC-адресов. | |
| system.localtime | Системное время. | |
| system.run | Выполнение указанной команды на узле сети. | |
| system.stat | Системная статистика. | |
| system.sw.arch | Сведения об архитектуре программного обеспечения. | |
| system.sw.os | Сведения об операционной системе. | |
| system.sw.os.get | Подробные сведения об операционной системе (версия, тип, название дистрибутива, минорная и мажорная версии и т. д). | |
| system.sw.packages | Список установленных пакетов. | |
| system.sw.packages.get | Подробный список установленных пакетов. | |
| system.swap.in | Статистика swap-in (из устройства в память). | |
| system.swap.out | Статистика swap-out (из памяти на устройство). | |
| system.swap.size | Размер пространства подкачки в байтах или в процентах от общего объема. | |
| system.uname | Идентификация системы. | |
| system.uptime | Время работы системы в секундах. | |
| system.users.num | Количество вошедших в систему пользователей. | |
| vfs.dev.discovery | Список блочных устройств и их тип. | Virtual file systems |
| vfs.dev.read | Статистика чтения диска. | |
| vfs.dev.write | Статистика записи на диск. | |
| vfs.dir.count | Количество записей каталога. | |
| vfs.dir.get | Список записей каталога. | |
| vfs.dir.size | Размер каталога. | |
| vfs.file.cksum | Контрольная сумма файла, вычисленная алгоритмом UNIX cksum. | |
| vfs.file.contents | Получение содержимого файла. | |
| vfs.file.exists | Проверка существования файла. | |
| vfs.file.get | Возвращает сведения о файле. | |
| vfs.file.md5sum | MD5-контрольная сумма файла. | |
| vfs.file.owner | Получение владельца файла. | |
| vfs.file.permissions | Возвращает 4-значную строку, содержащую восьмеричное число с правами UNIX. | |
| vfs.file.regexp | Получение строки в файле. | |
| vfs.file.regmatch | Поиск строки в файле. | |
| vfs.file.size | Размер файла. | |
| vfs.file.time | Сведения о времени файла. | |
| vfs.fs.discovery | Список смонтированных файловых систем с их типом и параметрами монтирования. | |
| vfs.fs.get | Список смонтированных файловых систем с их типом, доступным дисковым пространством, статистикой inode и параметрами монтирования. | |
| vfs.fs.inode | Количество или процент inode. | |
| vfs.fs.size | Дисковое пространство в байтах или в процентах от общего объема. | |
| vm.memory.size | Объем памяти в байтах или в процентах от общего объема. | Virtual memory |

| Item key | Description | Item group |
|---|---|---|
| web.page.get | Получение содержимого веб-страницы. | Web monitoring |
| web.page.perf | Время загрузки полной веб-страницы. | |
| web.page.regexp | Поиск строки на веб-странице. | |
| agent.hostmetadata | Метаданные узла сети агента. | Zabbix |
| agent.hostname | Имя узла сети агента. | |
| agent.ping | Проверка доступности агента. | |
| agent.variant | Вариант агента Zabbix (Zabbix agent или Zabbix agent 2). | |
| agent.version | Версия агента Zabbix. | |
| zabbix.stats | Возвращает набор внутренних метрик сервера Zabbix или прокси. Если указаны `ip` и `port`, метрики извлекаются с удаленного экземпляра; в противном случае — с локального экземпляра. | |
| zabbix.stats (queue) | Возвращает количество контролируемых элементов данных в очереди, задержанных на сервере Zabbix или прокси. Если указаны `ip` и `port`, метрики извлекаются с удаленного экземпляра; в противном случае — с локального экземпляра. | |

Поддерживаемые платформы

Если в описании элемента данных не указано иное, элементы данных агента (и все параметры) поддерживаются на:

- **Linux**
- **FreeBSD**
- **Solaris**
- **HP-UX**
- **AIX**
- **MacOS X**
- **OpenBSD**
- **NetBSD**

Многие элементы данных агента также поддерживаются на **Windows**. Подробности смотрите на странице Элементы данных агента Windows.

Подробности ключей элементов данных

Параметры без угловых скобок обязательны. Параметры, обозначенные угловыми скобками **< >**, опциональны.

kernel.maxfiles

<br> Максимальное количество открытых файлов, поддерживаемое ОС.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, MacOS X, OpenBSD, NetBSD.

kernel.maxproc

<br> Максимальное количество процессов, поддерживаемое ОС.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux 2.6 и новее, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

kernel.openfiles

<br> Количество открытых в данный момент файловых дескрипторов.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux (элемент данных может работать и на других UNIX-подобных платформах).

log[файл,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<вывод>,<максзадержка>,<опции>,<постоянн

<br> Мониторинг файлов журналов (логов).<br> Возвращаемое значение: *Журнал (лог)*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь и имя файла журнала;<br>
- **регулярное выражение** — регулярное выражение, описывающее необходимый шаблон содержимого;<br>
- **кодировка** — идентификатор кодовой страницы;<br>
- **макс. кол-во строк** — максимальное количество новых строк в секунду, которое агент будет отправлять Zabbix серверу или прокси. Этот параметр переопределяет значение «MaxLinesPerSecond» в zabbix_agentd.conf;<br>
- **режим** — возможные значения: *all* (по умолчанию) или *skip* — пропуск обработки старых данных (влияет только на недавно созданные элементы данных);<br>
- **вывод** — опциональный шаблон форматирования вывода. Управляющая последовательность **\0** заменяется найденной частью текста (начиная от первого символа совпадения, заканчивая символом, где совпадение

заканчивается), тогда как управляющая последовательность **\N** (где N=1...9) заменяется N-ой совпадающей группой (или пустой строкой, если N превышает количество найденных групп);<br>

- **максзадержка** — максимальная задержка в секундах. Тип: число с плавающей точкой. Значения: 0 — (по умолчанию) никогда не игнорировать строки в файлах журналов; > 0.0 — игнорировать более старые строки с целью получения наиболее новых строк, проанализированных в течение «максзадержка» секунд. Перед использованием прочитайте заметки по максзадержка!<br>
- **опции** — дополнительные опции:<br>*mtime-noreread* — не уникальные записи, перечитывать только, если размер файла изменяется (игнорировать изменение времени модификации). (Этот параметр устарел, начиная с 5.0.2, так как сейчас mtime игнорируется.)<br>
- **постоянное_хранилище** (только в zabbix_agentd на Unix системах; не поддерживается в Agent2) — абсолютный путь к директории, где хранить данные долговременного хранения. Смотрите также дополнительные заметки о файлах долговременного хранения.

Комментарии:

- Элемент данных должен быть настроен как активная проверка;
- Если файл не существует или доступ к нему не разрешён, элемент данных перейдёт в неподдерживаемое состояние;
- Если параметр оставить пустым — будет возвращена вся строка, содержащая текст, соответствующий регулярному выражению. Обратите внимание, что все глобальные регулярные выражения, за исключением «Результат ИСТИНА», всегда возвращают всю строку, соответствующую выражению, и параметр игнорируется.
- Извлечение содержимого согласно параметру производится на стороне агента.

Примеры:

```
log[/var/log/syslog]
log[/var/log/syslog,error]
log[/home/zabbix/logs/logfile,,,100]
```

Пример использования параметра для извлечения числа из записи в журнале:

```
log[/app1/app.log,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #
```

Пример использования параметра для изменения вывода записи журнала перед отправкой на сервер:

```
log[/app1/app.log,"([0-9 :-]+) task run ([0-9.]+) sec, processed ([0-9]+) records, ([0-9]+) errors",,,,"\1
```

log.count[файл,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<максзадержка>,<опции>,<постоянное_х

<br> Количество совпадающих строк в наблюдаемом файле журнала (логе).<br> Возвращаемое значение: *Целое число*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь и имя файла журнала;<br>
- **регулярное выражение** — регулярное выражение, описывающее необходимый шаблон содержимого;<br>
- **кодировка** — идентификатор кодовой страницы;<br>
- **макс. кол-во строк** — максимальное количество новых строк в секунду, которое агент будет анализировать (не может превышать 10000). Значением по умолчанию является 10*«MaxLinesPerSecond» в zabbix_agentd.conf.<br>
- **режим** — возможные значения: *all* (по умолчанию) или *skip* — пропуск обработки старых данных (влияет только на недавно созданные элементы данных).<br>
- **максзадержка** — максимальная задержка в секундах. Тип: число с плавающей точкой. Значения: 0 — (по умолчанию) никогда не игнорировать строки в файлах журналов; > 0.0 — игнорировать более старые строки с целью получения наиболее новых строк, проанализированных в течение «максзадержка» секунд. Перед использованием прочитайте заметки по максзадержка!<br>
- **опции** — дополнительные опции:<br>*mtime-noreread* — не уникальные записи, перечитывать только, если размер файла изменяется (игнорировать изменения времени модификации). (Этот параметр устарел, начиная с 5.0.2, так как сейчас mtime игнорируется.)<br>
- **постоянное_хранилище** (только в zabbix_agentd на Unix системах; не поддерживается в Agent2) — абсолютный путь к директории, где хранить данные долговременного хранения. Смотрите также дополнительные заметки о файлах долговременного хранения.

Комментарии:

- Элемент данных должен быть настроен как активная проверка;
- Количество совпадающих строк вычисляется за счёт новых строк, начиная со времени последней проверки журнала агентом; таким образом, количество зависит от интервала обновления элемента данных;
- Если файл не существует или доступ к нему не разрешён, элемент данных переходит в неподдерживаемое состояние.

logrt[регулярное выражение файла,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<вывод>,<максзадерж

<br> Мониторинг файлов журналов (логов) с поддержкой ротации файлов журналов.<br> Возвращаемое значение: *Журнал (лог)*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **регулярное выражение файла** — абсолютный путь к файлу и регулярное выражение, описывающее его имя. Обратите внимание, что регулярное выражение применяется только к имени файла и не обязательно должно соответствовать всему имени (например, /path/to/agent будет соответствовать zabbix_agentd.log).<br>
- **регулярное выражение** — регулярное выражение, описывающее необходимый шаблон содержимого;<br>
- **кодировка** — идентификатор кодовой страницы;<br>
- **макс. кол-во строк** — максимальное количество новых строк в секунду, которое агент будет отправлять Zabbix серверу или прокси. Этот параметр переопределяет значение «MaxLinesPerSecond» в zabbix_agentd.conf.<br>
- **режим** возможные значения: *all* (по умолчанию) или *skip* — пропуск обработки старых данных (влияет только на недавно созданные элементы данных).<br>
- **вывод** — опциональный шаблон форматирования вывода. Управляющая последовательность **\0** заменяется найденной частью текста (начиная от первого символа совпадения, заканчивая символом, где совпадение заканчивается), тогда как управляющая последовательность **\N** (где N=1...9) заменяется N-ой совпадающей группой (или пустой строкой, если N превышает количество найденных групп).<br>
- **максзадержка** — максимальная задержка в секундах. Тип: число с плавающей точкой. Значения: 0 — (по умолчанию) никогда не игнорировать строки в файлах журналов; > 0.0 — игнорировать более старые строки с целью получения наиболее новых строк, проанализированных в течение «максзадержка» секунд. Перед использованием прочитайте заметки по максзадержка!<br>
- **опции** — тип ротации файлов журналов и другие опции. Возможные значения:<br>*rotate* (по умолчанию),<br>*copytruncate* — обратите внимание, что *copytruncate* нельзя использовать вместе с *maxdelay*. В этом случае *maxdelay* должен быть равен 0 или не указан; смотрите заметки о copytruncate,<br>*mtime-reread* — не уникальные записи, перечитывать, если время модификации или размер изменились (используется по умолчанию, если явно не указана опция `mtime-*`),<br>*mtime-noreread* — не уникальные записи, перечитывать только если размер файла изменяется (игнорирование изменения времени модификации).<br>
- **постоянное_хранилище** (только в zabbix_agentd на Unix системах; не поддерживается в Agent2) — абсолютный путь к директории, где хранить данные долговременного хранения. Смотрите также дополнительные заметки о файлах долговременного хранения.

Комментарии:

- Элемент данных должен быть настроен как активная проверка;
- Ротация журналов основывается на времени последней модификации файлов;
- Обратите внимание, что logrt разработан для работы с одним активным файлом журнала, с несколькими другими совпадающими уже устаревшими неактивными файлами. Если, например, директория содержит множество активных файлов журналов, по каждому такому файлу необходимо создавать отдельный элемент данных logrt. В противном случае, если один элемент данных logrt подхватит слишком много файлов, это может привести к исчерпанию памяти и сбою в мониторинге.
- Если параметр оставить пустым — будет возвращена вся строка, содержащая текст, соответствующий регулярному выражению. Обратите внимание, что все глобальные регулярные выражения, за исключением «Результат ИСТИНА», всегда возвращают всю соответствующую выражению строку, и параметр игнорируется.
- Извлечение содержимого согласно параметру производится на стороне агента.
- В параметре _ _ путь к директории файла журнала и регулярное выражение имени файла журнала должны быть разделены правильным разделителем:
  - в Windows разделителем должна быть обратная косая черта (\); прямая косая черта может допускаться в других позициях, за исключением той, которая разделяет путь к директории файла журнала и регулярное выражение имени файла журнала (смотрите примеры ниже).
  - в других системах это должна быть прямая косая черта (/).

Примеры для Windows:

```
logrt["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #                          «c:/dir1/dir2/di
logrt["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #                -   ,
```

Примеры для других систем:

```
logrt["/home/zabbix/logs/^logfile[0-9]{1,3}$",,,100] #                «logfile1» (           «.logfile1»
logrt["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #
```

Пример использования параметра для извлечения числа из записи в журнале:

```
logrt[/app1/^test.*log$,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #
```

Пример использования параметра для изменения вывода записи журнала перед отправкой на сервер:

```
logrt[/app1/^test.*log$,"([0-9 :-]+) task run ([0-9.]+) sec, processed ([0-9]+) records, ([0-9]+) errors",
```

logrt.count[регулярное выражение файла ,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<максзадержка

<br> Количество совпадающих строк в наблюдаемом файле журнала (логе) с поддержкой ротации файлов журналов.<br> Возвращаемое значение: *Целое число*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **регулярное выражение файла** — абсолютный путь к файлу и регулярное выражение, описывающее его имя. Обратите внимание, что регулярное выражение применяется только к имени файла и не обязательно должно соответствовать всему имени (например, /path/to/agent будет соответствовать zabbix_agentd.log).<br>
- **регулярное выражение** — регулярное выражение, описывающее необходимый шаблон содержимого;<br>
- **кодировка** — идентификатор кодовой страницы;<br>
- **макс. кол-во строк** — максимальное количество новых строк в секунду, которое агент будет анализировать (не может превышать 10000). Значением по умолчанию является 10*«MaxLinesPerSecond» в zabbix_agentd.conf.<br>
- **режим** — возможные значения: *all* (по умолчанию) или *skip* — пропуск обработки старых данных (влияет только на недавно созданные элементы данных).<br>
- **максзадержка** — максимальная задержка в секундах. Тип: число с плавающей точкой. Значения: 0 — (по умолчанию) никогда не игнорировать строки в файлах журналов; > 0.0 — игнорировать более старые строки с целью получения наиболее новых строк, проанализированных в течение «максзадержка» секунд. Перед использованием прочитайте заметки по максзадержка!<br>
- **опции** — тип ротации файлов журналов и другие опции. Возможные значения:<br>*rotate* (по умолчанию),<br>*copytruncate* — обратите внимание, что *copytruncate* нельзя использовать вместе с *maxdelay*. В этом случае *maxdelay* должен быть равен 0 или не указан; смотрите заметки о copytruncate,<br>*mtime-reread* — не уникальные записи, перечитывать, если время модификации или размер изменились (по умолчанию),<br>*mtime-noreread* — не уникальные записи, перечитывать только если размер файла изменяется (игнорирование изменения времени модификации).<br>
- **постоянное_хранилище** (только в zabbix_agentd на Unix системах; не поддерживается в Agent2) — абсолютный путь к директории, где хранить данные долговременного хранения. Смотрите также дополнительные заметки о файлах долговременного хранения.

Комментарии:

- Элемент данных должен быть настроен как активная проверка;
- Количество совпадающих строк вычисляется за счёт новых строк, начиная со времени последней проверки журнала агентом; таким образом, количество зависит от интервала обновления элемента данных;
- Ротация журналов основывается на времени последней модификации файлов;
- В параметре _ _ путь к директории файла журнала и регулярное выражение имени файла журнала должны быть разделены правильным разделителем:
  - в Windows разделителем должна быть обратная косая черта (\); прямая косая черта может допускаться в других позициях, за исключением той, которая разделяет путь к директории файла журнала и регулярное выражение имени файла журнала (смотрите примеры ниже).
  - в других системах это должна быть прямая косая черта (/).

Примеры для Windows:

```
logrt.count["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #                    «pattern_to_m
logrt.count["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #
```

Примеры для других систем:

```
logrt.count["/home/zabbix/logs/.*","^logfile[0-9]{1,3}$",,100] #                    «^logfile[0-9]{1,3
logrt.count["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #       ,          «pat
```

modbus.get[endpoint,<slave id>,<function>,<address>,<count>,<type>,<endianness>,<offset>]

<br> Читает данные Modbus.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux, Windows.

Параметры:

- **endpoint** - конечная точка, заданная как `protocol://connection_string`;<br>
- **slave id** - ID ведомого устройства;<br>
- **function** - функция Modbus;<br>
- **address** - адрес первого регистра, катушки или входа;<br>
- **count** - количество записей для чтения;<br>
- **type** - тип данных;<br>

- **endianness** - порядок байтов;<br>
- **offset** - количество регистров, начиная с 'address', результаты которых будут отброшены.

В Windows этот элемент данных следует UNC-путям в параметре **endpoint**, что может представлять риск безопасности.

См. подробное описание параметров.

net.dns[<ip>,имя,<тип>,<время ожидания>,<количество>,<протокол>]

<br> Проверка состояния службы DNS.<br> Возвращаемые значения: 0 — DNS не работает (сервер не ответил или ответ DNS некорректен); 1 — DNS работает.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **ip** (на Windows поддерживается Zabbix агентом 2, игнорируется Zabbix агентом) — IP адрес DNS сервера (оставьте пустым, чтобы использовались DNS сервера по умолчанию);
- **имя** — проверяемое имя DNS;
- **тип** — запрашиваемый тип записи (по умолчанию *SOA*);
- **время ожидания** (игнорируется в Windows, если только не используется Zabbix агент 2) — время ожидания ответа в секундах (по умолчанию 1 секунда);
- **количество** (игнорируется в Windows, если только не используется Zabbix агент 2) — количество попыток запросов (по умолчанию 2);
- **протокол** — используемый протокол при выполнении DNS запросов: *udp* (по умолчанию) или *tcp*.

Комментарии:

- Возможные значения для : *ANY*, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (не поддерживается Zabbix агентом на Windows, Zabbix агентом 2 на всех ОС), *HINFO*, *MINFO*, *TXT*, *SRV*
- Для обратных DNS-запросов (когда `type` установлен на *PTR*) вы можете указать DNS-имя как в обратном, так и в необратном формате (см. примеры ниже). Обратите внимание, что при запросе записи PTR имя DNS на самом деле является IP-адресом.
- Интернационализированные доменные имена не поддерживаются; вместо них, пожалуйста, используйте имена в IDNA кодировке.

Примеры:

```
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>]

<br> Проверяет производительность службы DNS.<br> Возвращаемое значение: *Float* (0 - служба недоступна; seconds - количество секунд, потраченных на ожидание ответа от службы).<br> См. поддерживаемые платформы.

Параметры:

- **ip** (игнорируется в Windows, если не используется Zabbix agent 2) - IP-адрес DNS-сервера (оставьте пустым для DNS-сервера по умолчанию);
- **name** - DNS-имя для запроса;
- **type** - тип записи, которую нужно запросить (по умолчанию *SOA*);
- **timeout** (игнорируется в Windows, если не используется Zabbix agent 2) - время ожидания запроса в секундах (по умолчанию 1 секунда);
- **count** (игнорируется в Windows, если не используется Zabbix agent 2) - количество попыток выполнения запроса (по умолчанию 2);
- **protocol** - протокол, используемый для выполнения DNS-запросов: *udp* (по умолчанию) или *tcp*.

Комментарии:

- Возможные значения `type`:<br>*ANY*, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (не поддерживается для Zabbix agent в Windows, Zabbix agent 2 во всех ОС), *HINFO*, *MINFO*, *TXT*, *SRV*.
- Для обратных DNS-запросов (когда `type` установлен в *PTR*) можно указать DNS-имя как в обратном, так и в прямом формате (см. примеры ниже). Обратите внимание, что при запросе записи PTR DNS-имя фактически является IP-адресом.
- Имена доменов в интернационализированном формате не поддерживаются, используйте вместо них имена, закодированные в IDNA.

- Элемент данных возвращает время ответа вместо 0, если DNS-сервер отвечает кодом ошибки (например, NXDOMAIN или SERVFAIL).

Примеры:

```
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>,имя,<тип>,<время ожидания>,<количество>,<протокол>]

<br> Выполнение DNS запроса.<br>

Zabbix агент 2 также предоставляет ключ net.dns.get, который предлагает дополнительные возможности, такие как больший набор типов записей и больший контроль над мониторингом DNS. Возвращаемое значение: строка символов с требуемым типом информации.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **ip** (игнорируется в Windows, если только не используется Zabbix агент 2) — IP адрес DNS сервера (оставьте пустым, чтобы использовались DNS сервера по умолчанию);
- **имя** — проверяемое имя DNS;
- **тип** — запрашиваемый тип записи (по умолчанию *SOA*);
- **время ожидания** (игнорируется в Windows, если только не используется Zabbix агент 2) — время ожидания ответа в секундах (по умолчанию 1 секунда);
- **количество** (игнорируется в Windows, если только не используется Zabbix агент 2) — количество попыток запросов (по умолчанию 2);
- **протокол** — используемый протокол при выполнении DNS запросов: *udp* (по умолчанию) или *tcp*.

Комментарии:

- Возможные значения для :<br>*ANY*, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (не поддерживается Zabbix агентом на Windows, Zabbix агентом 2 на всех ОС), *HINFO*, *MINFO*, *TXT*, *SRV*
- Для обратных DNS-запросов (когда type установлен на *PTR*) вы можете указать DNS-имя как в обратном, так и в необратном формате (см. примеры ниже). Обратите внимание, что при запросе записи PTR имя DNS на самом деле является IP-адресом.
- Интернационализированные доменные имена не поддерживаются; вместо них, пожалуйста, используйте имена в IDNA кодировке.

Примеры:

```
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]

<br> Количество коллизий за пределами окна (out-of-window).<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD. На NetBSD требуются права суперпользователя (root).

Параметр:

- **if** — имя сетевого интерфейса

net.if.discovery

<br> Список сетевых интерфейсов. Используется для обнаружения на низком уровне.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

net.if.in[if,<режим>]

<br> Статистика по входящему трафику сетевого интерфейса.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. На NetBSD требуются права суперпользователя (root).

Параметры:

- **if** — имя сетевого интерфейса (Unix); полное описание сетевого интерфейса или IPv4 адрес; или, если в фигурных скобках, GUID сетевого интерфейса (Windows);
- **режим** — возможные значения:<br>*bytes* — количество байт (по умолчанию)<br>*packets* — количество пакетов<br>*errors* — количество ошибок<br>*dropped* — количество отброшенных пакетов<br>*overruns (fifo)* — количество ошибок FIFO буфера<br>*frame* — количество ошибок кадрирования пакетов<br>*compressed* — количество сжатых пакетов, полученных драйвером устройства<br>*multicast* — количество кадров многоадресного вещания, которые получены драйвером устройства

Комментарии:

- Вы можете использовать этот ключ с шагом предобработки *Изменение в секунду* для получения статистики байт в секунду;
- Режим *dropped* поддерживается только в Linux, FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD;
- Режимы *overruns*, *frame*, *compressed*, *multicast* поддерживаются только в Linux;
- На HP-UX этот элемент данных не предоставляет сведений о loopback-интерфейсах (например, lo0).

Примеры:

```
net.if.in[eth0]
net.if.in[eth0,errors]
```

net.if.out[if,<режим>]

<br> Статистика по исходящему трафику сетевого интерфейса.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. На NetBSD требуются права суперпользователя (root).

Параметры:

- **if** — имя сетевого интерфейса (Unix); полное описание сетевого интерфейса или IPv4 адрес; или, если в фигурных скобках, GUID сетевого интерфейса (Windows);
- **режим** — возможные значения:<br>*bytes* — количество байт (по умолчанию)<br>*packets* — количество пакетов<br>*errors* — количество ошибок<br>*dropped* — количество отброшенных пакетов<br>*overruns (fifo)* — количество ошибок FIFO буфера<br>*collisions (colls)* — количество коллизий, обнаруженных на интерфейсе<br>*carrier* — количество потерь несущей, обнаруженных драйвером устройства<br>*compressed* — количество сжатых пакетов, переданных драйвером устройства

Комментарии:

- Вы можете использовать этот ключ с шагом предобработки *Изменение в секунду* для получения статистики байт в секунду;
- Режим *dropped* поддерживается только в Linux, HP-UX;
- Режимы *overruns*, *collision*, *carrier*, *compressed* поддерживаются только в Linux;
- На HP-UX этот элемент данных не предоставляет сведений о loopback-интерфейсах (например, lo0).

Примеры:

```
net.if.out[eth0]
net.if.out[eth0,errors]
```

net.if.total[if,<режим>]

<br> Суммарная статистика входящего и исходящего трафика на сетевом интерфейсе.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. На NetBSD требуются права суперпользователя (root).

Параметры:

- **if** — имя сетевого интерфейса (Unix); полное описание сетевого интерфейса или IPv4 адрес; или, если в фигурных скобках, GUID сетевого интерфейса (Windows);
- **режим** — возможные значения:<br>*bytes* — количество байт (по умолчанию)<br>*packets* — количество пакетов<br>*errors* — количество ошибок<br>*dropped* — количество отброшенных пакетов<br>*overruns (fifo)* — количество ошибок FIFO буфера<br>*collisions (colls)* — количество коллизий, обнаруженных на интерфейсе<br>*compressed* — количество сжатых пакетов, переданных или полученных драйвером устройства

Комментарии:

- Вы можете использовать этот ключ с шагом предобработки *Изменение в секунду* для получения статистики байт в секунду;
- Режим *dropped* поддерживается только в Linux, HP-UX. Отброшенные пакеты поддерживаются, только если оба ключа `net.if.in` и `net.if.out` работают с отброшенными пакетами на вашей платформе.
- Режимы *overruns*, *collision*, *compressed* поддерживаются только в Linux;
- На HP-UX этот элемент данных не предоставляет сведений о loopback-интерфейсах (например, lo0).

Примеры:

```
net.if.total[eth0]
net.if.total[eth0,errors]
```

net.tcp.listen[порт]

<br> Проверка, находится ли TCP порт в состоянии LISTEN.<br> Возвращаемые значения: 0 — не находится в состоянии LISTEN; 1 — находится в состоянии LISTEN.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, MacOS X, Windows.

Параметр:

- **порт** — номер TCP порта

На ядрах Linux 2.6.14 и более новых информация о TCP сокетах в состоянии listen берётся из интерфейса ядра NETLINK, если это возможно. В противном случае информация берётся из файлов /proc/net/tcp и /proc/net/tcp6.

Пример:

```
net.tcp.listen[80]
```

net.tcp.port[<ip>,порт]

<br> Проверка возможности создания TCP подключения на указанный номер порта.<br> Возвращаемые значения: 0 — не удалось подключиться; 1 — удалось подключиться.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **ip** — IP адрес или DNS имя (по умолчанию 127.0.0.1);
- **порт** — номер порта.

Комментарии:

- Для простого тестирования производительности TCP используйте `net.tcp.service.perf[tcp,<ip>,<  >]`;
- Эти проверки могут привести к дополнительным записям в системных файлах журналов (обычно журналируются сессии SMTP и SSH).

Пример:

```
net.tcp.port[,80] #                          -    ,          80
```

net.tcp.service[сервис,<ip>,<порт>]

<br> Проверка того, что сервис запущен и принимает TCP подключения.<br> Возвращаемые значения: 0 — сервис недоступен; 1 — сервис запущен.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **сервис** — один из: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* или *telnet* (смотрите подробности);
- **ip** — IP адрес или DNS имя (по умолчанию 127.0.0.1);
- **порт** — номер порта (по умолчанию используется стандартный номер порта сервиса).

Комментарии:

- Эти проверки могут привести к дополнительным записям в системных файлах журналов (обычно журналируются сессии SMTP и SSH);
- Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. В качестве обходного решения, пожалуйста, для подобных проверок используйте `net.tcp.port[]`.
- Проверка LDAP и HTTPS на платформе Windows поддерживается только Zabbix агентом 2;
- Проверка telnet ищет запрос на вход (с «:» в конце).

Пример:

```
net.tcp.service[ftp,,45] #                              FTP        TCP 45    .
```

net.tcp.service.perf[сервис,<ip>,<порт>]

<br> Проверка производительности TCP сервиса.<br> Возвращаемые значения: *Число с плавающей точкой* (0 — сервис недоступен; секунды — количество секунд, потраченное на подключение к сервису).<br> Смотрите Поддерживаемые платформы.

Параметры:

- **сервис** — один из: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* или *telnet* (смотрите подробности);
- **ip** — IP адрес или DNS имя (по умолчанию 127.0.0.1);
- **порт** — номер порта (по умолчанию используется стандартный номер порта сервиса).

Комментарии:

- Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. В качестве обходного решения, пожалуйста, для подобных проверок используйте `net.tcp.service.perf[`
- Проверка telnet ищет запрос на вход (с «:» в конце).

Пример:

```
net.tcp.service.perf[ssh] #                                    SSH
```

net.tcp.socket.count[<локальный_адрес>,<локальный_порт>,<удалённый_адрес>,<удалённый_порт>,<состояние>]

<br> Получение количества TCP сокетов, соответствующих заданным параметрам.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, Windows.

Параметры:

- **локальный_адрес** — локальный IPv4/6 адрес или CIDR подсеть;
- **локальный_порт** — номер локального порта или имя сервиса;
- **удалённый_адрес** — удалённый IPv4/6 адрес или CIDR подсеть;
- **удалённый_порт** — номер удалённого порта или имя сервиса;
- **состояние** — состояние соединения (*established*, *syn_sent*, *syn_recv*, *fin_wait1*, *fin_wait2*, *time_wait*, *close*, *close_wait*, *last_ack*, *listen*, *closing*).

Пример:

```
net.tcp.socket.count[,80,,,established] #                  TCP     80         «established»
```

net.udp.listen[порт]

<br> Проверка, находится ли UDP порт в состоянии LISTEN.<br> Возвращаемые значения: 0 — не находится в состоянии LISTEN; 1 — находится в состоянии LISTEN.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, MacOS X.

Параметр:

- **порт** — номер UDP порта

Пример:

```
net.udp.listen[68]
```

net.udp.service[сервис,<ip>,<порт>]

<br> Проверка, запущен ли сервис и отвечает ли он на UDP запросы.<br> Возвращаемые значения: 0 — сервис недоступен; 1 — сервис запущен.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **сервис** — *ntp* (смотрите подробности);
- **ip** — IP адрес или DNS имя (по умолчанию 127.0.0.1);
- **порт** — номер порта (по умолчанию используется стандартный номер порта сервиса).

Пример:

```
net.udp.service[ntp,,45] #                                NTP         45     UDP
```

net.udp.service.perf[сервис,<ip>,<порт>]

<br> Проверка производительности UDP сервиса.<br> Возвращаемые значения: *Число с плавающей точкой* (0 — сервис недоступен; секунды — количество секунд, потраченное на ожидание ответа от сервиса).<br> Смотрите Поддерживаемые платформы.

Параметры:

- **сервис** — *ntp* (смотрите подробности);
- **ip** — IP адрес или DNS имя (по умолчанию 127.0.0.1);

- **порт** — номер порта (по умолчанию используется стандартный номер порта сервиса).

Пример:

```
net.udp.service.perf[ntp] #                                    NTP
```

net.udp.socket.count[<локальный_адрес>,<локальный_порт>,<удалённый_адрес>,<удалённый_порт>,<состояние>]

<br> Получение количества UDP сокетов, соответствующих заданным параметрам.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, Windows.

Параметры:

- **локальный_адрес** — локальный IPv4/6 адрес или CIDR подсеть;
- **локальный_порт** — номер локального порта или имя сервиса;
- **удалённый_адрес** — удалённый IPv4/6 адрес или CIDR подсеть;
- **удалённый_порт** — номер удалённого порта или имя сервиса;
- **состояние** — состояние соединения (*established*, *unconn*).

Пример:

```
net.udp.socket.count[,,,,established] #             UDP             «established»
```

proc.cpu.util[<имя>,<пользователь>,<тип>,<cmdline>,<режим>,<зона>]

<br> Использование процессом CPU в процентах.<br> Возвращаемое значение: *Число с плавающей точкой*.<br> Поддерживаемые платформы: Linux, Solaris[6].

Параметры:

- **имя** — имя процесса (по умолчанию *все процессы*);
- **пользователь** — имя пользователя (по умолчанию *все пользователи)*);
- **тип** — тип использования CPU: *total* (по умолчанию), *user* или *system*;
- **cmdline** — фильтр по командной строке (является регулярным выражением);
- **режим** — режим сбора данных: *avg1* (по умолчанию), *avg5* или *avg15*;
- **зона** — целевая зона: *current* (по умолчанию) или *all*. Этот параметр поддерживается только на платформе Solaris.

Комментарии:

- Возвращаемое значение основывается на использовании одного ядра CPU, в процентах. Например, утилизация CPU процесса, полностью использующего два ядра, равна 200%.
- Данные использования CPU процессами собираются коллектором, который поддерживает максимально до 1024 уникальных (по имени, пользователю и командной строке) запросов. Запросы, к которым не было обращения в течение последних 24 часов, удаляются из коллектора.
- В случае, когда агент скомпилирован на Solaris без поддержки зон, если задать параметр значением *current* (или по умолчанию) и запустить его на более новом Solaris, где зоны поддерживаются, то агент вернёт NOTSUPPORTED (агент не может ограничить результаты в пределах только текущей зоны). Однако, значение *all* в этом случае поддерживается.

Примеры:

```
proc.cpu.util[,root] #        CPU           ,                  «root»
proc.cpu.util[zabbix_server,zabbix] #        CPU           zabbix_server,                  zabbix
```

proc.get[<name>,<user>,<cmdline>,<mode>]

<br> Список процессов ОС и их параметров. Может использоваться для низкоуровневого обнаружения.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux, FreeBSD, Windows, OpenBSD, NetBSD.

Параметры:

- **name** - имя процесса (по умолчанию *all processes*);
- **user** - имя пользователя (по умолчанию *all users*);
- **cmdline** - фильтрация по командной строке (это регулярное выражение). Этот параметр не поддерживается в Windows; на других платформах он не поддерживается, если для mode задано значение 'summary'.
- **mode** - возможные значения:<br>*process* (по умолчанию), *thread* (не поддерживается в NetBSD), *summary*. См. список параметров процесса, возвращаемых для каждого режима и ОС.

Комментарии:

- Если значение не удается получить, например из-за ошибки (процесс уже завершился, недостаточно прав, сбой системного вызова), будет возвращено −1.
- См. примечания по выбору процессов с параметрами `name` и `cmdline` (только для Linux).

Примеры:

```
proc.get[zabbix_server,zabbix,,process] #             zabbix_server,                zabbix,
proc.get[java,,,thread] #             Java,
proc.get[,zabbix,,summary] #                         ,                zabbix,
```

proc.mem[<имя>,<пользователь>,<режим>,<cmdline>,<тип памяти>]

<br> Количество памяти, используемое процессом, в байтах.<br> Возвращаемое значение: *Целое число* — с , равным *max*, *min*, *sum*; *Число с плавающей точкой* — с , равным *avg*<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

Параметры:

- **имя** — имя процесса (по умолчанию *все процессы*);
- **пользователь** — имя пользователя (по умолчанию *все пользователи*);
- **режим** — возможные значения: *avg*, *max*, *min* или *sum* (по умолчанию);
- **cmdline** — фильтр по командной строке (является регулярным выражением);
- **тип памяти** — тип памяти, используемый процессом

Комментарии:

- Параметр поддерживается только на Linux, FreeBSD, Solaris[6], AIX;
- Когда несколько процессов используют разделяемую память, сумма используемой процессами памяти может дать в результате большие, нереальные значения.<br><br>Смотрите примечания по выбору процессов с параметрами и cmdline (специфично для Linux).<br><br>Когда этот элемент данных вызывается из командной строки и содержит параметр командной строки (например, при использовании тестового режима агента: zabbix_agentd -t proc.mem[,,,apache2]), будет засчитан один дополнительный процесс, так как агент посчитает самого себя.

Примеры:

```
proc.mem[,root] #    ,                  ,                  «root»
proc.mem[zabbix_server,zabbix] #    ,                  zabbix_server,                  zabbix
proc.mem[,oracle,max,oracleZABBIX] #    ,                              ,                  oracle       orac
```

proc.num[<имя>,<пользователь>,<состояние>,<cmdline>,<зона>]

<br> Количество процессов.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris[6], HP-UX, AIX, OpenBSD, NetBSD, Windows.

Параметры:

- **имя** — имя процесса (по умолчанию *все процессы*);
- **пользователь** — имя пользователя (по умолчанию *все пользователи*);
- **состояние** — возможные значения:<br>*all* (по умолчанию),<br>*disk* — непрерываемый сон,<br>*run* — в процессе работы,<br>*sleep* — прерываемый сон,<br>*trace* — остановлен,<br>*zomb* — зомби;
- **cmdline** — фильтр по командной строке (является регулярным выражением);
- **зона** — целевая зона: *current* (по умолчанию) или *all*. Этот параметр поддерживается только на платформе Solaris.

Комментарии:

- Параметры состояния *disk* и *trace* поддерживаются только в Linux, FreeBSD, OpenBSD, NetBSD;
- В системах Windows поддерживаются только параметры and ;
- Когда этот элемент данных вызывается из командной строки и содержит параметр командной строки (например, при использовании тестового режима агента: zabbix_agentd -t proc.num[,,,apache2]), будет засчитан один дополнительный процесс, так как агент посчитает самого себя;
- В случае, когда агент скомпилирован на Solaris без поддержки зон, если задать параметр значением *current* (или по умолчанию) и запустить его на более новом Solaris, где зоны поддерживаются, тогда агент вернёт NOTSUPPORTED (агент не может ограничить результаты в пределах только текущей зоны). Однако, значение *all* в этом случае поддерживается.
- Смотрите примечания по выбору процессов с параметрами и cmdline (специфично для Linux).

Примеры:

```
proc.num[,mysql] #           ,                  mysql
proc.num[apache2,www-data] #           apache2,                  www-data
proc.num[,oracle,sleep,oracleZABBIX] #                         ,                  oracle       oracleZABBIX
```

sensor[device,sensor,<mode>]

<br> Показания аппаратного датчика.<br> Возвращаемое значение: *Float*.<br> Поддерживаемые платформы: Linux, OpenBSD.

Параметры:

- **device** - имя устройства;
- **sensor** - имя датчика;
- **mode** - возможные значения: *avg*, *max* или *min*.

Комментарии:

- Читает /proc/sys/dev/sensors в Linux 2.4.
- Читает /sys/class/hwmon в Linux 2.6+. См. более подробное описание элемента данных sensor в Linux.
- Читает MIB *hw.sensors* в OpenBSD.
- В Linux 2.4, если указан mode, то device и sensor рассматриваются как регулярные выражения.
- В Linux 2.6+ и OpenBSD, либо если mode опущен, device и sensor рассматриваются как буквальные значения.

Примеры:

```
sensor[w83781d-i2c-0-2d,temp1]
sensor[cpu0,temp0] #            CPU
sensor["cpu[0-2]$",temp,avg] #                    CPU
```

system.boottime

<br> Штамп времени (timestamp) загрузки системы.<br> Возвращаемое значение: *Целое число (Unix timestamp)*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

system.cpu.discovery

<br> Список обнаруженных CPU/ядер CPU. Используется для обнаружения на низком уровне.<br> Возвращаемое значение: *строка JSON*.<br> См. поддерживаемые платформы.

system.cpu.intr

<br> Прерывания устройств.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.load[<cpu>,<режим>]

<br> Загрузка CPU.<br> Возвращаемое значение: *Число с плавающей точкой*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **cpu** — возможные значения: *all* (по умолчанию) или *percpu* (общая загрузка делится на количество CPU онлайн);
- **режим** — возможные значения: *avg1* (усреднение за одну минуту, по умолчанию), *avg5* или *avg15*.

Пример:

```
system.cpu.load[,avg5]
```

system.cpu.num[<тип>]

<br> Количество CPU.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Параметр:

- **тип** — возможные значения: *online* (по умолчанию) или *max*

Значение *max* параметра «тип» поддерживается только в Linux, FreeBSD, Solaris, MacOS X.

Пример:

```
system.cpu.num
```

system.cpu.switches

<br> Количество переключений контекста.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.util[<cpu>,<type>,<mode>,<logical or physical>]

<br> Процент использования CPU.<br> Возвращаемое значение: *Float*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

Параметры:

- **cpu** - *<номер CPU>* или *all* (по умолчанию);

- **type** - возможные значения: *user* (по умолчанию), *idle*, *nice*, *system*, *iowait*, *interrupt*, *softirq*, *steal*, *spin* (в OpenBSD), *guest* (в ядрах Linux 2.6.24 и новее) или *guest_nice* (в ядрах Linux 2.6.33 и новее);
- **mode** - возможные значения: *avg1* (среднее за одну минуту, по умолчанию), *avg5* или *avg15*;
- **logical or physical** - возможные значения: *logical* (по умолчанию) или *physical*. Этот параметр поддерживается только в AIX.

Комментарии:

- Параметр типа *nice* поддерживается только в Linux, FreeBSD, HP-UX, OpenBSD, NetBSD.
- Параметр типа *iowait* поддерживается только в Linux 2.6 и новее, Solaris, AIX.
- Параметр типа *interrupt* поддерживается только в Linux 2.6 и новее, FreeBSD, OpenBSD.
- Параметры типа *softirq*, *steal*, *guest*, *guest_nice* поддерживаются только в Linux 2.6 и новее.
- Параметры режима *avg5* и *avg15* поддерживаются в Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD.

Пример:

```
system.cpu.util[0,user,avg5]
```

system.hostname[<тип>, <преобразование>]

<br> Системное имя хоста.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **тип** — возможные значения: *netbios* (по умолчанию на Windows), *host* (по умолчанию на Linux), *shorthost* (возвращает часть имени хоста до первой точки, всю строку для имён без точек), *fqdn* (возвращает полное доменное имя);
- **преобразование** — возможные значения: *none* (по умолчанию) или *lower* (конвертация в нижний регистр).

Значение получается путём извлечения значения `nodename` из вывода системного API uname().

Примеры возвращаемых значений:

```
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[<информация>]

<br> Информация о шасси.<br> Возвращаемое значение: *Строка*.<br> Поддерживаемые платформы: Linux.

Параметр:

- **информация** — возможные значения: *full* (по умолчанию), *model*, *serial*, *type* или *vendor*

Комментарии:

- Этот ключ элемента данных зависит от доступности таблицы SMBIOS;
- Будет выполнена попытка чтения таблицы DMI из sysfs; если доступ к sysfs будет неудачным, тогда будет попытка чтения напрямую из памяти;
- Требуются **права root**, потому что значение читается из sysfs или памяти.

Пример:

```
system.hw.chassis[full] → Hewlett-Packard HP Pro 3010 Small Form Factor PC CZXXXXXXXX Desktop
```

system.hw.cpu[<cpu>,<информация>]

<br> Информация о CPU.<br> Возвращаемое значение: *Строка* или *Целое число*.<br> Поддерживаемые платформы: Linux.

Параметры:

- **cpu** — <*номер CPU*> или *all* (по умолчанию);
- **информация** — возможные значения: *full* (по умолчанию), *curfreq*, *maxfreq*, *model* или *vendor*.

Комментарии:

- Информация собирается из `/proc/cpuinfo` и из `/sys/devices/system/cpu/[cpunum]/cpufreq/cpuinfo_max_freq`;
- Если указаны номер CPU и *curfreq* или *maxfreq*, то возвращается числовое значение (Гц).

Пример:

```
system.hw.cpu[0,vendor] → AuthenticAMD
```

system.hw.devices[<тип>]

<br> Список PCI или USB устройств.<br> Возвращаемое значение: *Текст*.<br> Поддерживаемые платформы: Linux.

Параметр:

- **тип** — *pci* (по умолчанию) или *usb*

Возвращает вывод утилит lspci или lsusb (выполняются без каких либо параметров).

Пример:

```
system.hw.devices → 00:00.0 Host bridge: Advanced Micro Devices [AMD] RS780 Host Bridge
```

system.hw.macaddr[<интерфейс>,<формат>]

<br> Список MAC адресов.<br> Возвращаемое значение: *Строка*.<br> Поддерживаемые платформы: Linux.

Параметры:

- **интерфейс** — *all* (по умолчанию) или регулярное выражение;
- **формат** — *full* (по умолчанию) или *short*

Комментарии:

- Список MAC адресов интерфейсов, чьи имена соответствуют заданному в параметре регулярному выражению (*all* перечисляет все интерфейсы);
- Если задан значением *short*, имена интерфейсов и одинаковые MAC адреса не попадут в список.

Пример:

```
system.hw.macaddr["eth0$",full] → [eth0] 00:11:22:33:44:55
```

system.localtime[<тип>]

<br> Системное время.<br> Возвращаемое значение: *Целое число* — с как *utc*; *String* — с как *local*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **тип** — возможные значения: *utc* — (по умолчанию) время с начала Эпохи (00:00:00 UTC, Январь 1, 1970), измеренное в секундах, или же *local* — время в формате 'гггг-мм-дд,чч:мм:сс.ннн,+чч:мм'

С Zabbix агентом необходимо использовать только как пассивную проверку; с Zabbix агентом 2 можно настраивать и как активную проверку.

Пример:

```
system.localtime[local] #              ,              ,                                                                    *  *.
```

system.run[команда,<режим>]

<br> Выполнение указанной команды на узле сети.<br> Возвращаемое значение: *Текст* с результатом выполнения команды, либо 1 — с со значением *nowait* (независимо от результата выполнения команды).<br> Смотрите Поддерживаемые платформы.

Параметры:

- **команда** — выполняемая команда;<br>
- **режим** — возможные значения: *wait* — ожидать завершения выполнения (по умолчанию) или *nowait* — не ожидать.

Комментарии:

- Этот элемент данных по умолчанию отключён. Узнайте, как их включить;
- Возвращаемое значение элемента данных является стандартным выводом совместно со стандартным выводом ошибок, которые получены от команды. Код завершения не проверяется;
- Для корректной обработки возвращаемое командой значение должно иметь тип данных . Пустой результат также допустим;
- Возвращаемое значение ограничено 16 МБ (включая конечные пробелы, которые усекаются); также применяются ограничения баз данных;
- Смотрите также: Выполнение команд.

Пример:

```
system.run[ls -l /] #
```

system.stat[ресурс,<тип>]

<br> Системная статистика.<br> Возвращаемое значение: *Целое число* или *число с плавающей точкой*.<br> Поддерживаемые платформы: AIX.

Параметры:

- **ent** — количество процессорных единиц, выделенных на ресурс (число с плавающей точкой);
- **kthr,<тип>** — информация о состояниях потоков ядра:<br>*r* — среднее количество запускаемых потоков ядра (число с плавающей точкой)<br>*b* — среднее количество потоков ядра, помещённых в очередь ожидания Менеджера Виртуальный Памяти (число с плавающей точкой)
- **memory,<тип>** — информация об использовании виртуальной и реальной памяти:<br>*avm* — активные виртуальные страницы (целое число)<br>*fre* — размер свободного списка (целое число)
- **page,<тип>** — информация об ошибках страниц и активности страниц:<br>*fi* — количество загрузок страниц в секунду (число с плавающей точкой)<br>*fo* — количество выгрузок страниц в секунду (число с плавающей точкой)<br>*pi* — страницы, загруженные из пространства подкачки (число с плавающей точкой)<br>*po* — страницы, выгруженные в пространство подкачки (число с плавающей точкой)<br>*fr* — освобождённые страницы (замена страниц) (число с плавающей точкой)<br>*sr* — количество страниц, сканированных алгоритмом замены страниц (число с плавающей точкой)
- **faults,<тип>** — соотношение ловушек и прерываний:<br>*in* — прерывания устройства (число с плавающей точкой)<br>*sy* — системные вызовы (число с плавающей точкой)<br>*cs* — переключения контекста потока ядра (число с плавающей точкой)
- **cpu,<тип>** — распределение процессорного времени в процентах:<br>*us* — пользовательское время (число с плавающей точкой)<br>*sy* — системное время (число с плавающей точкой)<br>*id* — время простоя (число с плавающей точкой)<br>*wa* — время простоя, в ходе которого система обрабатывала запросы I/O диска/NFS (число с плавающей точкой)<br>*pc* — количество использованных физических процессоров (число с плавающей точкой)<br>*ec* — процент потреблённой выделенной ёмкости (число с плавающей точкой)<br>*lbusy* — отображает утилизацию в процентах для логических процессор(ов), которая произошла при выполнении уровней пользователя и системы (число с плавающей точкой)<br>*app* — отображает количество доступных физических процессоров в общем пуле (число с плавающей точкой)
- **disk,<тип>** — статистика о дисках:<br>*bps* — показывает объём данных, переданных (прочитанных или записанных) на диск в байтах в секунду (целое число)<br>*tps* — показывает количество передач за секунду, которые произошли на физическом диске/ленте (число с плавающей точкой)

Комментарии:

- Обратите внимание на следующие ограничения в этих элементах данных:<br> `system.stat[cpu,app]` — поддерживается только на AIX LPAR с типом «Shared»<br> `system.stat[cpu,ec]` — поддерживается на AIX LPAR с типом «Shared» и «Dedicated» («Dedicated» всегда возвращает 100 (процентов))<br> `system.stat[cpu,lbusy]` — поддерживается только на AIX LPAR с типом «Shared»<br> `system.stat[cpu,pc]` — поддерживается на AIX LPAR с типом «Shared» и «Dedicated»<br> `system.stat[ent]` — поддерживается на AIX LPAR с типом «Shared» и «Dedicated»

system.sw.arch

<br> Информация об архитектуре программного обеспечения.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Для получения информации используется функция `uname()`.

Пример:

```
system.sw.arch → i686
```

system.sw.os[<информация>]

<br> Информация об операционной системе.<br> Возвращаемое значение: *Строка*.<br> Поддерживаемые платформы: Linux, Windows.

Параметр:

- **информация** — возможные значения: *full* (по умолчанию), *short* или *name*

Информация извлекается из (обратите внимание, что не все файлы и опции представлены во всех дистрибутивах):

- `/proc/version` (*full*) в Linux;
- `/proc/version_signature` (*short*) в Linux;

- Параметр PRETTY_NAME из `/etc/os-release` на Linux-системах, которые его поддерживают, или `/etc/issue.net` (*name*);
- ключ реестра `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion` в Windows.

Примеры:

```
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

<br> Подробная информация об операционной системе (версия, тип, название дистрибутива, младшая и старшая версии и т. д).<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux, Windows.

system.sw.packages[<регулярное выражение>,<менеджер>,<формат>]

<br> Список установленных пакетов.<br> Возвращаемое значение: *Текст*.<br> Поддерживаемые платформы: Linux.

Параметры:

- **регулярное выражение** — *all* (по умолчанию) или регулярное выражение;
- **менеджер** — *all* (по умолчанию) или конкретный менеджер пакетов;
- **формат** — *full* (по умолчанию) или *short*.

Комментарии:

- Список (в алфавитном порядке) установленных пакетов, имена которых соответствуют заданному регулярному выражению (*all* перечисляет их все);
- Поддерживаемые менеджеры пакетов (выполняемые команды):<br>dpkg (dpkg --get-selections)<br>pkgtool (ls /var/log/packages)<br>rpm (rpm -qa)<br>pacman (pacman -Q)<br>portage
- Если указан значением *full*, пакеты группируются по менеджерам пакетов (каждый менеджер с отдельной строки, которая начинается с его имени, заключённого в квадратные скобки);
- Если указан значением *short*, пакеты не группируются и отображаются одной строкой.

Примеры:

```
system.sw.packages[mini,dpkg,short] → python-minimal, python2.6-minimal, ubuntu-minimal
```

system.sw.packages.get[<regexp>,<manager>]

<br> Подробный список установленных пакетов.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux.

Параметры:

- **regexp** - *all* (по умолчанию) или регулярное выражение;
- **manager** - *all* (по умолчанию) или менеджер пакетов (возможные значения: *rpm*, *dpkg*, *pkgtool*, *pacman* или *portage*).

Комментарии:

- Возвращает неформатированный JSON с установленными пакетами, имя которых соответствует заданному регулярному выражению.
- Вывод представляет собой массив объектов, каждый из которых содержит следующие ключи: name, manager, version, size, architecture, buildtime и installtime (см. подробности).

system.swap.in[<устройство>,<тип>]

<br> Статистика файла подкачки в (с устройства в память).<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, OpenBSD.

Parameters:

- **устройство** — указать устройство, используемое для файла подкачки (только Linux) или *all* (по умолчанию);
- **тип** — возможные значения: *count* (количество подкачек, по умолчанию на платформах, отличных от Linux), *sectors* (подкачанные сектора) или *pages* (подкачанные страницы, по умолчанию в Linux).

Комментарии:

- Источником этой информации является:<br>/proc/swaps, /proc/partitions, /proc/stat (Linux 2.4)<br>/proc/swaps, /proc/diskstats, /proc/vmstat (Linux 2.6)
- Обратите внимание, что *pages* будет работать, только если устройство не указано;
- Параметр типа *sectors* поддерживается только в Linux.

Пример:

```
system.swap.in[,pages]
```

system.swap.out[<устройство>,<тип>]

<br> Статистика файла подкачки из (из памяти на устройство).<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, OpenBSD.

Параметры:

- **устройство** — указать устройство, используемое для файла подкачки (только Linux) или *all* (по умолчанию);
- **тип** — возможные значения: *count* (количество выгрузок, по умолчанию на платформах, отличных от Linux), *sectors* (выгруженные сектора) или *pages* (выгруженные страницы, по умолчанию в Linux).

Комментарии:

- Источником этой информации является:<br>`/proc/swaps`, `/proc/partitions`, `/proc/stat` (Linux 2.4)<br>`/proc/swaps`, `/proc/diskstats`, `/proc/vmstat` (Linux 2.6)
- Обратите внимание, что *pages* будет работать, только если устройство не указано;
- Параметр типа *sectors* поддерживается только в Linux.

Пример:

```
system.swap.out[,pages]
```

system.swap.size[<устройство>,<тип>]

<br> Размер файла подкачки в байтах или процентах от общего размера.<br> Возвращаемое значение: *Целое число* — для количества байт; *Число с плавающей точкой* — для процентов.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, OpenBSD, Windows.

Параметры:

- **устройство** — устройство, используемое для файла подкачки (только FreeBSD) или *all* (по умолчанию);
- **тип** — возможные значения: *free* (свободно в файле подкачки, по умолчанию), *pfree* (свободно в файле подкачки, в процентах), *pused* (используемое место в файле подкачки, в процентах), *total* (полный размер файла подкачки) или *used* (используется места в файле подкачки).

Комментарии:

- Обратите внимание, что *pfree*, *pused* не поддерживаются в Windows, если размер подкачки равен 0;
- Если устройство не указано, Zabbix агент будет брать во внимание только swap устройства (файлы), физическая память будет игнорироваться. Например, на Solaris системах команда `swap -s` включает часть физической памяти и swap устройства (в отличие от `swap -l`).

Пример:

```
system.swap.size[,pfree] →
```

system.uname

<br> Идентификация системы.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Комментарии:

- В Unix значение этого элемента данных извлекается при помощи системного вызова uname();
- Этот элемент данных в Windows возвращает архитектуру ОС, тогда как в Unix он возвращает архитектуру CPU.

Примеры:

```
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

<br> Время работы в секундах.<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Для получения читабельных значений используйте при настройке элемента данных единицы измерения **s** или **uptime**.

system.users.num

<br> Количество пользователей, находящихся в системе.<br> Возвращаемое значение: *Целое число*.<br> Смотрите Поддерживаемые платформы.

Для получения значений используется команда **who** на стороне агента.

vfs.dev.discovery

<br> Список блочных устройств и их тип. Используется для обнаружения на низком уровне.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux.

vfs.dev.read[<устройство>,<тип>,<режим>]

<br> Статистика чтения диска.<br> Возвращаемое значение: *Целое число* — с , равным *sectors*, *operations*, *bytes*; *Число с плавающей точкой* — с , равным *sps*, *ops*, *bps*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Параметры:

- **устройство** — дисковое устройство (по умолчанию *all* [3]);
- **тип** — возможные значения: *sectors*, *operations*, *bytes*, *sps*, *ops* или *bps* (*sps*, *ops*, *bps* соответствуют: секторам, операциям, байтам в секунду соответственно);
- **режим** — возможные значения: *avg1* (усреднение за минуту, по умолчанию), *avg5* или *avg15*. Этот параметр поддерживается, только если один из: sps, ops, bps.

Комментарии:

- При использовании интервала обновления три часа или более[2] этот элемент данных всегда будет возвращать «0»;
- Параметры типа *sectors* и *sps* поддерживаются только в Linux;
- Параметр типа *ops* поддерживается только в Linux и FreeBSD;
- Параметр типа *bps* поддерживается только в FreeBSD;
- Параметр типа *bytes* поддерживается только в FreeBSD, Solaris, AIX, OpenBSD;
- Параметр поддерживается только в Linux, FreeBSD;
- Можно использовать относительные имена устройств (например, sda), а также необязательный префикс /dev/ (например, /dev/sda)
- Поддерживаются логические тома LVM;
- Значения по умолчанию параметра «тип» для различных ОС:<br>*AIX* — operations<br>*FreeBSD* — bps<br>*Linux* — sps<br>*OpenBSD* — operations<br>*Solaris* — bytes
- *sps*, *ops* и *bps* на поддерживаемых платформах ограничены 1024 устройствами (1023 индивидуальных и одно для *all*).

Пример:

```
vfs.dev.read[,operations]
```

vfs.dev.write[<устройство>,<тип>,<режим>]

<br> Статистика записи на диск.<br> Возвращаемое значение: *Целое число* — при тип равном *sectors*, *operations*, *bytes*; *Число с плавающей точкой* — при тип равном *sps*, *ops*, *bps*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Параметры:

- **устройство** — дисковое устройство (по умолчанию *all* [3]);
- **тип** — возможные значения: *sectors*, *operations*, *bytes*, *sps*, *ops* или *bps* (*sps*, *ops*, *bps* соответствуют: секторам, операциям, байтам в секунду соответственно);
- **режим** — возможные значения: *avg1* (усреднение за минуту, по умолчанию), *avg5* или *avg15*. Этот параметр поддерживается, только если один из: sps, ops, bps.

Комментарии:

- При использовании интервала обновления три часа или более[2] этот элемент данных всегда будет возвращать «0»;
- Параметры типа *sectors* и *sps* поддерживаются только в Linux;
- Параметр типа *ops* поддерживается только в Linux и FreeBSD;
- Параметр типа *bps* поддерживается только в FreeBSD;
- Параметр типа *bytes* поддерживается только в FreeBSD, Solaris, AIX, OpenBSD;
- Параметр поддерживается только в Linux, FreeBSD;
- Можно использовать относительные имена устройств (например, sda), а также необязательный префикс /dev/ (например, /dev/sda);
- Поддерживаются логические тома LVM;
- Значения по умолчанию параметра «тип» для различных ОС:<br>*AIX* — operations<br>*FreeBSD* — bps<br>*Linux* — sps<br>*OpenBSD* — operations<br>*Solaris* — bytes
- *sps*, *ops* и *bps* на поддерживаемых платформах ограничены 1024 устройствами (1023 индивидуальных и одно для *all*).

Пример:

```
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>]

<br> Количество записей в каталоге.<br> Возвращаемое значение: *Integer*.<br> См. поддерживаемые платформы.

Параметры:

- **dir** - абсолютный путь к каталогу;
- **regex incl** - регулярное выражение, описывающее шаблон имени объекта (файл, каталог, символическая ссылка), который нужно включить; если пусто, включаются все (значение по умолчанию);
- **regex excl** - регулярное выражение, описывающее шаблон имени объекта (файл, каталог, символическая ссылка), который нужно исключить; если пусто, ничего не исключается (значение по умолчанию);
- **types incl** - типы записей каталога, которые нужно учитывать, возможные значения: *file* - обычный файл, *dir* - подкаталог, *sym* - символическая ссылка, *sock* - сокет, *bdev* - блочное устройство, *cdev* - символьное устройство, *fifo* - FIFO, *dev* - синоним для "bdev,cdev", *all* - все типы (по умолчанию), то есть "file,dir,sym,sock,bdev,cdev,fifo". Несколько типов должны быть разделены запятыми и заключены в кавычки.
- **types excl** - типы записей каталога (см. `types incl`), которые НЕ нужно учитывать. Если один и тот же тип записи указан и в `types incl`, и в `types excl`, записи этого типа НЕ учитываются.
- **max depth** - максимальная глубина обхода подкаталогов:<br>**-1** (по умолчанию) - без ограничений,<br>**0** - без перехода в подкаталоги.
- **min size** - минимальный размер (в байтах) файла, который будет учитываться. Файлы меньшего размера учитываться не будут. Можно использовать суффиксы памяти.
- **max size** - максимальный размер (в байтах) файла, который будет учитываться. Файлы большего размера учитываться не будут. Можно использовать суффиксы памяти.
- **min age** - минимальный возраст (в секундах) записи каталога, которая будет учитываться. Более новые записи учитываться не будут. Можно использовать суффиксы времени.
- **max age** - максимальный возраст (в секундах) записи каталога, которая будет учитываться. Записи такой давности и старше учитываться не будут (время изменения). Можно использовать суффиксы времени.
- **regex excl dir** - регулярное выражение, описывающее шаблон имени каталога, который нужно исключить. Все содержимое каталога будет исключено (в отличие от regex_excl)

Комментарии:

- В Windows этот элемент данных следует по UNC-путям, что может представлять риск безопасности.
- Переменные среды, например %APP_HOME%, $HOME и %TEMP%, не поддерживаются.
- Псевдокаталоги "." и ".." никогда не учитываются.
- Символические ссылки никогда не используются при обходе каталогов.
- И `regex incl`, и `regex excl` применяются к файлам и каталогам при вычислении количества записей, но игнорируются при выборе подкаталогов для обхода (если `regex incl` равно "(?i)^.+\.zip$" и `max depth` не задан, то будут обходиться все подкаталоги, но учитываться будут только файлы типа zip).
- Время выполнения ограничено значением тайм-аута в конфигурации агента configuration (3 sec). Поскольку обход больших каталогов может занять больше времени, данные возвращены не будут, и элемент данных перейдет в состояние неподдерживаемого. Частичный результат не возвращается.
- При фильтрации по размеру значимы только обычные файлы. В Linux и BSD каталоги тоже имеют ненулевой размер (обычно несколько КБ). У устройств размер равен нулю, например размер **/dev/sda1** не отражает размер соответствующего раздела. Поэтому при использовании `<min_size>` и `<max_size>` рекомендуется указывать `<types_incl>` как "*file*", чтобы избежать неожиданностей.

Примеры:

```
vfs.dir.count[/dev] #                    /dev (Linux)
vfs.dir.count["C:\Users\ADMINI~1\AppData\Local\Temp"] #
```

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>]

<br> Список записей каталога.<br> Возвращаемое значение: *JSON string*.<br> См. поддерживаемые платформы.

Параметры:

- **dir** - абсолютный путь к каталогу;
- **regex incl** - регулярное выражение, описывающее шаблон имени объекта (файл, каталог, символическая ссылка) для включения; если пусто, включаются все (значение по умолчанию);
- **regex excl** - регулярное выражение, описывающее шаблон имени объекта (файл, каталог, символическая ссылка) для исключения; если пусто, ничего не исключается (значение по умолчанию);

- **types incl** - типы записей каталога для вывода, возможные значения: *file* - обычный файл, *dir* - подкаталог, *sym* - символическая ссылка, *sock* - сокет, *bdev* - блочное устройство, *cdev* - символьное устройство, *fifo* - FIFO, *dev* - синоним для "bdev,cdev", *all* - все типы (по умолчанию), то есть "file,dir,sym,sock,bdev,cdev,fifo". Несколько типов должны быть разделены запятыми и заключены в кавычки.
- **types excl** - типы записей каталога (см. `types incl`), которые НЕ нужно выводить. Если какой-либо тип записи указан и в `types incl`, и в `types excl`, записи этого типа НЕ выводятся.
- **max depth** - максимальная глубина обхода подкаталогов:<br>**-1** (по умолчанию) - без ограничений,<br>**0** - без перехода в подкаталоги.
- **min size** - минимальный размер (в байтах) файла для вывода. Файлы меньшего размера не будут выведены. Можно использовать суффиксы памяти.
- **max size** - максимальный размер (в байтах) файла для вывода. Файлы большего размера не будут выведены. Можно использовать суффиксы памяти.
- **min age** - минимальный возраст (в секундах) записи каталога для вывода. Более новые записи не будут выведены. Можно использовать суффиксы времени.
- **max age** - максимальный возраст (в секундах) записи каталога для вывода. Записи такой давности и старше не будут выведены (время изменения). Можно использовать суффиксы времени.
- **regex excl dir** - регулярное выражение, описывающее шаблон имени каталога для исключения. Все содержимое каталога будет исключено (в отличие от `regex excl`)

Комментарии:

- В Windows этот элемент данных следует по UNC-путям, что может быть риском безопасности.
- Переменные среды, например %APP_HOME%, $HOME и %TEMP%, не поддерживаются.
- Псевдокаталоги "." и ".." никогда не выводятся.
- Символические ссылки никогда не используются для обхода каталогов.
- И `regex incl`, и `regex excl` применяются к файлам и каталогам при формировании списка записей, но игнорируются при выборе подкаталогов для обхода (если `regex incl` равно "(?i)^.+\.zip$" и `max depth` не задан, то будут обходиться все подкаталоги, но учитываться будут только файлы типа zip).
- Время выполнения ограничено значением тайм-аута в конфигурации агента configuration. Поскольку обход больших каталогов может занять больше времени, данные не будут возвращены, и элемент данных станет неподдерживаемым. Частичный список возвращен не будет.
- При фильтрации по размеру значимы только размеры обычных файлов. В Linux и BSD каталоги также имеют ненулевой размер (обычно несколько КБ). Устройствам соответствует нулевой размер, например размер **/dev/sda1** не отражает размер соответствующего раздела. Поэтому при использовании `min size` и `max size` рекомендуется указывать `types incl` как "*file*", чтобы избежать неожиданностей.

Примеры:

```
vfs.dir.get[/dev] #                  /dev (Linux)
vfs.dir.get["C:\Users\ADMINI~1\AppData\Local\Temp"] #
```

vfs.dir.size[директория,<regex_вкл>,<regex_искл>,<режим>,<макс_глубина>,<regex_искл_директории>]

<br> Размер директории (в байтах).<br> Возвращаемое значение: *Целое число*.<br> Поддерживаемые платформы: Linux, Windows. Элемент данных может работать на других UNIX-подобных платформах.

Параметры:

- **директория** — абсолютный путь к директории;
- **regex_вкл** — регулярное выражение, описывающее шаблон включаемых имён сущностей (файлов, директорий и символических ссылок); если пусто, включается всё (по умолчанию);
- **regex_искл** — регулярное выражение, описывающее шаблон исключаемых имен сущностей (файлов, директорий и символических ссылок); если пусто (по умолчанию), исключения не применяются;
- **режим** — возможные значения: *apparent* (по умолчанию) — получение реальных размеров файлов вместо использования диска (работает как `du -sb `), *disk* — получение использования диска (работает как `du -s -B1 `). В отличие от команды du, элемент данных vfs.dir.size при вычислении размера директории учитывает скрытые файлы (работает как `du -sb .[^.]* *` в пределах директории).
- **макс_глубина** — максимальная глубина сканируемых поддиректорий: **-1** (по умолчанию) — без ограничения, **0** — не опускаться в поддиректории.
- **regex_искл_директории** — регулярное выражение, описывающее шаблон имён исключаемых директорий. Всё содержимое директории будет исключено (в отличие от `regex_ `).

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Подсчитываются только те директории, к которым пользователь *zabbix* имеет по крайней мере права чтения. Для директорий с доступом только на чтение рассчитывается размер самой директории. Директории с доступом на чтение и выполнение обсчитываются, включая содержимое.

- При наличии больших директорий или медленных дисков этот элемент данных может превысить время ожидания из-за настройки Timeout в файлах конфигурации агента и сервера/прокси. При необходимости увеличьте эти значения времени ожидания.
- Ограничение размера файлов зависит от наличия поддержки больших файлов.

Примеры:

```
vfs.dir.size[/tmp,log] #                    /tmp,                    «log»
vfs.dir.size[/tmp,log,^.+\.old$] #                    /tmp,                    «log»,                ,
```

vfs.file.cksum[файл,<режим>]

<br> Контрольная сумма файла, вычисленная по алгоритму, используемому в UNIX cksum.<br> Возвращаемое значение: *Целое число* — с в значении *crc32*, *Строка* — с в значениях *md5*, *sha256*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **режим** — *crc32* (default), *md5* или *sha256*.

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Ограничение размера файла зависит от поддержки больших файлов.

Пример:

```
vfs.file.cksum[/etc/passwd]
```

Примеры возвращаемых значений (для crc32/md5/sha256 соответственно):

```
675436101
9845acf68b73991eb7fd7ee0ded23c44
ae67546e4aac995e5c921042d0cf0f1f7147703aa42bfbfb65404b30f238f2dc
```

vfs.file.contents[файл,<кодировка>]

<br> Получение содержимого файла[7].<br> Возвращаемое значение: *Текст*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **кодировка** — идентификатор кодовой страницы.

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Возвращаемое значение ограничено 16 МБ (включая конечные пробелы, которые усекаются); также применяются ограничения баз данных;
- Возвращается пустая строка, если файл пустой или содержит только символы LF/CR;
- Маркер последовательности байтов (BOM) исключён из вывода.

Пример:

```
vfs.file.contents[/etc/passwd]
```

vfs.file.exists[файл,<типы_вкл>,<типы_искл>]

<br> Проверка существования файла.<br> Возвращаемое значение: 0 — файл не найден; 1 — файл заданного типа существует.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **типы_вкл** — список включаемых типов файлов, возможные значения: *file* (обычный файл, по умолчанию (если типы_искл не задан)), *dir* (директория), *sym* (символическая ссылка), *sock* (сокет), *bdev* (блочное устройство), *cdev* (устройство посимвольного ввода-вывода), *fifo* (FIFO), *dev* (синоним к "bdev,cdev"), *all* (все вышеупомянутые типы, по умолчанию если типы_искл задан).
- **типы_искл** — список исключаемых типов файлов, смотрите типы_вкл для получения возможных значений (по умолчанию никакие типы не исключаются).

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;

- Несколько типов необходимо разделять запятой и весь список заключать в "" кавычки;
- Если какие-либо типы записей указаны как в <типы_вкл>, так и в <типы_искл>, файлы этих типов будут исключены;
- Ограничение размера файла зависит от поддержки больших файлов.

Примеры:

```
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]

<br> Возвращает информацию о файле.<br> Возвращаемое значение: *JSON string*.<br> См. поддерживаемые платформы.

Параметр:

- **file** - полный путь к файлу

Комментарии:

- В Windows этот элемент данных следует по UNC-путям, что может представлять риск для безопасности.
- Поддерживаемые типы файлов в UNIX-подобных системах: обычный файл, каталог, символическая ссылка, сокет, блочное устройство, символьное устройство, FIFO.
- Ограничение размера файла зависит от поддержки больших файлов.

Пример:

```
vfs.file.get[/etc/passwd] #return a JSON with information about the /etc/passwd file (type, user, permissi
```

vfs.file.md5sum[файл]

<br> MD5 контрольная сумма файла.<br> Возвращаемое значение: Строка (MD5 хеш-сумма файла).<br> Смотрите Поддерживаемые платформы.

Параметр:

- **файл** — абсолютный путь к файлу

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Ограничение размера файла зависит от поддержки больших файлов.

Пример:

```
vfs.file.md5sum[/usr/local/etc/zabbix_agentd.conf]
```

Пример возвращаемого значения:

```
b5052decb577e0fffd622d6ddc017e82
```

vfs.file.owner[файл,<тип_владельца>,<тип_результата>]

<br> Получение владельца файла.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **тип_владельца** — *user* (по умолчанию) или *group* (только в Unix);
- **тип_результата** — *name* (по умолчанию) или *id*; в случае с id — получение uid/gid на Unix, SID на Windows.

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Ограничение размера файла зависит от поддержки больших файлов.

Примеры:

```
vfs.file.owner[/tmp/zabbix_server.log] #               /tmp/zabbix_server.log
vfs.file.owner[/tmp/zabbix_server.log,,id] #      ID           /tmp/zabbix_server.log
```

vfs.file.permissions[файл]

<br> Получение строки из 4 цифр, содержащей восьмеричное число с правами доступа Unix.<br> Возвращаемое значение: *Строка*.<br> Поддерживаемые платформы: Linux. Элемент данных может работать на других UNIX-подобных платформах.

Параметры:

- **файл** — абсолютный путь к файлу

Ограничение размера файла зависит от поддержки больших файлов.

Пример:

```
vfs.file.permissions[/etc/passwd] #                /etc/passwd,      , «0644»
```

vfs.file.regexp[файл,регулярное выражение,<кодировка>,<начальная строка>,<конечная строка>,<вывод>]

<br> Извлечение строки в файле[7].<br> Возвращаемое значение: Строка, содержащая совпадающую подстроку, или то, что определено в дополнительном параметре .<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **регулярное выражение** — регулярное выражение, описывающее требуемый шаблон;
- **кодировка** — идентификатор кодовой страницы;
- **начальная строка** — номер первой строки для поиска (по умолчанию первая строка файла);
- **конечная строка** — номер последней строки для поиска (по умолчанию последняя строка файла);
- **вывод** — дополнительный шаблон форматирования вывода. Управляющая последовательность **\0** заменяется найденной частью текста (начиная от первого символа совпадения, заканчивая символом, где совпадение заканчивается), тогда как управляющая последовательность **\N** (где N=1..9) заменяется N-ной совпадающей группой (или пустой строкой, если N превышает количество найденных групп).

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Ограничение размера файла зависит от поддержки больших файлов.
- Возвращается только первая совпавшая строка;
- Если совпадений с выражением не найдено, будет возвращена пустая строка;
- Маркер последовательности байтов (BOM) исключается из вывода;
- Извлечение содержимого при использовании параметра выполняется агентом.

Примеры:

```
vfs.file.regexp[/etc/passwd,zabbix]
vfs.file.regexp[/path/to/some/file,"([0-9]+)$",,3,5,\1]
vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1] →       ID         *zabbix*
```

vfs.file.regmatch[файл,регулярное выражение,<кодировка>,<начальная строка>,<конечная строка>]

<br> Поиск строки в файле[7].<br> Возвращаемые значения: 0 — совпадение не найдено; 1 — найдено.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **регулярное выражение** — регулярное выражение, описывающее требуемый шаблон;
- **кодировка** — идентификатор кодовой страницы;
- **начальная строка** — номер первой строки для поиска (по умолчанию первая строка файла);
- **конечная строка** — номер последней строки для поиска (по умолчанию последняя строка файла).

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Ограничение размера файла зависит от поддержки больших файлов.
- Маркер последовательности байтов (BOM) игнорируется.

Пример:

```
vfs.file.regmatch[/var/log/app.log,error]
```

vfs.file.size[файл,<режим>]

<br> Размер файла (в байтах).<br> Возвращаемое значение: *Целое число*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — абсолютный путь к файлу;
- **режим** — возможные значения: *bytes* (по умолчанию) или *lines* (пустые строки также считаются).

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Пользователь *zabbix* должен иметь права на чтение файла;
- Ограничение размера файла зависит от поддержки больших файлов.

Пример:

```
vfs.file.size[/var/log/syslog]
```

vfs.file.time[файл,<режим>]

<br> Информация о времени файла.<br> Возвращаемое значение: *Целое число* (Unix timestamp).<br> Смотрите Поддерживаемые платформы.

Параметры:

- **файл** — полный путь к файлу;
- **режим** — возможные значения:<br>*modify* (по умолчанию) — время последней модификации содержимого файла,<br>*access* — время последнего чтения файла,<br>*change* — время последнего изменения свойств файла.

Комментарии:

- В Windows этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Ограничение размера файла зависит от поддержки больших файлов.

Пример:

```
vfs.file.time[/etc/passwd,modify]
```

vfs.fs.discovery

<br> Список смонтированных файловых систем с указанием их типа и параметров монтирования. Используется для обнаружения на низком уровне.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

vfs.fs.get

<br> Список смонтированных файловых систем с их типом, доступным дисковым пространством, статистикой inode и параметрами монтирования. Может использоваться для низкоуровневого обнаружения.<br> Возвращаемое значение: *JSON string*.<br> Поддерживаемые платформы: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Комментарии:

- Также сообщаются файловые системы, у которых количество inode равно нулю, что может быть характерно для файловых систем с динамическими inode (например, btrfs).
- См. также: Обнаружение смонтированных файловых систем.

vfs.fs.inode[fs,<режим>]

<br> Количество или процент inodes.<br> Возвращаемое значение: *Целое число* — для количества; *Число с плавающей точной* — для процентов.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **fs** — файловая система;
- **режим** — возможные значения: *total* (по умолчанию), *free*, *used*, *pfree* (свободно, в процентах) или *pused* (использовано, в процентах).

Если количество inode'ов станет нулевым, что может случаться на файловых системах с динамическим количеством файловых дескрипторов (например, btrfs), значения pfree/pused будут сообщаться как «100» и «0» соответственно.

Пример:

```
vfs.fs.inode[/,pfree]
```

vfs.fs.size[fs,<режим>]

<br> Дисковое пространство в байтах или процентах от общего размера.<br> Возвращаемое значение: *Целое число* — для байт; *Число с плавающей точкой* — для процентов.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **fs** — файловая система;
- **режим** — возможные значения: *total* (по умолчанию), *free*, *used*, *pfree* (доступно, в процентах) или *pused* (использовано, в процентах).

Комментарии:

- Если файловая система не примонтирована, будет возвращён размер локальной файловой системы, где расположена точка монтирования;
- Зарезервированное место на файловой системе принимается во внимание и не включено при использовании режима *free*.

Пример:

```
vfs.fs.size[/tmp,free]
```

vm.memory.size[<mode>]

<br> Размер памяти в байтах или в процентах от общего объема.<br> Возвращаемое значение: *Integer* - для байтов; *Float* - для процентов.<br> См. поддерживаемые платформы.

Параметр:

- **mode** - возможные значения: *total* (по умолчанию), *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*, *used*, *pused* (used, percentage), *available* или *pavailable* (available, percentage).

Комментарии:

- Этот элемент данных принимает три категории параметров:
  - *total* - общий объем памяти;
  - типы памяти, зависящие от платформы: *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*;
  - оценки на уровне пользователя о том, сколько памяти используется и доступно: *used*, *pused*, *available*, *pavailable*.
- Параметр режима *active* поддерживается только в FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Параметры режима *anon*, *exec*, *file* поддерживаются только в NetBSD.
- Параметр режима *buffers* поддерживается только в Linux, FreeBSD, OpenBSD, NetBSD.
- Параметр режима *cached* поддерживается только в Linux, FreeBSD, AIX, OpenBSD, NetBSD.
- Параметры режима *inactive*, *wired* поддерживаются только в FreeBSD, MacOS X, OpenBSD, NetBSD.
- Параметр режима *pinned* поддерживается только в AIX.
- Параметр режима *shared* поддерживается только в Linux 2.4, FreeBSD, OpenBSD, NetBSD.
- См. также дополнительные сведения для этого элемента данных.

Пример:

```
vm.memory.size[pavailable]
```

web.page.get[хост,<путь>,<порт>]

<br> Получение содержимого веб-страницы.<br> Возвращаемое значение: Исходная веб-страница в виде текста (включая заголовки).<br> Смотрите Поддерживаемые платформы.

Параметры:

- **хост** — имя хоста или URL (в виде `://:/`, где обязателен только *хост*). Допустимые схемы URL: *http*, *https*[4]. При отсутствии схемы будет обрабатываться как *http*. Если указан URL, тогда параметры и должны быть пустыми. При подключении к серверам, которые требуют аутентификацию, возможность указать имя пользователя/пароль, например: `http://user:password@www.example.com`, имеется только с поддержкой cURL [4]. В именах хостов поддерживается punycode.
- **путь** — путь к HTML документу (по умолчанию /);
- **порт** — номер порта (по умолчанию 80 для HTTP)

Комментарии:

- Этот элемент данных становится неподдерживаемым, если заданный в ресурс не существует или недоступен;
- может быть именем хоста, доменным именем, адресом IPv4 или IPv6. Но для поддержки IPv6 адресов Zabbix агент должен быть скомпилирован с включённой поддержкой IPv6.

Примеры:

```
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.perf[хост,<путь>,<порт>]

<br> Время полной загрузки веб-страницы (в секундах).<br> Возвращаемое значение: *Число с плавающей точкой*.<br> Смотрите Поддерживаемые платформы.

Параметры:

- **хост** — имя хоста или URL (в виде `://:/`, где обязателен только *хост*). Допустимые схемы URL: *http*, *https*[4]. При отсутствии схемы будет обрабатываться как *http*. Если указан URL, тогда параметры и должны быть пустыми. При подключении к серверам, которые требуют аутентификацию, возможность указать имя пользователя/пароль, например: `http://user:password@www.example.com`, имеется только с поддержкой cURL [4]. В именах хостов поддерживается punycode.
- **путь** — путь к HTML документу (по умолчанию /);
- **порт** — номер порта (по умолчанию 80 для HTTP)

Комментарии:

- Этот элемент данных становится неподдерживаемым, если заданный в ресурс не существует или недоступен;
- может быть именем хоста, доменым именем, адресом IPv4 или IPv6. Но для поддержки IPv6 адресов Zabbix агент должен быть скомпилирован с включённой поддержкой IPv6.

Примеры:

```
web.page.perf[www.example.com,index.php,80]
web.page.perf[https://www.example.com]
```

web.page.regexp[хост,<путь>,<порт>,регулярное выражение,<длина>,<вывод>]

<br> Поиск строки на веб-странице.<br> Возвращаемое значение: Совпадающая строка, или указанная опциональным параметром .<br> Смотрите Поддерживаемые платформы.

Параметры:

- **хост** — имя хоста или URL (в виде `://:/`, где обязателен только *хост*). Допустимые схемы URL: *http*, *https*[4]. При отсутствии схемы будет обрабатываться как *http*. Если указан URL, тогда параметры и должны быть пустыми. При подключении к серверам, которые требуют аутентификацию, возможность указать имя пользователя/пароль, например: `http://user:password@www.example.com`, имеется только с поддержкой cURL [4]. В именах хостов поддерживается punycode.
- **путь** — путь к HTML документу (по умолчанию /);
- **порт** — номер порта (по умолчанию 80 для HTTP)
- **регулярное выражение** — регулярное выражение, описывающее требуемый шаблон;
- **длина** — максимальное количество возвращаемых символов;
- **вывод** — дополнительный шаблон форматирования вывода. Управляющая последовательность **\0** заменяется найденной частью текста (начиная от первого символа совпадения, заканчивая символом, где совпадение заканчивается), тогда как управляющая последовательность **\N** (где N=1..9) заменяется N-ой совпадающей группой (или пустой строкой, если N превышает количество найденных групп).

Комментарии:

- Этот элемент данных становится неподдерживаемым, если заданный в ресурс не существует или недоступен;
- может быть именем хоста, доменым именем, адресом IPv4 или IPv6. Но для поддержки IPv6 адресов Zabbix агент должен быть скомпилирован с включённой поддержкой IPv6.
- Извлечение содержимого при использовании параметра производится на стороне агента.

Примеры:

```
web.page.regexp[www.example.com,index.php,80,OK,2]
web.page.regexp[https://www.example.com,,,OK,2]
```

agent.hostmetadata

<br> Метаданные узла сети агента.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Возвращает значение параметров HostMetadata или HostMetadataItem, или пустую строку, если ничего не задано.

agent.hostname

<br> Имя узла сети агента.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Возвращает:

- Пассивной проверкой — имя первого узла сети, указанного в параметре Hostname в файле конфигурации агента;
- Активной проверкой — текущее значение имени хоста.

agent.ping

<br> Проверка доступности агента.<br> Возвращаемое значение: Ничего — недоступен; 1 — доступен.<br> Смотрите Поддерживаемые платформы.

Используйте триггерную функцию **nodata()** для проверки недоступности узла сети.

agent.variant

<br> Проверка разновидности агента (Zabbix агент или Zabbix агент 2).<br> Возвращаемое значение: 1 — Zabbix агент; 2 — Zabbix агент 2.<br> Смотрите Поддерживаемые платформы.

agent.version

<br> Версия Zabbix агента.<br> Возвращаемое значение: *Строка*.<br> Смотрите Поддерживаемые платформы.

Пример возвращаемого значения:

```
6.0.3
```

zabbix.stats[<ip>,<port>]

<br> Возвращает набор внутренних метрик сервера Zabbix или прокси. Если указаны `ip` и `port`, метрики запрашиваются у удаленного экземпляра; в противном случае — у локального экземпляра.<br> Возвращаемое значение: *JSON string*.<br> См. поддерживаемые платформы.

Параметры:

- **ip** - список IP/DNS/сетевых масок серверов/прокси, к которым выполняется удаленный запрос (по умолчанию 127.0.0.1);
- **port** - порт сервера/прокси, к которому выполняется удаленный запрос (по умолчанию 10051)

Комментарии:

- Этот элемент данных возвращает выбранный набор внутренних метрик. Подробности см. в разделе Удаленный мониторинг статистики Zabbix.
- Обратите внимание, что запрос статистики будет принят только с адресов, перечисленных в параметре 'StatsAllowedIP' server/proxy на целевом экземпляре.

zabbix.stats[<ip>,<port>,queue,<from>,<to>]

<br> Возвращает количество контролируемых элементов данных в очереди, которые задерживаются на сервере Zabbix или прокси. Если указаны `ip` и `port`, метрики извлекаются с удаленного экземпляра; в противном случае — с локального экземпляра.<br> Возвращаемое значение: *JSON string*.<br> См. поддерживаемые платформы.

Параметры:

- **ip** - список IP/DNS/сетевых масок серверов/прокси, к которым выполняется удаленный запрос (по умолчанию 127.0.0.1);
- **port** - порт сервера/прокси, к которому выполняется удаленный запрос (по умолчанию 10051)
- **queue** - константа (используется как есть)
- **from** - задержка не менее чем на (по умолчанию 6 секунд)
- **to** - задержка не более чем на (по умолчанию бесконечность)

Обратите внимание, что запрос stats будет принят только с адресов, перечисленных в параметре 'StatsAllowedIP' server/proxy на целевом экземпляре.

Примечания

[1] Специфичное для Linux примечание. Zabbix агент должен иметь права «только на чтение» на файловую систему */proc*. Патчи к ядру от www.grsecurity.org ограничивают права доступа непривилегированных пользователей.

[2] `vfs.dev.read[]`, `vfs.dev.write[]`: Zabbix агент будет прерывать «протухшие» подключения к устройствам, если значения элементов данных не опрашиваются более 3 часов. Это может произойти, если система имеет устройства с динамически меняющимися путями или если устройство было удалено вручную. Обратите также внимание, что эти элементы данных при использовании интервала обновления от 3 часов и более всегда будут возвращать значение «0».

[3] `vfs.dev.read[]`, `vfs.dev.write[]`: Если первым параметром используется параметр по умолчанию *all*, тогда ключ вернёт суммарную статистику, включая все блочные устройства, такие как sda, sdb и их разделы (sda1, sda2, sdb3...) и несколько устройств (MD raid) на основе этих блочных устройств/разделов и логические разделы (LVM) на основе этих блочных устройств/разделов. В этих случаях возвращаемые значения следует рассматривать как относительные значения (изменяемые во времени), но не как абсолютные значения.

[4] SSL (HTTPS) поддерживается, только если агент скомпилирован с поддержкой cURL. В противном случае элемент данных станет неподдерживаемым.

[5] Значения `bytes` и `errors` не поддерживаются для интерфейсов loopback в системах Solaris до Solaris 10 6/06 включительно, поскольку статистика байтов, ошибок и использования не сохраняется и/или не сообщается ядром. Однако, если вы отслеживаете систему Solaris через net-snmp, значения могут быть возвращены, поскольку net-snmp несёт устаревший код из cmu-snmp, датируемый 1997 годом, который при неудачном чтении значений байтов из статистики интерфейса возвращает счётчик пакетов (который существует на интерфейсах loopback), умноженный на произвольное значение 308. Предполагается, что средняя длина пакета составляет 308 октетов, что является очень грубой оценкой, поскольку ограничение MTU в системах Solaris для интерфейсов loopback составляет 8892 байта. Эти значения не следует считать правильными или даже близкими к точным. Они являются приблизительными. Zabbix агент не делает никаких предположений, но net-snmp вернёт значение для этих полей.

[6] Командная строка в Solaris, полученная из /proc/pid/psinfo, ограничена 80 байтами и содержит командную строку в том виде, в котором она была при запуске процесса.

[7] Элементы данных `vfs.file.contents[]`, `vfs.file.regexp[]`, `vfs.file.regmatch[]` можно использовать для получения содержимого файлов. Если вы хотите ограничить доступ к определённым файлам с конфиденциальной информацией, запустите агент Zabbix под пользователем, у которого нет прав доступа для просмотра этих файлов.

Использование с утилитами командной строки

Обратите внимание, при тестировании или использовании ключей элементов данных с утилитами zabbix_agentd или zabbix_get из командной строки вы также должны учитывать синтаксис командной оболочки.

Например, если определённый параметр ключа должен быть заключён в двойные кавычки, вам придется явно экранировать эти двойные кавычки, в противном случае они будут обрезаны командной оболочкой как специальные символы и не будут переданы утилите Zabbix.

Примеры:

```
zabbix_agentd -t 'vfs.dir.count[/var/log,,,"file,dir",,0]'

zabbix_agentd -t vfs.dir.count[/var/log,,,\"file,dir\",,0]
```

Настройки кодировки

Чтобы убедиться, что полученные данные не повреждены, вы можете указать правильную кодировку для обработки проверки (например, `vfs.file.contents`) в параметре `encoding`. Список поддерживаемых кодировок (идентификаторов кодовых страниц) можно найти в документации libiconv (проект GNU) или в документации Microsoft Windows SDK по "Code Page Identifiers". Обратите внимание, что Microsoft иногда помечает некоторые кодовые страницы как "доступные только управляемым приложениям" — то есть доступные только через среду выполнения .NET и недоступные через нативные API Win32. Агент Zabbix реализует собственную логику преобразования кодировки, поэтому такие кодовые страницы поддерживаются агентом Zabbix даже тогда, когда нативные функции Windows их не предоставляют.

Если в параметре `encoding` кодировка не указана, применяются следующие стратегии определения:

- Если кодировка не указана (или указана пустая строка), считается, что используется UTF-8, данные обрабатываются "как есть";
- Анализ BOM - применимо к элементам данных 'vfs.file.contents', 'vfs.file.regexp', 'vfs.file.regmatch'. Пытается определить правильную кодировку с использованием сигнатуры порядка байтов (BOM) в начале файла. Если BOM отсутствует - вместо этого применяется стандартное определение (см. выше).

Устранение неполадок с элементами данных агента

В случае пассивных проверок, чтобы предотвратить отсутствие значения элемента данных из-за истечения времени ожидания запроса сервера к агенту, следует отметить следующее:

- Если версия агента старее версии сервера, может потребоваться, чтобы значение *Timeout* в настройках элемента данных (или глобальное время ожидания) было больше значения `Timeout` в файле конфигурации агента.
- Если версия агента новее версии сервера, может потребоваться, чтобы значение `Timeout` в файле конфигурации сервера было больше значения `Timeout` в файле конфигурации агента.

1. Zabbix агент 2

Агент Zabbix 2 поддерживает все ключи элементов данных, поддерживаемые для агента Zabbix на Unix и Windows. На этой странице приведены сведения о дополнительных ключах элементов данных, которые можно использовать только с агентом Zabbix 2, сгруппированных по плагинам, к которым они относятся.

Ключи элементов данных перечислены без параметров и дополнительной информации. Щелкните ключ элемента данных, чтобы увидеть полные сведения.

| Ключ элемента данных | Описание | Плагин |
|---|---|---|
| ceph.df.details | Использование данных кластера и их распределение по пулам. | Ceph |
| ceph.osd.stats | Сводная статистика и статистика по каждому OSD. | |
| ceph.osd.discovery | Список обнаруженных OSD. | |
| ceph.osd.dump | Пороги использования и состояния OSD. | |
| ceph.ping | Проверяет, можно ли установить соединение с Ceph. | |
| ceph.pool.discovery | Список обнаруженных пулов. | |
| ceph.status | Общее состояние кластера. | |
| docker.container_info | Низкоуровневая информация о контейнере. | Docker |
| docker.container_stats | Статистика использования ресурсов контейнера. | |
| docker.containers | Возвращает список контейнеров. | |
| docker.containers.discovery | Возвращает список контейнеров. Используется для низкоуровневого обнаружения. | |
| docker.data.usage | Информация о текущем использовании данных. | |
| docker.images | Возвращает список образов. | |
| docker.images.discovery | Возвращает список образов. Используется для низкоуровневого обнаружения. | |
| docker.info | Системная информация. | |
| docker.ping | Проверяет, работает ли демон Docker. | |
| ember.get | Возвращает результат для требуемого устройства. | Ember+ |
| memcached.ping | Проверяет, активно ли соединение. | Memcached |
| memcached.stats | Получает вывод команды STATS. | |
| mongodb.collection.stats | Возвращает различные статистические данные о хранилище для заданной коллекции. | MongoDB |
| mongodb.collections.discovery | Возвращает список обнаруженных коллекций. | |
| mongodb.collections.usage | Возвращает статистику использования коллекций. | |
| mongodb.connpool.stats | Возвращает сведения об открытых исходящих соединениях от текущего экземпляра базы данных к другим участникам шардированного кластера или набора реплик. | |
| mongodb.db.stats | Возвращает статистику, отражающую состояние заданной базы данных. | |
| mongodb.db.discovery | Возвращает список обнаруженных баз данных. | |
| mongodb.jumbo_chunks.count | Возвращает количество jumbo chunks. | |
| mongodb.oplog.stats | Возвращает состояние набора реплик, используя данные, опрошенные из oplog. | |
| mongodb.ping | Проверяет, активно ли соединение. | |
| mongodb.rs.config | Возвращает текущую конфигурацию набора реплик. | |
| mongodb.rs.status | Возвращает состояние набора реплик с точки зрения участника, на котором выполняется метод. | |
| mongodb.server.status | Возвращает состояние базы данных. | |
| mongodb.sh.discovery | Возвращает список обнаруженных шардов, присутствующих в кластере. | |
| mongodb.version | Возвращает версию сервера базы данных. | |
| mqtt.get | Подписывается на определенную тему или темы (с подстановочными знаками) указанного брокера и ожидает публикаций. | MQTT |
| mssql.availability.group.get | Возвращает группы доступности. | MSSQL |
| mssql.custom.query | Возвращает результат пользовательского запроса. | |
| mssql.db.get | Возвращает все доступные базы данных MSSQL. | |
| mssql.job.status.get | Возвращает состояние заданий. | |
| mssql.last.backup.get | Возвращает время последнего резервного копирования для всех баз данных. | |
| mssql.local.db.get | Возвращает базы данных, участвующие в группе доступности Always On и реплике (основной или вторичной), которые расположены на сервере, к которому было установлено соединение. | |
| mssql.mirroring.get | Возвращает сведения о зеркалировании. | |
| mssql.nonlocal.db.get | Возвращает базы данных, участвующие в группе доступности Always On и реплике (основной или вторичной), расположенные на других серверах (база данных не является локальной для экземпляра SQL Server, к которому было установлено соединение). | |
| mssql.perfcounter.get | Возвращает счетчики производительности. | |
| mssql.ping | Проверяет, активно ли соединение. | |
| mssql.quorum.get | Возвращает сведения о кворуме. | |
| mssql.quorum.member.get | Возвращает участников кворума. | |
| mssql.replica.get | Возвращает реплики. | |
| mssql.version | Возвращает версию MSSQL. | |

| Ключ элемента данных | Описание | Плагин |
|---|---|---|
| mysql.custom.query | Возвращает результат пользовательского запроса. | MySQL |
| mysql.db.discovery | Возвращает список баз данных MySQL. | |
| mysql.db.size | Размер базы данных в байтах. | |
| mysql.get_status_variables | Значения глобальных переменных состояния. | |
| mysql.ping | Проверяет, активно ли соединение. | |
| mysql.replication.discovery | Возвращает список репликаций MySQL. | |
| mysql.replication.get_slave_status | Состояние репликации. | |
| mysql.version | Версия MySQL. | |
| net.dns.get | Выполняет DNS-запрос и возвращает подробные сведения о DNS-записи. | Network |
| nvml.device.count | Количество GPU-устройств. | NVIDIA GPU |
| nvml.device.decoder.utilization | Использование декодера GPU-устройства. | |
| nvml.device.ecc.mode | Режим ECC GPU-устройства. | |
| nvml.device.encoder.stats.get | Статистика кодировщика GPU-устройства. | |
| nvml.device.encoder.utilization | Использование кодировщика GPU-устройства. | |
| nvml.device.energy.consumption | Потребление энергии GPU-устройством. | |
| nvml.device.errors.memory | Сведения об ошибках памяти ECC GPU-устройства. | |
| nvml.device.errors.register | Сведения об ошибках регистров ECC GPU-устройства. | |
| nvml.device.fan.speed.avg | Средняя скорость вентилятора GPU-устройства. | |
| nvml.device.get | Обнаруженные GPU с UUID и именем. | |
| nvml.device.graphics.frequency | Частота графического ядра GPU-устройства. | |
| nvml.device.memory.bar1.get | Сведения о памяти BAR1 GPU-устройства. | |
| nvml.device.memory.fb.get | Сведения о памяти framebuffer GPU-устройства. | |
| nvml.device.memory.frequency | Частота памяти GPU-устройства. | |
| nvml.device.pci.utilization | Использование PCI GPU-устройства. | |
| nvml.device.performance.state | Состояние производительности GPU-устройства. | |
| nvml.device.power.limit | Ограничение мощности GPU-устройства. | |
| nvml.device.power.usage | Потребление мощности GPU-устройством. | |
| nvml.device.serial | Серийный номер GPU-устройства. | |
| nvml.device.sm.frequency | Частота streaming multiprocessor GPU-устройства. | |
| nvml.device.temperature | Температура GPU-устройства. | |
| nvml.device.utilization | Статистика использования GPU-устройства. | |
| nvml.device.video.frequency | Частота видеоядра GPU-устройства. | |
| nvml.system.driver.version | Версия драйвера NVIDIA. | |
| nvml.version | Версия библиотеки NVML. | |
| oracle.diskgroups.stats | Возвращает статистику дисковых групп Automatic Storage Management (ASM). | Oracle |
| oracle.diskgroups.discovery | Возвращает список дисковых групп ASM. | |
| oracle.archive.info | Статистика архивных журналов. | |
| oracle.cdb.info | Сведения о Container Databases (CDBs). | |
| oracle.custom.query | Результат пользовательского запроса. | |
| oracle.datafiles.stats | Возвращает статистику файлов данных. | |
| oracle.db.discovery | Возвращает список баз данных. | |
| oracle.fra.stats | Возвращает статистику Fast Recovery Area (FRA). | |
| oracle.instance.info | Статистика экземпляра. | |
| oracle.pdb.info | Сведения о Pluggable Databases (PDBs). | |
| oracle.pdb.discovery | Возвращает список PDB. | |
| oracle.pga.stats | Возвращает статистику Program Global Area (PGA). | |
| oracle.ping | Проверяет, можно ли установить соединение с Oracle. | |
| oracle.proc.stats | Возвращает статистику процессов. | |
| oracle.redolog.info | Сведения о файлах журнала из control file. | |
| oracle.sga.stats | Возвращает статистику System Global Area (SGA). | |
| oracle.sessions.stats | Возвращает статистику сеансов. | |
| oracle.sys.metrics | Возвращает набор значений системных метрик. | |
| oracle.sys.params | Возвращает набор значений системных параметров. | |
| oracle.ts.stats | Возвращает статистику табличных пространств. | |
| oracle.ts.discovery | Возвращает список табличных пространств. | |
| oracle.user.info | Возвращает сведения о пользователе Oracle. | |
| oracle.version | Возвращает версию сервера базы данных. | |
| pgsql.autovacuum.count | Количество рабочих процессов autovacuum. | PostgreSQL |
| pgsql.archive | Сведения об архивированных файлах. | |

| Ключ элемента данных | Описание | Плагин |
|---|---|---|
| pgsql.bgwriter | Общее количество контрольных точек для кластера базы данных с разбивкой по типу контрольной точки. | |
| pgsql.cache.hit | Коэффициент попаданий в буферный кэш PostgreSQL. | |
| pgsql.connections | Возвращает соединения по типам. | |
| pgsql.custom.query | Возвращает результат пользовательского запроса. | |
| pgsql.db.age | Возраст самого старого FrozenXID базы данных. | |
| pgsql.db.bloating_tables | Количество bloating-таблиц для каждой базы данных. | |
| pgsql.db.discovery | Список баз данных PostgreSQL. | |
| pgsql.db.size | Размер базы данных в байтах. | |
| pgsql.dbstat | Собирает статистику по каждой базе данных. | |
| pgsql.dbstat.sum | Сводные данные по всем базам данных в кластере. | |
| pgsql.locks | Сведения о предоставленных блокировках по каждой базе данных. | |
| pgsql.oldest.xid | Возраст самого старого XID. | |
| pgsql.ping | Проверяет, активно ли соединение. | |
| pgsql.queries | Метрики запросов по времени выполнения. | |
| pgsql.replication.count | Количество резервных серверов. | |
| pgsql.replication.process | Задержка flush, задержка write и задержка replay для каждого процесса-отправителя. | |
| pgsql.replication.process.discovery | Обнаружение имен процессов репликации. | |
| pgsql.replication.recovery_role | Состояние восстановления. | |
| pgsql.replication.status | Состояние репликации. | |
| pgsql.replication_lag.b | Задержка репликации в байтах. | |
| pgsql.replication_lag.sec | Задержка репликации в секундах. | |
| pgsql.uptime | Время работы PostgreSQL в миллисекундах. | |
| pgsql.version | Возвращает версию PostgreSQL. | |
| pgsql.wal.stat | Статистика WAL. | |
| redis.config | Получает параметры конфигурации экземпляра Redis, соответствующие шаблону. | Redis |
| redis.info | Получает вывод команды INFO. | |
| redis.ping | Проверяет, активно ли соединение. | |
| redis.slowlog.count | Количество записей slow log с момента запуска Redis. | |
| smart.attribute.discovery | Возвращает список атрибутов устройств S.M.A.R.T. | S.M.A.R.T. |
| smart.disk.discovery | Возвращает список устройств S.M.A.R.T. | |
| smart.disk.get | Возвращает все доступные свойства устройств S.M.A.R.T. | |
| systemd.unit.get | Возвращает все свойства unit systemd. | Systemd |
| systemd.unit.info | Сведения о unit systemd. | |
| systemd.unit.discovery | Список unit systemd и их сведения. | |
| web.certificate.get | Проверяет сертификаты и возвращает сведения о сертификате. | Web certificates |

См. также:

- Встроенные плагины
- Загружаемые плагины

Подробности ключей элементов данных

Параметры без угловых скобок обязательны. Параметры, обозначенные угловыми скобками **< >**, опциональны.

ceph.df.details[строка_подкл,<пользователь>,<ключ_api>]

<br> Использование данных кластера и распределение между пулами.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

ceph.osd.stats[строка_подкл,<пользователь>,<ключ_api]

<br> Агрегированная статистика и статистика по каждому OSD.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>

- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

ceph.osd.discovery[строка_подкл,<пользователь>,<ключ_api]

<br> Список обнаруженных OSD. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

ceph.osd.dump[строка_подкл,<пользователь>,<ключ_api]

<br> Пороги использования и состояния OSD.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

ceph.ping[строка_подкл,<пользователь>,<ключ_api>]

<br> Проверка, можно ли установить подключение к Ceph.<br> Возвращаемое значение: *0* — подключение разорвано (если имеется какая-либо ошибка, включая проблемы с аутентификацией или проблемы с настройками); *1* — подключение успешно.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

ceph.pool.discovery[строка_подкл,<пользователь>,<ключ_api>]

<br> Список обнаруженных пулов. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

ceph.status[строка_подкл,<пользователь>,<ключ_api>]

<br> Общее состояние кластера.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, ключ_api** — имя пользователя и ключ API для подключения к Ceph.<br>

docker.container_info[<ID>,<info>]

<br> Низкоуровневая информация о контейнере.<br> Возвращаемое значение: Вывод API вызова ContainerInspect [en], преобразованный в JSON.

Параметры:

- **ID** — ID или имя контейнера;<br>
- **info** — объём возвращаемой информации. Возможные значения: *short* (по умолчанию) или *full*.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.container_stats[<ID>]

<br> Статистика использования ресурсов контейнером.<br> Возвращаемое значение: Вывод API вызова ContainerStats [en] и использование CPU в процентах, преобразованные в JSON.

Параметр:

- **ID** — ID или имя контейнера.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.containers

<br> Список контейнеров.<br> Возвращаемое значение: Вывод API вызова ContainerList [en], преобразованный в JSON.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.containers.discovery[<опции>]

<br> Получение списка контейнеров. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметр:

- **опции** — задаёт, следует ли обнаруживать все или только запущенные контейнеры. Поддерживаемые значения: *true* — возвращать все контейнеры; *false* — возвращать только запущенные контейнеры (по умолчанию).

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.data.usage

<br> Информация о текущем использовании данных.<br> Возвращаемое значение: Вывод API вызова SystemDataUsage [en], преобразованный в JSON.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.images

<br> Получение списка образов.<br> Возвращаемое значение: Вывод API вызова ImageList [en], преобразованный в JSON.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.images.discovery

<br> Получение списка образов. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.info

<br> Информация о системе.<br> Возвращаемое значение: Вывод API вызова SystemInfo [en], преобразованный в JSON.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

docker.ping

<br> Проверка, активен или нет Docker демон.<br> Возвращаемое значение: *1* — подключение установлено; *0* — подключение разорвано.

Пользователь Агента 2 («zabbix») должен быть добавлен в группу [en] «docker» для достаточных привилегий. В противном случае проверка завершится с ошибкой.

ember.get[<uri>,<путь>]

<br> Возвращает результат требуемого устройства.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI устройства Ember+ . По умолчанию: 127.0.0.1:9998;<br>
- **путь** — OID путь к устройству. По умолчанию пусто, возвращает данные корневой коллекции.<br>

memcached.ping[строка_подкл,<пользователь>,<пароль>]

<br> Проверка, можно ли установить подключение.<br> Возвращаемое значение: *1* — подключение активно; *0* — подключение разорвано (если имеется какая-либо ошибка, включая проблемы с аутентификацией или проблемы с настройками).

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, пароль** — учётные данные для входа в Memchached.<br>

memcached.stats[строка_подкл,<пользователь>,<пароль>,<тип>]

<br> Получение вывода команды STATS.<br> Возвращаемое значение: *JSON* — вывод, преобразованный в JSON.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пользователь, пароль** — учётные данные для входа в Memchached;<br>
- **тип** — тип возвращаемой статистики: *items*, *sizes*, *slabs* или *settings* (по умолчанию пусто, возвращается общая статистика).

mongodb.collection.stats[строка_подкл,<имя_пользователя>,<пароль>,<база_данных>,коллекция]

<br> Получение различной статистики хранилища по заданной коллекции.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB;<br>
- **база_данных** — имя базы данных (по умолчанию: admin);<br>
- **коллекция** — имя коллекции.

mongodb.collections.discovery[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение списка обнаруженных коллекций. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.collections.usage[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение статистики использования по коллекциям.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.connpool.stats[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение информации касательно открытых исходящих подключений с текущего экземпляра базы данных к другим членам кластера сегментов или набора реплик.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB;<br>

mongodb.db.stats[строка_подкл,<имя_пользователя>,<пароль>,<база_данных>]

<br> Получение статистики, отражающей состояние системы заданной базы.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB;<br>
- **база_данных** — имя базы (по умолчанию: admin).<br>

mongodb.db.discovery[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение списка обнаруженных баз данных. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.jumbo_chunks.count[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение количества jumbo chunk.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.oplog.stats[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение состояния набора реплик, используя данные полученные от oplog.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.ping[строка_подкл,<имя_пользователя>,<пароль>]

<br> Проверка, можно ли установить подключение.<br> Возвращаемое значение: *1* — подключение успешно; *0* — подключение разорвано (если имеется какая-либо ошибка, включая проблемы с аутентификацией или проблемы с настройками).

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.rs.config[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение текущей конфигурации набора реплик.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.rs.status[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение состояния набора реплик с точки зрения члена, откуда вызывается метод.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.server.status[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение состояния базы данных.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.sh.discovery[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение списка обнаруженных сегментов, находящихся в кластере.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mongodb.version[строка_подкл,<имя_пользователя>,<пароль>]

<br> Получение версии сервера базы данных.<br> Возвращаемое значение: *Строка*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные к MongoDB.<br>

mqtt.get[<url_ брокера>,тема,<имя_пользователя>,<пароль>]

<br> Подписка на заданную тему или темы (с шаблонами), которые предоставляются брокером, и ожидание публикаций.<br> Возвращаемое значение: Зависит от содержимого темы. Если использовался шаблон, возвращает содержимое темы в виде JSON.

Параметры:

- **url_брокера** — URL брокера MQTT в формате `://` `:` без параметров запроса (поддерживаемые протоколы: `tcp`, `ssl`, `ws`). Если значение не указано, агент будет использовать `tcp://localhost:1883`. Если протокол или порт не указаны, будет использоваться протокол по умолчанию (`tcp`) или порт по умолчанию (1883); <br>

- **тема** — тема MQTT (обязательно). Поддерживаются шаблоны (+,#);<br>
- **имя_пользователя,пароль** — учётные данные для аутентификации (если требуется).<br>

Комментарии:

- Элемент данных необходимо настроить как активная проверка (тип элемента данных «Zabbix агент (активный)»);
- Можно использовать сертификаты TLS шифрования, сохранив их в место по умолчанию (например, в Ubuntu директория `/etc/ssl/certs/`). Для TLS используйте схему `tls://`.

mssql.availability.group.get[URI,<user>,<password>]

<br> Возвращает группы доступности.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **URI** - URI сервера MSSQL (единственная поддерживаемая схема — `sqlserver://`). Встроенные учетные данные будут проигнорированы. Можно указать имя экземпляра как часть URI, например: `sqlserver://localhost/InstanceName` (без порта). Если указан порт, имя экземпляра игнорируется;<br>
- **user, password** - имя пользователя, пароль для отправки на защищенный сервер MSSQL.<br>

Дополнительные сведения см. в файле readme MSSQL plugin.

mssql.custom.query[URI,<user>,<password>,queryName,<args...>]

<br> Возвращает результат пользовательского запроса.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **URI** - URI сервера MSSQL (единственная поддерживаемая схема — `sqlserver://`). Встроенные учетные данные будут проигнорированы. Можно указать имя экземпляра как часть URI, например: `sqlserver://localhost/InstanceName` (без порта). Если указан порт, имя экземпляра игнорируется;<br>
- **user, password** - имя пользователя, пароль для отправки на защищенный сервер MSSQL;<br>
- **queryName** - имя пользовательского запроса, настроенного в `Plugins.MSSQL.CustomQueriesDir`, без расширения .sql;<br>
- **args** - один или несколько аргументов, разделенных запятыми, для передачи в запрос.

Дополнительную информацию см. в файле README плагина MSSQL.

mssql.db.get

<br> Возвращает все доступные базы данных MSSQL.<br> Возвращаемое значение: *объект JSON*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mssql.job.status.get

<br> Возвращает статус заданий.<br> Возвращаемое значение: *объект JSON*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mssql.last.backup.get

<br> Возвращает время последнего резервного копирования для всех баз данных.<br> Возвращаемое значение: *объект JSON*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mssql.local.db.get

<br> Возвращает базы данных, которые участвуют в группе доступности Always On и реплике (основной или вторичной) и расположены на сервере, к которому было установлено соединение.<br> Возвращаемое значение: *объект JSON*.

Дополнительные сведения см. в файле readme MSSQL plugin.

mssql.mirroring.get

<br> Возвращает информацию о зеркалировании.<br> Возвращаемое значение: *JSON-объект*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mssql.nonlocal.db.get

<br> Возвращает базы данных, которые участвуют в группе доступности Always On и реплике (основной или вторичной), расположенной на других серверах (база данных не является локальной для экземпляра SQL Server, к которому было установлено подключение).<br> Возвращаемое значение: *JSON object*.

Дополнительные сведения см. в файле readme плагина MSSQL.

mssql.perfcounter.get

<br> Возвращает счетчики производительности.<br> Возвращаемое значение: *объект JSON*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mssql.ping

<br> Проверка доступности базы данных. Проверяет, правильно ли настроено соединение.<br> Возвращаемое значение: *1* - доступен, *0* - недоступен.

Для получения дополнительной информации см. файл readme плагина MSSQL.

mssql.quorum.get

<br> Возвращает информацию о кворуме.<br> Возвращаемое значение: *объект JSON*.

Дополнительные сведения см. в файле readme плагина MSSQL.

mssql.quorum.member.get

<br> Возвращает членов кворума.<br> Возвращаемое значение: *JSON object*.

Дополнительные сведения см. в файле readme плагина MSSQL.

mssql.replica.get

<br> Возвращает реплики.<br> Возвращаемое значение: *JSON object*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mssql.version

<br> Возвращает версию MSSQL.<br> Возвращаемое значение: *String*.

Дополнительную информацию см. в файле readme плагина MSSQL.

mysql.custom.query[connString,<user>,<password>,queryName,<args...>]

<br> Возвращает результат выполнения пользовательского запроса.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - URI или имя сеанса;<br>
- **user, password** - учетные данные для входа в MySQL;<br>
- **queryName** - имя пользовательского запроса, должно совпадать с именем файла SQL без расширения;<br>
- **args** - один или несколько аргументов, разделенных запятыми, для передачи в запрос.

Дополнительные сведения см. в файле readme плагина MySQL, раздел *Custom queries*.

mysql.db.discovery[строка_подкл, <имя_пользователя>, <пароль>]

<br> Возвращает список баз данных MySQL. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: Результат SQL запроса «show databases» в формате LLD JSON.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL.<br>

mysql.db.size[строка_подкл, <имя_пользователя>, <пароль>,<имя_базы>]

<br> Размер базы данных в байтах.<br> Возвращаемое значение: Результат SQL запроса «select coalesce(sum(data_length + index_length),0) as size from information_schema.tables where table_schema=?» для указанной базы данных в байтах.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL;<br>
- **имя_базы** — имя базы данных.

mysql.get_status_variables[строка_подкл, <имя_пользователя>, <пароль>]

<br> Значения глобальных переменных состояния.<br> Возвращаемое значение: Результат SQL запроса «show global status» в формате JSON.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL.<br>

mysql.ping[строка_подкл, <имя_пользователя>, <пароль>]

<br> Проверка, можно ли установить подключение.<br> Возвращаемое значение: *1* — подключение успешно; *0* — подключение разорвано (если имеется какая-либо ошибка, включая проблемы с аутентификацией или проблемы с настройками).

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL.<br>

mysql.replication.discovery[строка_подкл, <имя_пользователя>, <пароль>]

<br> Возвращает список репликаций MySQL. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: результат SQL запроса `SHOW SLAVE STATUS` или `SHOW REPLICA STATUS` в формате LLD JSON.

Parameters:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL.<br>

mysql.replication.get_slave_status[строка_подкл, <имя_пользователя>, <пароль>, <хост_мастера>]

<br> Состояние репликации.<br> Возвращаемое значение: Результат SQL запроса SHOW SLAVE STATUS  SHOW REPLICA STATUS' в формате JSON.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL;<br>
- **хост_мастера** — Имя хоста мастера репликации. Если не найдено, возвращается ошибка. Если данный параметр не указан, возвращаются все хосты.<br>

mysql.version[строка_подкл, <имя_пользователя>, <пароль>]

<br> Версия MySQL.<br> Возвращаемое значение: *Строка* (с версией экземпляра MySQL).

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные для входа в MySQL.<br>

net.dns.get[<ip>,имя,<тип>,<время ожидания>,<количество>,<протокол>,"<флаги>"]

Выполнение DNS-запроса и получение подробной информации о DNS-записи.<br> Этот элемент данных является расширенной версией элемента данных Zabbix агента `net.dns.record` с поддержкой большего количества типов записей и настраиваемыми флагами.<br> Возвращаемые значения: *Объект JSON*

Параметры:

- **ip** — IP адрес DNS сервера (оставьте пустым, чтобы использовались DNS сервера по умолчанию);
- **имя** — проверяемое имя DNS;
- **тип** — запрашиваемый тип записи (по умолчанию *SOA*);
- **время ожидания** — время ожидания ответа в секундах (по умолчанию 1 секунда);
- **количество** — количество попыток запросов (по умолчанию 2);
- **протокол** — используемый протокол при выполнении DNS запросов: *udp* (по умолчанию) или *tcp*;
- **флаги** — один или несколько аргументов через запятую для передачи в запрос.

Комментарии:

- Возможные значения для параметра : *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*. Обратите внимание, что значения должны быть только в верхнем регистре; значения в нижнем регистре или смешанном регистре не поддерживаются.
- Для обратных DNS-поисков (когда задан значением *PTR*) можно указать DNS-имя как в обратном, так и в необратном формате (см. примеры ниже). Обратите внимание, что при запросе записи PTR имя DNS на самом деле является IP-адресом.
- Возможные значения для параметра : *cdflag* или *nocdflag* (по умолчанию), *rdflag* (по умолчанию) или *nordflag*, *dnssec* или *nodnssec* (по умолчанию), *nsid* или *nonsid* (по умолчанию), *edns0* (по умолчанию) или *noedns0*, *aaflag* или *noaaflag* (по умолчанию), *adflag* или *noadflag* (по умолчанию). Значения параметра *dnssec* и *nsid* нельзя

использовать вместе с *noedns0*, так как для обоих требуется *edns0*. Обратите внимание, что значения должны быть только в нижнем регистре; значения в верхнем регистре или смешанные значения не поддерживаются.
- Интернационализированные доменные имена не поддерживаются; вместо них, пожалуйста, используйте имена в IDNA кодировке.
- Выходные данные представляют собой объект, содержащий информацию о записи DNS на основе предоставленных параметров (см. подробнее).

Примеры:

```
net.dns.get[192.0.2.0,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.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR,,,,"cdflag,rdfla
```

nvml.device.count

<br> Количество GPU-устройств.<br> Возвращаемое значение: *Integer*.

Дополнительную информацию см. в файле readme плагина NVIDIA GPU.

nvml.device.decoder.utilization[<deviceUUID>]

<br> Использование декодера устройства GPU в процентах.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.ecc.mode[<deviceUUID>]

<br> Сведения о режиме ECC устройства GPU (текущий, ожидающий).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.encoder.stats.get[<deviceUUID>]

<br> Статистика кодировщика устройства GPU.<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.encoder.utilization[<deviceUUID>]

<br> Загрузка кодировщика устройства GPU в процентах.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.energy.consumption[<deviceUUID>]

<br> Общее энергопотребление GPU-устройства в миллиджоулях (mJ) с момента последней перезагрузки драйвера.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Для получения дополнительной информации см. файл readme NVIDIA GPU plugin.

nvml.device.errors.memory[<deviceUUID>]

<br> Информация об ошибках памяти ECC на GPU-устройстве (исправленные, неисправленные).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.errors.register[<deviceUUID>]

<br> Информация об ошибках регистрации ECC на GPU-устройстве (исправленные, неисправленные).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.fan.speed.avg[<deviceUUID>]

<br> Средняя скорость вращения вентилятора устройства GPU в процентах от максимальной скорости.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme NVIDIA GPU plugin.

nvml.device.get

<br> Обнаруженные GPU с UUID и именем.<br> Возвращаемое значение: *JSON*.

Дополнительную информацию см. в файле readme плагина NVIDIA GPU.

nvml.device.graphics.frequency[<deviceUUID>]

<br> Тактовая частота графического процессора GPU в МГц.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Для получения дополнительной информации см. файл readme плагина NVIDIA GPU.

nvml.device.memory.fb.get[<deviceUUID>]

<br> Статистика памяти framebuffer GPU-устройства (общий объем, зарезервировано, свободно, используется).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme NVIDIA GPU plugin.

nvml.device.memory.bar1.get[<deviceUUID>]

<br> Статистика памяти BAR1 устройства GPU (всего, свободно, используется).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme NVIDIA GPU plugin.

nvml.device.memory.frequency[<deviceUUID>]

<br> Частота памяти GPU-устройства в МГц.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Для получения дополнительной информации см. файл readme NVIDIA GPU plugin.

nvml.device.pci.utilization[<deviceUUID>]

<br> Использование PCI GPU-устройства (пропускная способность передачи/приема в KBps).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле README плагина NVIDIA GPU.

nvml.device.performance.state[<deviceUUID>]

<br> Состояние производительности устройства GPU.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительную информацию см. в файле readme плагина NVIDIA GPU.

nvml.device.power.limit[<deviceUUID>]

<br> Ограничение мощности GPU-устройства в милливаттах.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.power.usage[<deviceUUID>]

<br> Текущее энергопотребление GPU-устройства в милливаттах.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.serial[<deviceUUID>]

<br> Серийный номер устройства GPU.<br> Возвращаемое значение: *String*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Для получения дополнительной информации см. файл readme плагина NVIDIA GPU.

nvml.device.sm.frequency[<deviceUUID>]

<br> Тактовая частота streaming multiprocessor GPU-устройства в МГц.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme NVIDIA GPU plugin.

nvml.device.temperature[<deviceUUID>]

<br> Температура GPU-устройства в градусах Цельсия.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.utilization[<deviceUUID>]

<br> Статистика загрузки устройства GPU (загрузка GPU/памяти в процентах).<br> Возвращаемое значение: *JSON*.

Параметр:

- **deviceUUID** - UUID устройства GPU.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.device.video.frequency[<deviceUUID>]

<br> Скорость видеочастоты GPU-устройства в МГц.<br> Возвращаемое значение: *Integer*.

Параметр:

- **deviceUUID** - UUID GPU-устройства.

Дополнительные сведения см. в файле readme плагина NVIDIA GPU.

nvml.system.driver.version

<br> Версия драйвера NVIDIA.<br> Возвращаемое значение: *String*.

Дополнительную информацию см. в файле readme NVIDIA GPU plugin.

nvml.version

<br> Версия библиотеки NVML.<br> Возвращаемое значение: *String*.

Дополнительную информацию см. в файле readme плагина NVIDIA GPU.

oracle.diskgroups.stats[connString,<user>,<password>,<service>,<diskgroup>]

<br> Возвращает статистику групп дисков Automatic Storage Management (ASM).<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle;<br>
- **diskgroup** - имя группы дисков ASM, для которой выполняется запрос.

oracle.diskgroups.discovery[connString,<user>,<password>,<service>]

<br> Возвращает список дисковых групп ASM. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.archive.info[connString,<user>,<password>,<service>,<destination>]

<br> Статистика архивных журналов.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет обработан как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>

    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из условий выше не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя сервиса Oracle;<br>
- **destination** - имя назначения, для которого выполняется запрос.

oracle.cdb.info[connString,<user>,<password>,<service>,<database>]

<br> Информация о контейнерных базах данных (CDB).<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае значение будет обрабатываться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя сервиса Oracle;<br>
- **database** - имя базы данных для запроса.

oracle.custom.query[connString,<user>,<password>,<service>,queryName,<args...>]

<br> Результат пользовательского запроса.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` задано значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет обработан как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается с Zabbix 7.4.3) - если для параметра ResolveTNS задано значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle;<br>
- **queryName** - имя пользовательского запроса, должно совпадать с именем SQL-файла без расширения;<br>
- **args** - один или несколько аргументов, разделенных запятыми, которые передаются в запрос.

Комментарии: - Возвращаемые данные автоматически преобразуются в JSON. - Не возвращайте JSON напрямую из запросов, так как при повторном преобразовании плагином в JSON он будет поврежден.

oracle.datafiles.stats[connString,<user>,<password>,<service>]

<br> Возвращает статистику файлов данных.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>

    - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя сервиса Oracle.

oracle.db.discovery[connString,<user>,<password>,<service>]

<br> Возвращает список баз данных. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае значение будет обработано как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.fra.stats[connString,<user>,<password>,<service>]

<br> Возвращает статистику Fast Recovery Area (FRA).<br> Возвращаемое значение: *JSON object*.

Параметры: - **connString** - может быть:<br> - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` задано значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и значение все равно будет обрабатываться как URI;<br> - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br> - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br> - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br> - **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br> - **password** - пароль Oracle;<br> - **service** - имя службы Oracle.<br>

oracle.instance.info[connString,<user>,<password>,<service>]

<br> Статистика экземпляра.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>

    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.pdb.info[connString,<user>,<password>,<service>,<database>]

<br> Информация о подключаемых базах данных (PDB).<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет обработан как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя сервиса Oracle;<br>
- **database** - имя базы данных для запроса.

oracle.pdb.discovery[connString,<user>,<password>,<service>]

<br> Возвращает список PDB. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться и в любом случае будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.pga.stats[connString,<user>,<password>,<service>]

<br> Возвращает статистику Program Global Area (PGA).<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет обработан как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>

- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.ping[connString,<user>,<password>,<service>]

<br> Проверяет, можно ли установить соединение с Oracle.<br> Возвращаемое значение: *1* - соединение успешно; *0* - соединение разорвано (если возникает любая ошибка, включая проблемы AUTH и конфигурации).

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

Примеры:

```
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>]

<br> Возвращает статистику процессов.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет интерпретирован как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.redolog.info[connString,<user>,<password>,<service>]

<br> Информация о файле журнала из файла управления.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с заданным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>

    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя сервиса Oracle.<br>

oracle.sga.stats[connString,<user>,<password>,<service>]

<br> Возвращает статистику System Global Area (SGA).<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя сервиса Oracle.<br>

oracle.sessions.stats[connString,<user>,<password>,<service>,<lockMaxTime>]

<br> Возвращает статистику сеансов.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и значение все равно будет обрабатываться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle;<br>
- **lockMaxTime** - максимальная длительность блокировки сеанса в секундах, при которой сеанс считается длительно заблокированным. По умолчанию: 600 секунд.

oracle.sys.metrics[connString,<user>,<password>,<service>,<duration>]

<br> Возвращает набор значений системных метрик.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>

    - ключ TNS (поддерживается с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle;<br>
- **duration** - интервал сбора (в секундах) значений системных метрик. Возможные значения: *60* — длительный интервал (по умолчанию), *15* — короткий интервал.

oracle.sys.params[connString,<user>,<password>,<service>]

<br> Возвращает набор значений системных параметров.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.<br>

oracle.ts.stats[connString,<user>,<password>,<service>,<tablespace>,<type>,<conname>]

<br> Возвращает статистику по табличным пространствам.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
    - URI - если сеанс с указанным именем не найден и `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
    - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
    - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
    - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle;<br>
- **tablespace** - имя табличного пространства, для которого выполняется запрос. Значение по умолчанию (если поле оставлено пустым и задан `type`):
    - "TEMP" (если `type` задан как "TEMPORARY");
    - "USERS" (если `type` задан как "PERMANENT").
- **type** - тип табличного пространства, для которого выполняется запрос. Значение по умолчанию (если задан `tablespace`): "PERMANENT".
- **conname** - имя контейнера, для которого требуется информация.

Если `tablespace`, `type` или `conname` не указаны, элемент данных вернет статистику по табличным пространствам для всех соответствующих контейнеров (включая PDB и CDB).

oracle.ts.discovery[connString,<user>,<password>,<service>]

<br> Возвращает список табличных пространств. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден, а `Plugins.Oracle.ResolveTNS` имеет значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS имеет значение true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.

oracle.user.info[connString,<user>,<password>,<service>,<username>]

<br> Возвращает информацию о пользователе Oracle.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае значение будет обработано как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle;<br>
- **username** - имя пользователя, для которого нужна информация. Имена пользователей в нижнем регистре не поддерживаются. По умолчанию: текущий пользователь.

oracle.version[connString,<user>,<password>,<service>]

<br> Возвращает версию сервера базы данных.<br> Возвращаемое значение: *String*.

Параметры:

- **connString** - может быть:<br>
  - URI - если сеанс с указанным именем не найден и для `Plugins.Oracle.ResolveTNS` установлено значение false. Если он содержит схему (например, "tcp://"), порт (например, 1521) или и то и другое, параметр ResolveTNS не будет учитываться, и в любом случае он будет трактоваться как URI;<br>
  - имя сеанса - если такое имя найдено в файле конфигурации плагина;<br>
  - ключ TNS (поддерживается начиная с Zabbix 7.4.3) - если параметр ResolveTNS установлен в true и ни одно из указанных выше условий не выполняется;<br>
  - значение TNS (поддерживается начиная с Zabbix 7.4.3) - если оно начинается с открывающей скобки "(" (начальные пробелы игнорируются). В этом случае параметр ResolveTNS не учитывается.<br>
- **user** - имя пользователя Oracle, поддерживает добавление одного из параметров входа `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` или `as sysrac` в формате `user as sysdba` (параметр входа не чувствителен к регистру, не должен содержать пробел в конце). Обратите внимание, что до Zabbix 7.4.2 поддерживаются только параметры входа `as sysdba`, `as sysoper` или `as sysasm`;<br>
- **password** - пароль Oracle;<br>
- **service** - имя службы Oracle.

pgsql.autovacuum.count[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Количество autovacuum worker.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.archive[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Информация об архивных файлах.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.bgwriter[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Общее количество контрольных точек (checkpoint) по кластеру базы данных, с разбивкой по типу контрольных точек.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.cache.hit[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Коэффициент успешности кэша буфера PostgreSQL.<br> Возвращаемое значение: *Число с плавающей точкой*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.connections[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Возвращает подключения по типу.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.custom.query[uri,<имя_пользователя>,<пароль>,имя_запроса,<аргументы...>]

<br> Возвращает результат пользовательского запроса.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_запроса** — имя пользовательского запроса, должно совпадать с именем sql файла без расширения;<br>
- **аргументы** — один или несколько аргументов (через запятую), которые будут переданы запросу.

pgsql.db.age[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Возраст самого старого FrozenXID в базе данных.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.db.bloating_tables[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Количество «распухших» (bloating) таблиц по каждой базе данных.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.db.discovery[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Список баз данных PostgreSQL. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.db.size[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Размер базы данных в байтах.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.dbstat[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Сбор статистики по каждой базе данных. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.dbstat.sum[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Обобщённые данные по всем базам данных в кластере.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.locks[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Информация о предоставленных блокировках по каждой базе данных. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.oldest.xid[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Возраст самого старого XID.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>

- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.ping[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Проверка, можно ли установить подключение.<br> Возвращаемое значение: *1* — подключение успешно; *0* — подключение разорвано (если имеется какая-либо ошибка, включая проблемы с аутентификацией или проблемы с настройками).

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.queries[uri,<имя_пользователя>,<пароль>,<имя_бд>,<период_времени>]

<br> Измерение времени выполнения запросов.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>
- **период_времени** — ограничение времени выполнения для подсчёта медленных запросов (должно быть целым положительным числом).

pgsql.replication.count[uri,<имя_пользователя>,<пароль>]

<br> Количество серверов в режиме ожидания.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.replication.process[uri,<имя_пользователя>,<пароль>]

<br> Отставание сброса, отставание записи и отставание replay по каждому процессу sender.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.replication.process.discovery[uri,<имя_пользователя>,<пароль>]

<br> Обнаружение имени процесса репликации.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.replication.recovery_role[uri,<имя_пользователя>,<пароль>]

<br> Состояние восстановления.<br> Возвращаемое значение: *0* — режим ведущего; *1* — восстановление ещё в процессе (режим ведомого).

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.replication.status[uri,<имя_пользователя>,<пароль>]

<br> Состояние репликации.<br> Возвращаемое значение: *0* — передача потока не работает; *1* — передача потока работает; *2* — режим ведущего.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.replication_lag.b[uri,<имя_пользователя>,<пароль>]

<br> Отставание репликации в байтах.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.replication_lag.sec[uri,<имя_пользователя>,<пароль>]

<br> Отставание репликации в секундах.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL.<br>

pgsql.uptime[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Время работы PostgreSQL в миллисекундах.<br> Возвращаемое значение: *Число с плавающей точкой*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.version[uri,<имя_пользователя>,<пароль>,<имя_бд>]

<br> Возвращает версию PostgreSQL.<br> Возвращаемое значение: *Строка*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

pgsql.wal.stat[uri,<имя_пользователя>,<пароль>,<имя_бд]

<br> Статистика WAL.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **uri** — URI или имя сессии;<br>
- **имя_пользователя, пароль** — учётные данные PostgreSQL;<br>
- **имя_бд** — имя базы данных (если опущено, используется значение Plugins.PostgreSQL.Default.Database из файла postgresql.conf).<br>

redis.config[строка_подкл,<пароль>,<шаблон>]

<br> Получение параметров конфигурации экземпляра Redis, которые соответствуют шаблону.<br> Возвращаемое значение: *JSON* — если использовался шаблон в стиле glob; одно значение — если шаблон не содержит какого-либо символа подстановки.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пароль** — пароль к Redis;<br>
- **шаблон** — шаблон в стиле glob (по умолчанию *).

Комментарий:

- Предупреждение о безопасности: `redis.config` выполняет команду Redis `CONFIG GET` и может возвращать конфиденциальные параметры конфигурации (например, `requirepass`, `masterauth`, настройки, связанные с TLS, и другие секреты). Использование шаблона по умолчанию (`*`) или широких шаблонов может раскрыть эти секреты любому, кто в Zabbix имеет разрешение на просмотр значений элементов данных.

redis.info[строка_подкл,<пароль>,<раздел>]

<br> Получение вывода команды INFO.<br> Возвращаемое значение: *JSON* — вывод преобразован в JSON.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>

- **пароль** — пароль к Redis;<br>
- **раздел** — раздел [en] информации (по умолчанию *default*).<br>

redis.ping[строка_подкл,<пароль>]

<br> Проверка, можно ли установить подключение.<br> Возвращаемое значение: *1* — подключение успешно; *0* — подключение разорвано (если имеется какая-либо ошибка, включая проблемы с аутентификацией или проблемы с настройками).

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пароль** — пароль к Redis.<br>

redis.slowlog.count[строка_подкл,<пароль>]

<br> Количество медленных записей в журнале с момента запуска Redis.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **строка_подкл** — URI или имя сессии;<br>
- **пароль** — пароль к Redis.<br>

smart.attribute.discovery

<br> Возвращает список S.M.A.R.T. атрибутов устройств.<br> Возвращаемое значение: *Объект JSON*.

Комментарии:

- Возвращаются следующие макросы и их значения: {#NAME}, {#DISKTYPE}, {#ID}, {#ATTRNAME}, {#THRESH};
- Поддерживаются типы дисков HDD, SSD и NVME. Диски могут быть одиночными или объединёнными в RAID. В случае RAID {#NAME} будет иметь дополнение, например: {”{#NAME}”: ”/dev/sda cciss,2”}.

smart.disk.discovery[<тип>]

<br> Возвращает список S.M.A.R.T. устройств.<br> Возвращаемое значение: *Объект JSON*.

Параметр:

- **тип** — указывает значение для сканирования дисков. Возможные значения: *id*, *name* (по умолчанию). В Windows *id* не поддерживается и при выборе будет возвращать ошибку.

Комментарии:

- Возвращаются следующие макросы и их значения: {#NAME}, {#DISKTYPE}, {#MODEL}, {#SN}, {#PATH}, {#ATTRIBUTES}, {#RAIDTYPE};
- Поддерживаются типы дисков HDD, SSD и NVME. Если диск не объединён в RAID, {#RAIDTYPE} будет пустым. В случае RAID {#NAME} будет иметь дополнение, например: {”{#NAME}”: ”/dev/sda cciss,2”}.

smart.disk.get[<путь>,<тип_raid>]

<br> Возвращает все доступные свойства S.M.A.R.T. устройств.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **путь** — путь к диску, в качестве значения можно использовать макрос {#PATH};<br>
- **тип_raid** — тип RAID, в качестве значения можно использовать макрос {#RAID}

Комментарии:

- Поддерживаются типы дисков HDD, SSD и NVME. Диски могут быть одиночными или объединёнными в RAID;<br>
- Данные включают версию smartctl и аргументы вызова, а также дополнительные поля:<br>*disk_name* — содержит имя с требуемым дополнением для обнаружения RAID, например: {”disk_name”: ”/dev/sda cciss,2”}<br>*disk_type* — содержит тип диска HDD, SSD или NVME, например: {”disk_type”: ”ssd”};<br>
- Если параметры не указаны, элемент данных вернёт информацию обо всех дисках.

systemd.unit.get[имя модуля,<интерфейс>]

<br> Возвращает все свойства модуля systemd.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **имя модуля** — имя модуля (вы, возможно, захотите использовать макрос {#UNIT.NAME} в прототипе элементов данных для обнаружения имени);<br>
- **интерфейс** — тип интерфейса модуля, возможные значения: *Unit* (по умолчанию), *Service*, *Socket*, *Device*, *Mount*, *Automount*, *Swap*, *Target*, *Path*.

Комментарии:

- Этот элемент данных поддерживается только на платформе Linux;
- LoadState, ActiveState и UnitFileState для интерфейса модуля возвращаются в виде текста и целого числа: `"ActiveState":{"state":1,"text":"active"}`.

systemd.unit.info[имя модуля,<свойство>,<интерфейс>]

<br> Информация о модуле systemd.<br> Возвращаемое значение: *Строка*.

Параметры:

- **имя модуля** — имя модуля (вы, возможно, захотите использовать макрос {#UNIT.NAME} в прототипе элементов данных для обнаружения имени);<br>
- **свойство** — имя свойства (например, ActiveState (по умолчанию), LoadState, Description);
- **интерфейс** — тип интерфейса (например, Unit (по умолчанию), Socket, Service).

Комментарии:

- Этот элемент данных поддерживается только на платформе Linux;
- Этот элемент данных позволяет получать определённые свойства с заданного типа интерфейса, которые описаны в dbus API [en].

Примеры:

```
systemd.unit.info["{#UNIT.NAME}"] #                          (     ,       ,      ,      ,      )
systemd.unit.info["{#UNIT.NAME}",LoadState] #                                        systemd
systemd.unit.info[mysqld.service,Id] #                      (mysqld.service)
systemd.unit.info[mysqld.service,Description] #                (MySQL Server)
systemd.unit.info[mysqld.service,ActiveEnterTimestamp] #              ,                        (156256
systemd.unit.info[dbus.socket,NConnections,Socket] #
```

systemd.unit.discovery[<тип>]

<br> Список модулей systemd и их подробностей. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметр:

- **тип** — возможные значения: *all*, *automount*, *device*, *mount*, *path*, *service* (по умолчанию), *socket*, *swap*, *target*.

Этот элемент данных поддерживается только на платформе Linux.

web.certificate.get[hostname,<port>,<address>]

<br> Проверяет сертификаты и возвращает сведения о сертификате.<br> Возвращаемое значение: *JSON object*.

Параметры:

- **hostname** - может быть IP или DNS.<br>Может содержать схему URL (*только https*), путь (он будет проигнорирован) и порт.<br>Если порт указан и в первом, и во втором параметрах, их значения должны совпадать.<br>Если указан address (3-й параметр), hostname используется только для SNI и проверки hostname;<br>
- **port** - номер порта (по умолчанию для HTTPS используется 443);<br>
- **address** - может быть либо IP, либо DNS. Если указан, он будет использоваться для подключения, а hostname (1-й параметр) будет использоваться для SNI и проверки узла сети. Если 1-й параметр является IP, а 3-й параметр - DNS, 1-й параметр будет использоваться для подключения, а 3-й параметр - для SNI и проверки узла сети.

Комментарии:

- Этот элемент данных станет неподдерживаемым, если назначение, указанное в конфигурации узла сети, не существует, недоступно или если TLS-handshake завершается с любой ошибкой, кроме недействительного сертификата;<br>
- В настоящее время не поддерживаются расширение X.509 AIA (Authority Information Access), CRL и OCSP (включая OCSP stapling), а также Certificate Transparency;
- Поля JSON-ответа:
  - *x509*: содержит сведения о сертификате X.509.
    - *version*: версия X.509 (например, 3).
    - *serial_number*: серийный номер сертификата.
    - *signature_algorithm*: алгоритм, использованный для подписи сертификата (например, SHA256-RSA).
    - *issuer*: издатель сертификата.
    - *not_before*: дата начала действия сертификата.
    - *not_after*: дата окончания действия сертификата.
    - *subject*: субъект сертификата.

        * *public_key_algorithm*: алгоритм, использованный для открытого ключа (например, RSA).
        * *alternative_names*: альтернативные имена субъекта (если присутствуют), иначе `null`.
      - result: содержит результат проверки.
        * *value*: статус проверки (см. возможные значения ниже).
        * *message*: подробное сообщение о проверке (например, `"certificate verified successfully"`).
    - *sha1_fingerprint*: отпечаток сертификата SHA-1.
    - *sha256_fingerprint*: отпечаток сертификата SHA-256.
- Поле `$.result.value` указывает результат проверки сертификата. Возможные значения:
    - *valid* - сертификат действителен и доверен.
    - *valid-but-self-signed* - сертификат действителен, но самоподписан, то есть его субъект совпадает с издателем.
    - *invalid* - сертификат недействителен из-за такой проблемы, как истечение срока действия, неверное имя узла сети или неизвестный центр подписи.

Пример:

```
web.certificate.get[example.com,443]
```

JSON Response:


```
{
  "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"
}
```


2. Zabbix агент для Windows

Обзор

Элементы данных агента Zabbix для Windows представлены в двух списках:

- Общие элементы данных — ключи элементов данных, которые являются общими с агентом Zabbix для UNIX;
- Элементы данных, специфичные для Windows — ключи элементов данных, которые поддерживаются **только** в Windows.

Обратите внимание, что все ключи элементов данных, поддерживаемые агентом Zabbix для Windows, также поддерживаются новым поколением Zabbix агента 2. Смотрите дополнительные ключи элементов данных, которые можно использовать только с агентом 2.

Смотрите также: Минимальные разрешения для элементов данных Windows.

Общие элементы данных

В таблице ниже перечислены элементы данных агента Zabbix, которые поддерживаются в Windows и являются общими с агентом Zabbix для UNIX:

- Ключ элемента данных является ссылкой на полные сведения об элементе данных агента Zabbix для UNIX
- Включены комментарии к элементам данных, относящиеся к Windows

| Item key | Description | Item group |
|---|---|---|
| log | Мониторинг файла журнала. Этот элемент данных не поддерживается для журнала событий Windows.<br>Параметр `persistent_dir` не поддерживается в Windows. | Мониторинг журналов |
| log.count | Количество совпавших строк в контролируемом файле журнала. Этот элемент данных не поддерживается для журнала событий Windows.<br>Параметр `persistent_dir` не поддерживается в Windows. | |
| logrt | Мониторинг ротируемого файла журнала. Этот элемент данных не поддерживается для журнала событий Windows.<br>Параметр `persistent_dir` не поддерживается в Windows. | |
| logrt.count | Количество совпавших строк в контролируемом ротируемом файле журнала. Этот элемент данных не поддерживается для журнала событий Windows.<br>Параметр `persistent_dir` не поддерживается в Windows. | |
| modbus.get | Читает данные Modbus. | Modbus |
| net.dns | Проверяет, доступна ли служба DNS.<br>Параметры `ip`, `timeout` и `count` в Windows игнорируются, если не используется Zabbix agent 2. | Сеть |
| net.dns.perf | Проверяет производительность службы DNS.<br>Параметры `ip`, `timeout` и `count` в Windows игнорируются, если не используется Zabbix agent 2. | |
| net.dns.record | Выполняет DNS-запрос.<br>Параметры `ip`, `timeout` и `count` в Windows игнорируются, если не используется Zabbix agent 2. | |
| net.if.discovery | Список сетевых интерфейсов.<br>В некоторых версиях Windows (например, Server 2008) для поддержки не-ASCII символов в именах интерфейсов могут потребоваться установленные последние обновления. | |
| net.if.in | Статистика входящего трафика на сетевом интерфейсе.<br>В Windows элемент данных получает значения из 64-битных счетчиков, если они доступны. 64-битные счетчики статистики интерфейсов были введены в Windows Vista и Windows Server 2008. Если 64-битные счетчики недоступны, агент использует 32-битные счетчики.<br>Имена интерфейсов с многобайтовыми символами в Windows поддерживаются.<br>Описания сетевых интерфейсов в Windows можно получить с помощью элементов данных net.if.discovery или net.if.list. | |
| net.if.out | Статистика исходящего трафика на сетевом интерфейсе.<br>В Windows элемент данных получает значения из 64-битных счетчиков, если они доступны. 64-битные счетчики статистики интерфейсов были введены в Windows Vista и Windows Server 2008. Если 64-битные счетчики недоступны, агент использует 32-битные счетчики.<br>Имена интерфейсов с многобайтовыми символами в Windows поддерживаются.<br>Описания сетевых интерфейсов в Windows можно получить с помощью элементов данных net.if.discovery или net.if.list. | |
| net.if.total | Сумма статистики входящего и исходящего трафика на сетевом интерфейсе.<br>В Windows элемент данных получает значения из 64-битных счетчиков, если они доступны. 64-битные счетчики статистики интерфейсов были введены в Windows Vista и Windows Server 2008. Если 64-битные счетчики недоступны, агент использует 32-битные счетчики.<br>Имена интерфейсов с многобайтовыми символами в Windows поддерживаются.<br>Описания сетевых интерфейсов в Windows можно получить с помощью элементов данных net.if.discovery или net.if.list. | |
| net.tcp.listen | Проверяет, находится ли этот TCP-порт в состоянии LISTEN. | |
| net.tcp.port | Проверяет, возможно ли установить TCP-соединение с указанным портом. | |

| Item key | Description | Item group |
|---|---|---|
| net.tcp.service | Проверяет, запущена ли служба и принимает ли она TCP-соединения.<br>Проверка LDAP и HTTPS в Windows поддерживается только Zabbix agent 2. | |
| net.tcp.service.perf | Проверяет производительность службы TCP.<br>Проверка LDAP и HTTPS в Windows поддерживается только Zabbix agent 2. | |
| net.tcp.socket.count | Возвращает количество TCP-сокетов, соответствующих параметрам.<br>Этот элемент данных поддерживается в Linux агентом Zabbix, но в Windows он поддерживается только Zabbix agent 2 в 64-битной Windows. | |
| net.udp.service | Проверяет, запущена ли служба и отвечает ли она на UDP-запросы. | |
| net.udp.service.perf | Проверяет производительность службы UDP. | |
| net.udp.socket.count | Возвращает количество UDP-сокетов, соответствующих параметрам.<br>Этот элемент данных поддерживается в Linux агентом Zabbix, но в Windows он поддерживается только Zabbix agent 2 в 64-битной Windows. | |
| proc.get | Список процессов ОС и их параметров.<br>Параметр `cmdline` не поддерживается в Windows. | Процессы |
| proc.num | Количество процессов.<br>В Windows поддерживаются только параметры `name` и `user`. | |
| system.cpu.discovery | Список обнаруженных CPU/ядер CPU. | Система |
| system.cpu.load | Нагрузка CPU.<br>При запуске процесса сборщика в агенте Zabbix инициализируются следующие счетчики производительности, которые затем используются для этого элемента данных: `\System\Processor Queue Length` | |
| system.cpu.num | Количество CPU. | |
| system.cpu.util | Процент загрузки CPU.<br>Значение получается с использованием счетчика производительности *Processor Time*. Обратите внимание, что начиная с Windows 8 Диспетчер задач показывает загрузку CPU на основе счетчика производительности *Processor Utility*, тогда как в предыдущих версиях использовался счетчик *Processor Time* (см. подробности).<br>*system* - единственный параметр `type`, поддерживаемый в Windows. | |
| system.hostname | Имя узла сети в системе.<br>Значение в Windows получается с помощью функций GetComputerName() (для **netbios**), GetComputerNameExA() (для **fqdn**) или gethostname() (для **host**).<br>См. также подробное описание. | |
| system.localtime | Системное время. | |
| system.run | Выполняет указанную команду на узле сети. | |
| system.sw.arch | Сведения об архитектуре программного обеспечения. | |
| system.sw.os | Сведения об операционной системе. | |
| system.sw.os.get | Подробные сведения об операционной системе (версия, тип, имя дистрибутива, минорная и мажорная версии и т. д.). | |
| system.swap.size | Размер пространства подкачки в байтах или в процентах от общего объема.<br>Параметр типа `pused` поддерживается в Linux агентом Zabbix, но в Windows он поддерживается только Zabbix agent 2.<br>Обратите внимание, что этот ключ может возвращать неверный размер/процент пространства подкачки на виртуализированных платформах Windows (VMware ESXi, VirtualBox). В этом случае можно использовать ключ `perf_counter[\700(_Total)\702]`, чтобы получить корректный процент пространства подкачки. | |
| system.uname | Идентификация системы.<br>В Windows значение этого элемента данных получается из классов WMI Win32_OperatingSystem и Win32_Processor. Имя ОС (включая редакцию) может быть переведено на язык отображения пользователя. В некоторых версиях Windows оно содержит символы товарных знаков и лишние пробелы. | |
| system.uptime | Время работы системы в секундах. | |
| vfs.dir.count | Количество записей каталога.<br>В Windows символьные ссылки на каталоги пропускаются, а жесткие ссылки учитываются только один раз. | Виртуальные файловые системы |
| vfs.dir.get | Список записей каталога.<br>В Windows символьные ссылки на каталоги пропускаются, а жесткие ссылки учитываются только один раз. | |
| vfs.dir.size | Размер каталога.<br>В Windows любая символьная ссылка пропускается, а жесткие ссылки учитываются только один раз. | |

| Item key | Description | Item group |
|---|---|---|
| vfs.file.cksum | Контрольная сумма файла, вычисленная по алгоритму UNIX cksum. | |
| vfs.file.contents | Получение содержимого файла. | |
| vfs.file.exists | Проверяет, существует ли файл.<br>В Windows при использовании утилиты командной строки для вызова zabbix_get.exe или agent2 двойные кавычки должны быть экранированы обратной косой чертой '\', а весь ключ элемента данных должен быть заключен в двойные кавычки.<br>Обратите внимание, что элемент данных может стать неподдерживаемым в Windows, если выполняется поиск каталога внутри несуществующего каталога, например `vfs.file.exists[C:\no\dir,dir]` (где 'no' не существует). | |
| vfs.file.get | Возвращает сведения о файле.<br>Поддерживаемые типы файлов в Windows: обычный файл, каталог, символическая ссылка | |
| vfs.file.md5sum | Контрольная сумма MD5 файла. | |
| vfs.file.owner | Получает владельца файла. | |
| vfs.file.regexp | Извлекает строку из файла. | |
| vfs.file.regmatch | Находит строку в файле. | |
| vfs.file.size | Размер файла. | |
| vfs.file.time | Сведения о времени файла.<br>В Windows XP `vfs.file.time[file,change]` может быть равен `vfs.file.time[file,access]`. | |
| vfs.fs.discovery | Список смонтированных файловых систем с их типом и параметрами монтирования.<br>Макрос {#FSLABEL} поддерживается в Windows. | |
| vfs.fs.get | Список смонтированных файловых систем с их типом, доступным дисковым пространством, статистикой inode и параметрами монтирования.<br>Макрос {#FSLABEL} поддерживается в Windows. | |
| vfs.fs.size | Дисковое пространство в байтах или в процентах от общего объема. | |
| vm.memory.size | Размер памяти в байтах или в процентах от общего объема. | Виртуальная память |
| web.page.get | Получает содержимое веб-страницы. | Мониторинг веб-сайтов |
| web.page.perf | Время загрузки полной веб-страницы. | |
| web.page.regexp | Находит строку на веб-странице. | |
| agent.hostmetadata | Метаданные узла сети агента. | Zabbix |
| agent.hostname | Имя узла сети агента. | |
| agent.ping | Проверка доступности агента. | |
| agent.variant | Вариант агента Zabbix (Zabbix agent или Zabbix agent 2). | |
| agent.version | Версия агента Zabbix. | |
| zabbix.stats | Удаленно возвращает набор внутренних метрик сервера Zabbix или прокси. | |
| zabbix.stats (queue) | Удаленно возвращает количество контролируемых элементов данных в очереди, задержанных на сервере Zabbix или прокси. | |

Элементы данных, специфичные для Windows

В таблице приведены сведения о ключах элементов данных, которые поддерживаются **только** Zabbix агентом для Windows.

Элементы данных, специфичные для Windows, иногда являются приблизительным аналогом аналогичного элемента данных агента; например, `proc_info`, поддерживаемый в Windows, примерно соответствует элементу данных `proc.mem`, который в Windows не поддерживается.

Ключ элемента данных является ссылкой на полную информацию о нём.

| Ключ элемента данных | Описание | Группа элементов данных |
|---|---|---|
| eventlog | Мониторинг журналов событий Windows. | Log monitoring |
| eventlog.count | Подсчёт строк в журнале событий Windows. | |

| Ключ элемента данных | Описание | Группа элементов данных |
|---|---|---|
| net.if.list | Список сетевых интерфейсов (включая тип, состояние, IPv4 адрес, описание интерфейса. | Network |
| perf_counter | Значение любого счётчика производительности Windows. | Performance counters |
| perf_counter_en | Значение любого счётчика производительности Windows на английском языке. | |
| perf_instance.discovery | Список экземпляров объекта счётчиков производительности Windows. | |
| perf_instance_en.discovery | Список экземпляров объекта счётчиков производительности Windows, обнаруженных с использованием имён объектов на английском языке. | |
| proc_info | Различная информация об указанном процессе(-ах). | Processes |
| registry.data | Получение данных для указанного имени значения в ключе Реестра Windows. | Registry |
| registry.get | Список значений или ключей Реестра Windows, расположенных в указанном ключе. | |
| service.discovery | Список служб Windows. | Services |
| service.info | Информация о службе. | |
| services | Список служб. | |
| vm.vmemory.size | Размер виртуального пространства в байтах или в процентах от общего размера. | Virtual memory |
| wmi.get | Выполнение WMI запроса и получение первого выбранного объекта. | WMI |
| wmi.getall | Выполнение WMI запроса и получение полного ответа. | |

Подробности ключей элементов данных

Параметры без угловых скобок обязательны. Параметры, обозначенные угловыми скобками **< >**, опциональны.

eventlog[name,<regexp>,<severity>,<source>,<eventid>,<maxlines>,<mode>]

<br> Мониторинг журнала событий.<br> Возвращаемое значение: *Log*.

Параметры:

- **name** - имя канала журнала событий (*Log Name* в графическом интерфейсе Event Viewer);<br>
- **regexp** - регулярное выражение, описывающее требуемый шаблон (с учетом регистра);<br>
- **severity** - регулярное выражение, описывающее уровень серьезности (без учета регистра). Этот параметр принимает регулярное выражение на основе следующих значений: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (при работе в Windows Vista или новее).<br>
- **source** - регулярное выражение, описывающее идентификатор источника (без учета регистра);<br>
- **eventid** - регулярное выражение, описывающее идентификатор(ы) события (с учетом регистра);<br>
- **maxlines** - максимальное количество новых строк в секунду, которое агент будет отправлять на сервер Zabbix или прокси. Этот параметр переопределяет значение ’MaxLinesPerSecond’ в zabbix_agentd.conf.<br>
- **mode** - возможные значения: *all* (по умолчанию) или *skip* - пропуск обработки более старых данных (влияет только на вновь созданные элементы данных).

Комментарии:

- Элемент данных должен быть настроен как активная проверка;
- Агент не может отправлять события из журнала ”Forwarded events”;
- Поддерживается Windows Eventing 6.0;
- Выбор для этого элемента данных не-Log типа информации приведет к потере локальной метки времени, а также сведений об уровне серьезности и источнике журнала;
- Дополнительные сведения см. также в разделе мониторинг журналов.

Примеры:

```
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] #here a custom regular expression named `TWOSHORT` is referenced (defined as
```

eventlog.count[name,<regexp>,<severity>,<source>,<eventid>,<maxproclines>,<mode>]

<br> Количество строк в журнале событий Windows.<br> Возвращаемое значение: *Integer*.

Параметры:

- **name** - имя канала журнала событий (*Log Name* в графическом интерфейсе Event Viewer);<br>
- **regexp** - регулярное выражение, описывающее требуемый шаблон (с учетом регистра);<br>
- **severity** - регулярное выражение, описывающее уровень серьезности (без учета регистра). Этот параметр принимает регулярное выражение на основе следующих значений: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (при работе в Windows Vista или более новой версии).<br>
- **source** - регулярное выражение, описывающее идентификатор источника (без учета регистра);<br>
- **eventid** - регулярное выражение, описывающее идентификатор(ы) события (с учетом регистра);<br>
- **maxproclines** - максимальное количество новых строк в секунду, которое агент будет анализировать (не может превышать 10000). Значение по умолчанию: 10*’MaxLinesPerSecond’ в zabbix_agentd.conf.<br>
- **mode** - возможные значения: *all* (по умолчанию) или *skip* - пропуск обработки более старых данных (влияет только на вновь созданные элементы данных).

Комментарии:

- Элемент данных должен быть настроен как активная проверка;
- Агент не может получать события из журнала ”Forwarded events”;
- Поддерживается Windows Eventing 6.0;
- Выбор для этого элемента данных не-Log типа информации приведет к потере локальной метки времени, а также сведений об уровне серьезности и источнике журнала;
- См. также дополнительную информацию о мониторинге журналов.

Примеры:

```
eventlog.count[System,,"Warning|Error"]
eventlog.count[Windows PowerShell,,,,,,skip]
```

net.if.list

<br> Список сетевых интерфейсов (включая тип, состояние, IPv4 адрес, описание интерфейса).<br> Возвращаемое значение: *Текст*.

Комментарии:

- Поддерживаются мультибайтовые имена интерфейса;
- Отключённые интерфейсы не входят в список;
- Включение/отключение некоторых компонентов Windows может изменить порядок имён интерфейсов в Windows;
- В некоторых версиях Windows (к примеру, Server 2008) может потребоваться установка последних обновлений для поддержки не-ASCII символов в именах интерфейсов.

perf_counter[счётчик,<период>]

<br> Значение произвольного счётчика производительности Windows.<br> Возвращаемое значение: *Целое число*, *число с плавающей точкой*, *строка* или *текст* (в зависимости от запроса).

Параметры:

- **счётчик** — путь к счётчику;<br>
- **период** — последние N секунд для сохранения усреднённого значения. Значение должно быть от 1 до 900 секунд (включительно), значение по умолчанию 1.

Комментарии:

- Этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Параметр используется для счётчиков, требующих более одной выборки (например, утилизация CPU); таким образом, проверка каждый раз возвращает среднее значение за последние «период» секунд;
- Можно использовать Мониторинг производительности (Performance Monitor) для получения списка доступных счётчиков.
- Смотрите также: Счётчики производительности в Windows.

perf_counter_en[счётчик,<период>]

<br> Значение произвольного счётчика производительности Windows на английском языке.<br> Возвращаемое значение: *Целое число*, *число с плавающей точкой*, *строка* или *текст* (в зависимости от запроса).

Параметры:

- **счётчик** — путь к счётчику на английском языке;<br>
- **период** — последние N секунд для сохранения усреднённого значения. Значение должно быть от 1 до 900 секунд (включительно), значение по умолчанию 1.

Комментарии:

- Этот элемент данных использует UNC-пути, что может представлять собой угрозу безопасности;
- Параметр  используется для счётчиков, требующих более одной выборки (например, утилизация CPU); таким образом, проверка каждый раз возвращает среднее значение за последние «период» секунд;
- Этот элемент данных поддерживается только на **Windows Server 2008/Vista** и более новых версиях;
- Вы можете найти список строк на английском языке, заглянув в следующую ветку реестра: `HKEY_LOCAL_MACHINE\SOFTWARE\M NT\CurrentVersion\Perflib\009`.

perf_instance.discovery[объект]

<br> Список экземпляров объекта счётчиков производительности Windows.  Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметр:

- **объект** — имя объекта (локализованное).

perf_instance_en.discovery[объект]

<br> Список экземпляров объекта счётчиков производительности Windows, обнаруженных с использованием имён объектов на английском языке. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*.

Параметр:

- **объект** — имя объекта (на английском языке).

proc_info[процесс,<атрибут>,<тип>]

<br> Различная информация об указанном процессе(-ах).<br> Возвращаемое значение: *Число с плавающей точкой*.

Параметры:

- **процесс** — имя процесса;<br>
- **атрибут** — запрашиваемый атрибут процесса;<br>
- **тип** — тип представления (имеет смысл, когда есть более одного процесса с одинаковым именем)

Комментарии:

- Поддерживаются следующие  :<br>*vmsize* (по умолчанию) — размер виртуальной памяти процесса в Кбайтах<br>*wkset* — размер working set процесса (количество используемой процессом физической памяти) в Кбайтах<br>*pf* — количество ошибок на страницах<br>*ktime* — время ядра процесса в миллисекундах<br>*utime* — пользовательское время процесса в миллисекундах<br>*io_read_b* — количество байт, прочитанных процессом во время операций ввода/вывода<br>*io_read_op* — количество операций чтения, выполненных процессом<br>*io_write_b* — количество байт, записанных процессом во время операций ввода/вывода<br>*io_write_op* — количество операций записи, выполненных процессом<br>*io_other_b* — количество байт, пересланных процессом во время операций, отличных от чтения и записи<br>*io_other_op* — количество выполненных процессом операций ввода/вывода, отличных от операций чтения и записи<br>*gdiobj* — количество используемых процессом объектов GDI<br>*userobj* — количество используемых процессом объектов USER;<br>
- Допустимые  :<br>*avg* (по умолчанию) — среднее значение среди всех процессов с именем <процесс><br>*min* — минимальное значение среди всех процессов с именем <процесс><br>*max* — максимальное значение среди всех процессов с именем <процесс><br>*sum* — сумма значений для всех процессов с именем <процесс>;
- Для корректной работы этого элемента данных на 64-битной системе требуется 64-битный Zabbix агент.<br>

Примеры:

```
proc_info[iexplore.exe,wkset,sum] #                    ,          Internet Explorer
proc_info[iexplore.exe,pf,avg] #                              Internet Explorer
```

registry.data[ключ,<имя значения>]

<br> Получение данных для указанного имени значения в ключе Реестра Windows.<br> Возвращаемое значение: *Целое число*, *строка* или *текст* (в зависимости от типа значения)

Параметры:

- **ключ** — ключ реестра, включая ключ корня; допускаются сокращения корней (например, HKLM);
- **имя значения** — имя значения реестра в ключе (по умолчанию — пустая строка ””). Если имя значения не указано, возвращается значение по умолчанию.

Комментарии:

- Поддерживаются сокращения корней:<br>HKCR — HKEY_CLASSES_ROOT<br>HKCC — HKEY_CURRENT_CONFIG<br>HKCU — HKEY_CURRENT_USER<br>HKCULS — HKEY_CURRENT_USER_LOCAL_SETTINGS<br>HKLM — HKEY_LOCAL_MACHINE<br>HKPD — HKEY_PERFORMANCE_DATA<br>HKPN — HKEY_PERFORMANCE_NLSTEXT<br>HKPT — HKEY_PERFORMANCE_TEXT<br>HKU — HKEY_USERS<br>
- Ключи с пробелами должны быть заключены в двойные кавычки.

Примеры:

```
registry.data["HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting"] #
registry.data["HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting","EnableZip"] #
```

registry.get[ключ,<режим>,<regexp имени>]

<br> Список значений или ключей Реестра Windows, расположенных в указанном ключе.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **ключ** — ключ реестра, включая корневой ключ; допускаются сокращения корней (например, HKLM; смотрите комментарии для registry.data[], чтобы увидеть полный список сокращений);<br>
- **режим** — возможные значения:<br>*values* (по умолчанию) или *keys*;<br>
- **regexp имени** — обнаружить только значения с именами, которые соответствуют регулярному выражению (по умолчанию — обнаружить все значения). Допустимо только когда имеет значение *values*.

Ключи с пробелами должны быть заключены в двойные кавычки.

Примеры:

```
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values,"^DisplayName|DisplayVersion$
JSON                    ,           ,        ,                    .
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values] #                    .
JSON                    ,           ,        ,                    .
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,keys] #                  .
JSON                                .
```

service.discovery

<br> Список служб Windows. Используется для низкоуровневого обнаружения.<br> Возвращаемое значение: *объект JSON*.

service.info[служба,<парам>]

<br> Информация о службе.<br> Возвращаемое значение: *Целое число* — с равным *state*, *startup*; *Строка* — с равным *displayname*, *path*, *user*; *Текст* — с равным *description*<br>Конкретно для *state*: 0 — запущена, 1 — пауза, 2 — ожидание старта, 3 — ожидание паузы, 4 — ожидание продолжения, 5 — ожидание остановки, 6 — остановлена, 7 — неизвестно, 255 — такой службы не существует<br>Конкретно для *startup*: 0 — автоматически, 1 — автоматически (отложенный запуск), 2 — вручную, 3 — отключена, 4 — неизвестно, 5 — автоматический запуск по триггеру, 6 — автоматический отложенный запуск по триггеру, 7 — ручной запуск по триггеру

Параметры:

- **служба** — фактическое имя службы или её отображаемое имя как в оснастке MMC Службы;
- **парам** — *state* (по умолчанию), *displayname*, *path*, *user*, *startup* или *description*.

Комментарии:

- Элементы данных service.info[ ,state] и service.info[ ] вернут одинаковую информацию;
- Только равный *state* у этого элемента данных возвращает значение по несуществующим службам (255).

Примеры:

```
service.info[SNMPTRAP] —           SNMPTRAP;
service.info[SNMP Trap] —               ,                ;
service.info[EventLog,startup] —
```

services[<тип>,<состояние>,<исключения>]

<br> Список служб.<br> Возвращаемое значение: *0* — если список служб пуст; *Текст* — список служб, разделенных новой строкой.

Параметры:

- **тип** — *all* (по умолчанию), *automatic*, *manual* или *disabled*;

- **состояние** — *all* (по умолчанию), *stopped*, *started*, *start_pending*, *stop_pending*, *running*, *continue_pending*, *pause_pending* или *paused*;
- **исключения** — список служб, исключаемых из результата. Исключаемые службы должны быть указаны в двойных кавычках, разделены запятой, без пробелов.

Примеры:

```
services[,started] #                    ;
services[automatic, stopped] #                      ,                ;
services[automatic, stopped, "service1,service2,service3"] #                      ,                ,
```

vm.vmemory.size[<тип>]

<br> Размер виртуальной памяти в байтах или в процентах от общего размера.<br> Возвращаемое значение: *Целое число* — для байт; *Число с плавающей точкой* — для процентов.

Параметр:

- **тип** — возможные значения: *available* (доступно виртуальной памяти), *pavailable* (доступно виртуальной памяти, в процентах), *pused* (использовано виртуальной памяти, в процентах), *total* (всего виртуальной памяти, по умолчанию) или *used* (использовано виртуальной памяти)

Комментарии:

- Мониторинг статистики виртуальной памяти основывается на:<br>
  - Общем количестве виртуальной памяти в Windows (всего физической + размер файла подкачки);<br>
  - Максимальном количестве памяти, которое может занять Zabbix агент;<br>
  - Текущем лимите выделенной памяти системе или Zabbix агенту, смотря что меньше.

Пример:

```
vm.vmemory.size[pavailable] #                      ,
```

wmi.get[<пространство_имён>,<запрос>]

<br> Выполнение WMI запроса и получение первого выбранного объекта.<br> Возвращаемое значение: *Целое число*, *число с плавающей точкой*, *строка* или *текст* (в зависимости от запроса).

Параметры:

- **пространство_имён** — название пространства имён WMI;<br>
- **запрос** — WMI запрос, возвращающий один объект.

WMI запросы выполняются с использованием WQL [en].

Пример:

```
wmi.get[root\cimv2,select status from Win32_DiskDrive where Name like '%PHYSICALDRIVE0%'] #
```

wmi.getall[<пространство_имён>,<запрос>]

<br> Выполнение WMI запроса и получение полного ответа. Можно использовать для низкоуровневого обнаружения.<br> Возвращаемое значение: *Объект JSON*

Параметры:

- **пространство_имён** — название пространства имён WMI;<br>
- **запрос** — WMI запрос.

Комментарии:

- Запросы WMI выполняются при помощи WQL [en].
- Можно использовать JSONPath предобработку при обращении к более конкретным значениям в полученном JSON.

Пример:

```
wmi.getall[root\cimv2,select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'] #
```

Мониторинг служб Windows

Это руководство содержит пошаговые инструкции по настройке мониторинга служб Windows. Предполагается, что Zabbix сервер и агент уже настроены и работают.

Шаг 1

Узнайте имя службы.

Вы можете получить имя службы, перейдя в оснастку MMC Службы и открыв свойства службы. На вкладке *Общие* вы должны увидеть поле, называемое «Имя службы». Его значение и будет тем именем, которое вы будете использовать при настройке элемента данных для наблюдения. Например, если вы хотите наблюдать службу «workstation», то ваша служба, вероятно, будет: **lanmanworkstation**.

Шаг 2

Настройте элемент данных для наблюдения за службой.

Элемент данных `service.info[ ,< >]` возвращает информацию о конкретной службе. В зависимости от требуемой вам информации, укажите опцию , которая принимает следующие значения: *displayname*, *state*, *path*, *user*, *startup* или *description*. Значением по умолчанию является *state*, если не указан (`service.info[ ]`).

Тип возвращаемого значения зависит от выбранного : целое число при *state* и *startup*; строка символов при *displayname*, *path* и *user*; текст при *description*.

Пример:

- *Ключ*: `service.info[lanmanworkstation]`
- *Тип информации*: Числовой (целое положительное)

Элемент данных `service.info[lanmanworkstation]` будет извлекать информацию о состоянии службы как числовое значение. Чтобы преобразовать числовое значение в текстовое представление в веб-интерфейсе («0» как «Running», «1» как «Paused» и т.д.), вы можете настроить преобразование значений на узле сети, на котором сконфигурирован элемент данных. Чтобы это сделать, либо присоедините шаблон *Windows services by Zabbix agent* или *Windows services by Zabbix agent active* к узлу сети, либо настройте на узле сети новое преобразование значений, основанное на преобразованиях значений *Windows service state*, уже настроенных в упомянутых шаблонах.

Обратите внимание, что оба упомянутых шаблона имеют правило обнаружения, настроенное для автоматического обнаружения служб. Если вы не хотите этого, вы можете деактивировать правило обнаружения на уровне узла сети после присоединения к нему шаблона.

Обнаружение служб Windows

Низкоуровневое обнаружение даёт возможность автоматического создания элементов данных, триггеров и графиков по различных объектам на компьютере. Zabbix может автоматически начать наблюдение за службами Windows на вашей машине, без необходимости знания точного имени службы или создания элементов данных по каждой службе вручную. Можно использовать фильтр для генерирования реальных элементов данных, триггеров и графиков только по интересующим службам.

3 Мониторинг файлов журнала

Обзор

Zabbix можно использовать для централизованного мониторинга и анализа файлов журналов с поддержкой ротации журналов или без нее.

Уведомления можно использовать, чтобы предупреждать пользователей, когда файл журнала содержит определенные строки или шаблоны строк.

Чтобы отслеживать файл журнала, у вас должны быть:

- запущен агент Zabbix на узле сети
- настроен элемент данных для мониторинга журнала

**Attention:**
Ограничение размера контролируемого файла журнала зависит от поддержки больших файлов.

Конфигурация

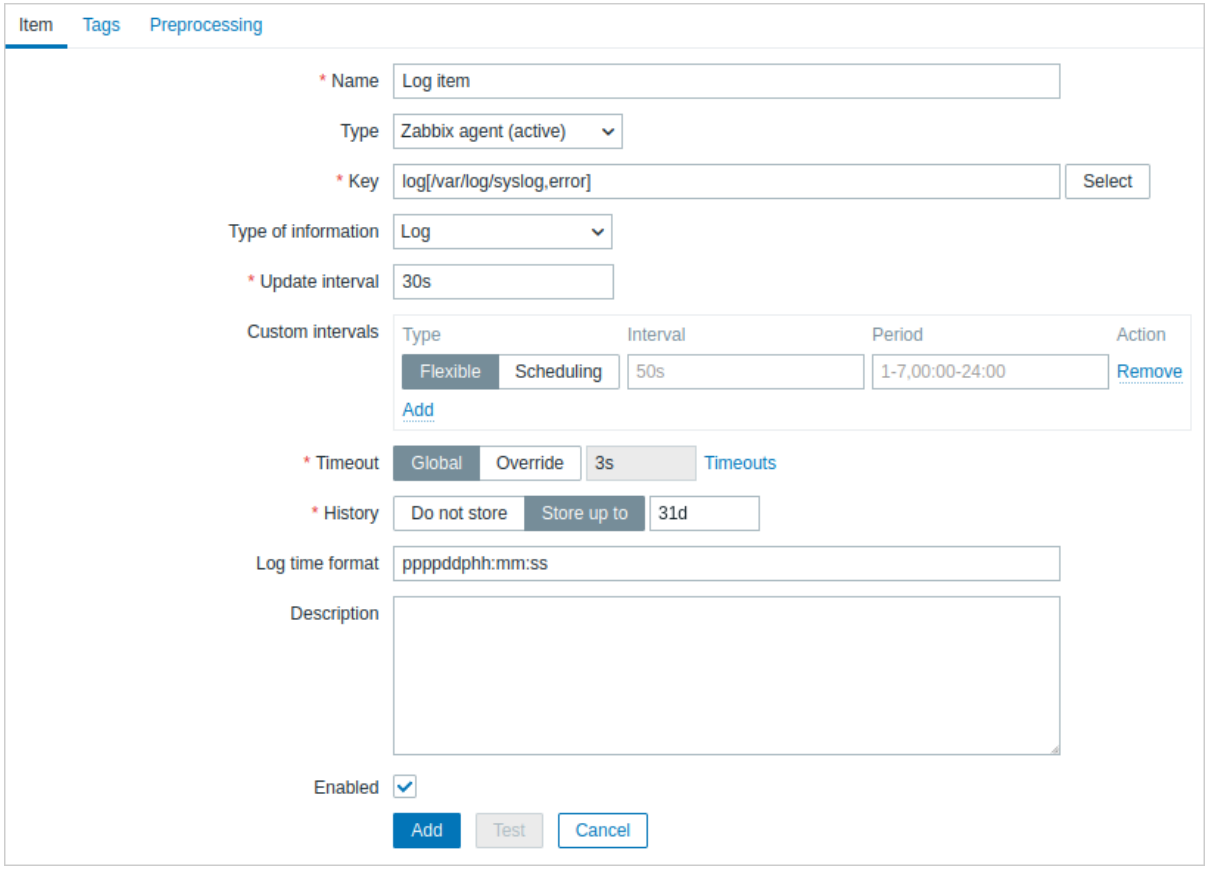
Проверьте параметры агента

Убедитесь, что в файле конфигурации агента:

- параметр `Hostname` совпадает с именем узла сети во веб-интерфейсе.
- серверы в параметре `ServerActive` указаны для обработки активных проверок.

Конфигурация элемента данных

Настройте элемент данных для мониторинга логов.




Все обязательные поля ввода отмечены красной звездочкой.

Для элементов данных мониторинга логов укажите:

| | |
|---|---|
| *Type* | Выберите здесь **Zabbix agent (active)**. |
| *Key* | Используйте один из следующих ключей элемента данных:<br>**log[]** или **logrt[]**:<br>Эти два ключа элемента данных позволяют мониторить логи и фильтровать записи журнала по регулярному выражению в содержимом, если оно указано.<br>Например: `log[/var/log/syslog,error]`. Убедитесь, что у файла есть права на чтение для пользователя 'zabbix', иначе статус элемента данных будет установлен в 'unsupported'.<br>**log.count[]** или **logrt.count[]**:<br>Эти два ключа элемента данных позволяют возвращать только количество совпадающих строк.<br>См. раздел поддерживаемых ключей элемента данных агента Zabbix для подробностей об использовании этих ключей элемента данных и их параметров. |
| *Type of information* | Заполняется автоматически:<br>Для элементов данных `log[]` или `logrt[]` - `Log`;<br>Для элементов данных `log.count[]` или `logrt.count[]` - `Numeric (unsigned)`.<br>Если дополнительно используется параметр `output`, вы можете вручную выбрать подходящий тип информации, отличный от `Log`.<br>Обратите внимание, что выбор типа информации, отличного от `Log`, приведет к потере локальной временной метки. |
| *Update interval (in sec)* | Этот параметр определяет, как часто агент Zabbix будет проверять наличие изменений в файле журнала. Установка значения 1 секунда обеспечит получение новых записей как можно быстрее. |

| | |
|---|---|
| *Log time format* | В этом поле вы можете дополнительно указать шаблон для разбора временной метки строки лога. Поддерживаемые заполнители:<br>* **y**: *Год (1970-2038)*<br>* **M**: *Месяц (01-12)*<br>* **d**: *День (01-31)*<br>* **h**: *Час (00-23)*<br>* **m**: *Минута (00-59)*<br>* **s**: *Секунда (00-59)*<br>Если оставить поле пустым, временная метка будет установлена в 0 по времени Unix, что соответствует 1 января 1970 года.<br>Например, рассмотрим следующую строку из файла журнала агента Zabbix:<br>” 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).”<br>Она начинается с шести символов для PID, затем идут дата, время и остальная часть сообщения.<br>Формат времени лога для этой строки будет ”ppppp:yyyyMMdd:hhmmss”.<br>Обратите внимание, что символы ”p” и ”:” являются заполнителями и могут быть любыми символами, кроме ”yMdhms”. |

Важные замечания

- Сервер и агент ведут учет размера отслеживаемого журнала и времени его последнего изменения (для logrt) в двух счетчиках. Дополнительно:
  - Агент также внутренне использует номера inode (в UNIX/GNU/Linux), индексы файлов (в Microsoft Windows) и MD5-суммы первых 512 байт файла журнала для улучшения решений, когда журналы обрезаются и ротируются.
  - В системах UNIX/GNU/Linux предполагается, что файловые системы, где хранятся файлы журналов, сообщают номера inode, которые можно использовать для отслеживания файлов.
  - В Microsoft Windows агент Zabbix определяет тип файловой системы, на которой находятся файлы журналов, и использует:
    * На файловых системах NTFS 64-битные индексы файлов.
    * На файловых системах ReFS (только начиная с Microsoft Windows Server 2012) 128-битные идентификаторы файлов.
    * На файловых системах, где индексы файлов изменяются (например, FAT32, exFAT), используется запасной алгоритм, чтобы принимать разумное решение в неопределенных условиях, когда ротация файла журнала приводит к появлению нескольких файлов журнала с одинаковым временем последнего изменения.
  - Номера inode, индексы файлов и MD5-суммы внутренне собираются агентом Zabbix. Они не передаются на сервер Zabbix и теряются при остановке агента Zabbix.
  - Не изменяйте время последнего изменения файла журнала (например, с помощью `touch`) и не заменяйте отслеживаемый файл журнала путем копирования файла обратно под его исходным именем (это создает новый inode). В любом из этих случаев Zabbix может считать файл другим файлом и перечитать его с начала, что может привести к дублированию оповещений.
  - Если для элемента `logrt[]` есть несколько совпадающих файлов журнала и агент Zabbix отслеживает самый последний из них, а этот самый последний файл журнала удаляется, в журнале записывается предупреждение `"there are no files matching "<regexp mask>" in "<directory>"`. Агент Zabbix игнорирует файлы журнала со временем изменения меньше, чем самое позднее время изменения, увиденное агентом для проверяемого элемента `logrt[]`.
- Агент начинает чтение файла журнала с того места, на котором остановился в прошлый раз.
- Количество уже проанализированных байт (счетчик размера) и время последнего изменения (счетчик времени) хранятся в базе данных Zabbix и передаются агенту, чтобы гарантировать, что агент начнет чтение файла журнала с этого места в случаях, когда агент только что запущен или получил элементы, которые ранее были отключены или не поддерживались. Однако если агент получил от сервера ненулевой счетчик размера, но элемент `logrt[]` или `logrt.count[]` не может найти совпадающие файлы, счетчик размера сбрасывается в 0, чтобы анализировать с начала, если файлы появятся позже.
- Каждый раз, когда файл журнала становится меньше, чем известный агенту счетчик размера журнала, счетчик сбрасывается в ноль, и агент начинает чтение файла журнала с начала, учитывая счетчик времени.
- Если в каталоге есть несколько совпадающих файлов с одинаковым временем последнего изменения, агент пытается корректно проанализировать все файлы журнала с одинаковым временем изменения и избежать пропуска данных или повторного анализа одних и тех же данных, хотя это не может быть гарантировано во всех ситуациях. Агент не предполагает какой-либо конкретной схемы ротации файлов журнала и не определяет ее. При наличии нескольких файлов журнала с одинаковым временем последнего изменения агент будет обрабатывать их в лексикографически убывающем порядке. Таким образом, для некоторых схем ротации файлы журнала будут анализироваться и сообщаться в их исходном порядке. Для других схем ротации исходный порядок

файлов журнала не будет соблюдаться, что может привести к сообщению записей совпавших файлов журнала в измененном порядке (проблема не возникает, если файлы журнала имеют разное время последнего изменения).

- Агент Zabbix обрабатывает новые записи файла журнала один раз в *Update interval* секунд.
- Агент Zabbix не отправляет более **maxlines** строк файла журнала в секунду. Это ограничение предотвращает перегрузку сетевых и процессорных ресурсов и переопределяет значение по умолчанию, заданное параметром **MaxLinesPerSecond** в файле конфигурации агента.
- Чтобы найти требуемую строку, Zabbix обработает в 10 раз больше новых строк, чем задано в MaxLinesPerSecond. Таким образом, например, если элемент `log[]` или `logrt[]` имеет *Update interval* 1 секунду, по умолчанию агент проанализирует не более 200 записей файла журнала и отправит не более 20 совпавших записей на сервер Zabbix за одну проверку. При увеличении **MaxLinesPerSecond** в файле конфигурации агента или задании параметра **maxlines** в ключе элемента предел можно увеличить до 10000 проанализированных записей файла журнала и 1000 совпавших записей, отправленных на сервер Zabbix за одну проверку. Если *Update interval* установлен в 2 секунды, пределы для одной проверки будут в 2 раза выше, чем при *Update interval* 1 секунда.
- Кроме того, значения log и log.count всегда ограничены 50% размера буфера отправки агента, даже если в нем нет других значений, не относящихся к журналам. Поэтому для отправки значений **maxlines** за одно соединение (а не за несколько) параметр BufferSize агента должен быть не меньше maxlines x 2. Агент Zabbix может загружать данные во время сбора журнала и тем самым освобождать буфер, тогда как агент Zabbix 2 остановит сбор журнала до тех пор, пока данные не будут загружены и буфер не освободится; это выполняется асинхронно.
- При отсутствии элементов журнала весь размер буфера агента используется для значений, не относящихся к журналам. Когда поступают значения журнала, они при необходимости заменяют более старые значения, не относящиеся к журналам, вплоть до выделенных 50%.
- Для записей файла журнала длиннее 256kB только первые 256kB сопоставляются с регулярным выражением, а остальная часть записи игнорируется. Однако если агент Zabbix остановлен во время обработки длинной записи, внутреннее состояние агента теряется, и длинная запись может быть проанализирована снова и иначе после повторного запуска агента.
- Особое примечание для разделителей пути "\": если file_format равен "file\.log", то каталога "file" быть не должно, поскольку невозможно однозначно определить, экранирована ли "." или это первый символ имени файла.
- Регулярные выражения для `logrt` поддерживаются только в имени файла, сопоставление регулярного выражения для каталога не поддерживается.
- На платформах UNIX элемент `logrt[]` становится NOTSUPPORTED, если каталог, в котором ожидается наличие файлов журнала, не существует.
- В Microsoft Windows, если каталог не существует, элемент не станет NOTSUPPORTED (например, если каталог указан с ошибкой в ключе элемента).
- Отсутствие файлов журнала для элемента `logrt[]` не делает его NOTSUPPORTED. Ошибки чтения файлов журнала для элемента `logrt[]` записываются как предупреждения в файл журнала агента Zabbix, но не делают элемент NOTSUPPORTED.
- Файл журнала агента Zabbix может помочь выяснить, почему элемент `log[]` или `logrt[]` стал NOTSUPPORTED. Zabbix может отслеживать файл журнала своего агента, кроме случаев, когда установлено DebugLevel=4 или DebugLevel=5.
- Поиск знака вопроса с помощью регулярного выражения, например `\?`, может приводить к ложноположительным результатам, если текстовый файл содержит символы NUL, поскольку они заменяются на "?" Zabbix для продолжения обработки строки до символа новой строки.

Извлечение совпадающей части регулярного выражения

Иногда нам может понадобиться извлечь из целевого файла только нужное значение вместо возврата всей строки, когда найдено совпадение с регулярным выражением.

Элементы данных типа log позволяют извлекать нужные значения из совпавших строк. Это реализуется с помощью дополнительного параметра **output** в элементах данных `log` и `logrt`.

Использование параметра `output` позволяет указать "захватывающую группу" совпадения, которая может быть нам интересна.

Например,

```
log[/path/to/the/file,"large result buffer allocation.*Entries: ([0-9]+)",,,,\1]
```

должно позволить возвращать количество записей, найденное в содержимом:

```
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
```

Будет возвращено только число, потому что **\1** относится к первой и единственной захватывающей группе: **([0-9]+)**.

А благодаря возможности извлекать и возвращать число это значение можно использовать для определения триггеров.

Использование параметра maxdelay

Параметр `maxdelay` в элементах данных типа log позволяет игнорировать некоторые более старые строки из файлов журнала, чтобы анализировались самые последние строки в пределах `maxdelay` секунд.

> **Warning:**
> Указание `maxdelay` > 0 может привести к **игнорированию важных записей файла журнала и пропущенным оповещениям**. Используйте его осторожно и на свой страх и риск, только при необходимости.

По умолчанию элементы данных для мониторинга журналов обрабатывают все новые строки, появляющиеся в файлах журнала. Однако существуют приложения, которые в некоторых ситуациях начинают записывать огромное количество сообщений в свои файлы журнала. Например, если база данных или DNS-сервер недоступны, такие приложения заполняют файлы журнала тысячами почти одинаковых сообщений об ошибках, пока нормальная работа не будет восстановлена. По умолчанию все эти сообщения будут добросовестно проанализированы, а совпадающие строки отправлены на сервер, как настроено в элементах данных `log` и `logrt`.

Встроенная защита от перегрузки состоит из настраиваемого параметра `maxlines` (защищает сервер от слишком большого количества входящих совпадающих строк журнала) и ограничения `10\*'maxlines'` (защищает CPU и I/O узла сети от перегрузки агентом за одну проверку). Тем не менее, у встроенной защиты есть 2 проблемы. Во-первых, на сервер поступает большое количество потенциально малоинформативных сообщений, которые занимают место в базе данных. Во-вторых, из-за ограниченного числа строк, анализируемых в секунду, агент может отставать от самых новых записей журнала на часы. Скорее всего, вы предпочтете раньше получать информацию о текущей ситуации в файлах журнала, а не разбирать старые записи в течение нескольких часов.

Решение обеих проблем — использование параметра `maxdelay`. Если указано `maxdelay` > 0, при каждой проверке измеряются количество обработанных байтов, количество оставшихся байтов и время обработки. На основе этих значений агент вычисляет оценочную задержку — сколько секунд потребуется, чтобы проанализировать все оставшиеся записи в файле журнала.

Если задержка не превышает `maxdelay`, агент продолжает анализ файла журнала как обычно.

Если задержка больше `maxdelay`, агент **игнорирует фрагмент файла журнала, "перепрыгивая" через него** к новой оценочной позиции, чтобы оставшиеся строки можно было проанализировать в пределах `maxdelay` секунд.

Обратите внимание, что агент даже не читает игнорируемые строки в буфер, а вычисляет приблизительную позицию, к которой нужно перейти в файле.

Факт пропуска строк файла журнала записывается в файл журнала агента следующим образом:

```
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
```

Значение "to byte" является приблизительным, потому что после "перепрыгивания" агент корректирует позицию в файле до начала строки журнала, которая может оказаться дальше в файле или раньше.

В зависимости от того, как скорость роста журнала соотносится со скоростью его анализа, вы можете не увидеть "перепрыгиваний", увидеть редкие или частые "перепрыгивания", большие или маленькие "перепрыгивания" или даже небольшое "перепрыгивание" при каждой проверке. Колебания нагрузки на систему и задержки сети также влияют на вычисление задержки и, следовательно, на "перепрыгивание" вперед, чтобы успевать за параметром `maxdelay`.

Не рекомендуется задавать `maxdelay` < `update interval` (это может привести к частым небольшим "перепрыгиваниям").

Примечания по обработке ротации лог-файлов с `copytruncate`

`logrt` с опцией `copytruncate` предполагает, что разные лог-файлы содержат разные записи (по крайней мере, их временные метки различаются), поэтому MD5-суммы начальных блоков (до первых 512 байт) будут разными. Два файла с одинаковыми MD5-суммами начальных блоков означают, что один из них является исходным, а другой — копией.

`logrt` с опцией `copytruncate` предпринимает усилия, чтобы корректно обрабатывать копии лог-файлов без сообщения о дубликатах. Однако такие ситуации, как создание нескольких копий лог-файла с одинаковой временной меткой, ротация лог-файлов чаще, чем интервал обновления элемента данных `logrt\[\]`, а также частые перезапуски агента, не рекомендуются. Агент пытается обрабатывать все эти ситуации достаточно корректно, но хорошие результаты не могут быть гарантированы во всех случаях.

Примечания о постоянных файлах для элементов данных log*[]

Назначение постоянных файлов

Когда агент Zabbix запускается, он получает список активных проверок от сервера Zabbix или прокси. Для метрик log*[] он получает обработанный размер журнала и время изменения, чтобы определить, с какого места начинать

мониторинг файла журнала. В зависимости от фактического размера файла журнала и времени изменения, сообщаемых файловой системой, агент решает либо продолжить мониторинг файла журнала с обработанного размера, либо повторно проанализировать файл журнала с самого начала.

Работающий агент поддерживает более широкий набор атрибутов для отслеживания всех контролируемых файлов журналов между проверками. Это состояние в памяти теряется при остановке агента.

Новый необязательный параметр **persistent_dir** задает каталог для сохранения этого состояния элемента данных log[], log.count[], logrt[] или logrt.count[] в файл. Состояние элемента данных log восстанавливается из постоянного файла после перезапуска агента Zabbix.

Основной сценарий использования — мониторинг файла журнала, расположенного на зеркальной файловой системе. До определенного момента времени файл журнала записывается на оба зеркала. Затем зеркала разделяются. На активной копии файл журнала по-прежнему растет, получая новые записи. Агент Zabbix анализирует его и отправляет серверу обработанный размер журнала и время изменения. На пассивной копии файл журнала остается без изменений, значительно отставая от активной копии. Позже операционная система и агент Zabbix перезагружаются с пассивной копии. Обработанный размер журнала и время изменения, которые агент Zabbix получает от сервера, могут быть недействительными для ситуации на пассивной копии. Чтобы продолжить мониторинг файла журнала с того места, где агент остановился в момент разделения зеркала файловой системы, агент восстанавливает свое состояние из постоянного файла.

Работа агента с постоянным файлом

При запуске агент Zabbix ничего не знает о постоянных файлах. Только после получения списка активных проверок от сервера Zabbix (прокси) агент видит, что для некоторых элементов данных журнала следует использовать постоянные файлы в указанных каталогах.

Во время работы агента постоянные файлы открываются для записи (с помощью fopen(filename, ”w”)) и перезаписываются последними данными. Вероятность потери данных постоянного файла, если перезапись и разделение зеркала файловой системы происходят одновременно, очень мала, поэтому специальная обработка не требуется. Запись в постоянный файл НЕ сопровождается принудительной синхронизацией с носителем данных (fsync() не вызывается).

Перезапись последними данными выполняется после успешной передачи серверу Zabbix соответствующей записи файла журнала или метаданных (обработанный размер журнала и время изменения). Это может происходить так часто, как при каждой проверке элемента данных, если файл журнала продолжает изменяться.

При завершении работы агента никаких специальных действий не выполняется.

После получения списка активных проверок агент помечает устаревшие постоянные файлы на удаление. Постоянный файл становится устаревшим, если:

1. Соответствующий элемент данных журнала больше не отслеживается.
2. Элемент данных журнала перенастраивается с другим расположением **persistent_dir**, чем раньше.

Удаление выполняется с задержкой 24 часа, потому что файлы журнала в состоянии NOTSUPPORTED не включаются в список активных проверок, но позже они могут стать SUPPORTED, и их постоянные файлы будут полезны.

Если агент остановлен до истечения 24 часов, устаревшие файлы не будут удалены, так как агент Zabbix больше не получает от сервера Zabbix информацию об их расположении.

> **Warning:**
> Если перенастроить **persistent_dir** элемента данных журнала обратно на старое расположение **persistent_dir** в то время, когда агент остановлен, и при этом пользователь не удалит старый постоянный файл, это приведет к восстановлению состояния агента из старого постоянного файла, что вызовет пропуск сообщений или ложные оповещения.

Именование и расположение постоянных файлов

Агент Zabbix различает активные проверки по их ключам. Например, logrt[/home/zabbix/test.log] и logrt[/home/zabbix/test.log,] являются разными элементами данных. Изменение элемента данных logrt[/home/zabbix/test.log„,10] во веб-интерфейсе на logrt[/home/zabbix/test.log„,20] приведет к удалению элемента данных logrt[/home/zabbix/test.log„,10] из списка активных проверок агента и созданию элемента данных logrt[/home/zabbix/test.log„,20] (некоторые атрибуты переносятся при изменении во веб-интерфейсе/сервере, но не в агенте).

Имя файла состоит из контрольной суммы MD5 ключа элемента данных с добавлением длины ключа, чтобы уменьшить вероятность коллизий. Например, состояние элемента данных logrt[/home/zabbix50/test.log„„„„/home/zabbix50/agent_private] будет храниться в постоянном файле c963ade4008054813bbc0a650bb8e09266.

Несколько log-элементов данных могут использовать одно и то же значение **persistent_dir**.

**persistent_dir** задается с учетом особенностей структуры файловой системы, точек монтирования и параметров монтирования, а также конфигурации зеркалирования хранилища - постоянный файл должен находиться на той же зеркалируемой файловой системе, что и контролируемый log-файл.

Если каталог **persistent_dir** не может быть создан или не существует, либо права доступа для агента Zabbix не позволяют создавать/записывать/читать/удалять файлы, log-элемент данных становится NOTSUPPORTED.

Если права доступа к файлам постоянного хранилища удаляются во время работы агента или возникают другие ошибки (например, заполнен диск), то ошибки записываются в файл журнала агента, но элемент данных журнала не переходит в состояние NOTSUPPORTED.

Нагрузка на I/O

Постоянный файл элемента данных обновляется после успешной отправки каждого пакета данных (содержащего данные элемента) на сервер. Например, значение `BufferSize` по умолчанию равно 100. Если для элемента журнала найдено 70 совпадающих записей, то первые 50 записей будут отправлены в одном пакете, постоянный файл будет обновлен, затем оставшиеся 20 записей будут отправлены (возможно, с некоторой задержкой, когда накопится больше данных) во втором пакете, и постоянный файл будет обновлен снова.

Действия при сбое связи между агентом и сервером

Каждая совпадающая строка из элемента данных `log[]` и `logrt[]`, а также результат каждой проверки элемента данных `log.count[]` и `logrt.count[]` требует свободного слота в выделенной 50% области буфера отправки агента. Элементы буфера регулярно отправляются на сервер (или прокси), после чего слоты буфера снова становятся свободными.

Пока в выделенной области журнала в буфере отправки агента есть свободные слоты и связь между агентом и сервером (или прокси) отсутствует, результаты мониторинга журнала накапливаются в буфере отправки. Это помогает смягчить последствия кратковременных сбоев связи.

При более длительных сбоях связи все слоты журнала заполняются, и выполняются следующие действия:

- Проверки элементов данных `log[]` и `logrt[]` останавливаются. Когда связь восстанавливается и в буфере доступны свободные слоты, проверки возобновляются с предыдущей позиции. Ни одна совпадающая строка не теряется, она просто будет сообщена позже.
- Проверки `log.count[]` и `logrt.count[]` останавливаются, если `maxdelay = 0` (по умолчанию). Поведение аналогично элементам `log[]` и `logrt[]`, как описано выше. Обратите внимание, что это может повлиять на результаты `log.count[]` и `logrt.count[]`: например, одна проверка подсчитывает 100 совпадающих строк в файле журнала, но поскольку в буфере нет свободных слотов, проверка останавливается. Когда связь восстанавливается, агент подсчитывает те же 100 совпадающих строк, а также 70 новых совпадающих строк. Теперь агент отправляет count = 170, как если бы они были найдены в одной проверке.
- Проверки `log.count[]` и `logrt.count[]` с `maxdelay > 0`: если во время проверки не было ”jump”, поведение аналогично описанному выше. Если произошел ”jump” через строки файла журнала, то позиция после ”jump” сохраняется, а подсчитанный результат отбрасывается. Таким образом, агент старается не отставать от растущего файла журнала даже при сбое связи.

Обработка ошибок компиляции и выполнения регулярных выражений

Если регулярное выражение, используемое в элементе данных `log[]`, `logrt[]`, `log.count[]` или `logrt.count[]`, не может быть скомпилировано библиотекой PCRE или PCRE2, элемент данных переходит в состояние NOTSUPPORTED с сообщением об ошибке.
Чтобы продолжить мониторинг элемента данных журнала, необходимо исправить регулярное выражение.

Если регулярное выражение успешно компилируется, но дает сбой во время выполнения (для некоторых или всех записей журнала), то элемент данных журнала остается поддерживаемым, и мониторинг продолжается.
Ошибка выполнения записывается в файл журнала агента Zabbix (без записи из файла журнала).

Частота записи ограничена одной ошибкой выполнения на одну проверку, чтобы позволить агенту Zabbix контролировать собственный файл журнала.
Например, если анализируются 10 записей и 3 записи завершаются ошибкой выполнения regexp, в журнале агента будет создана одна запись.

Исключение: если `MaxLinesPerSecond=1` и интервал обновления равен 1 (за одну проверку разрешено анализировать только 1 запись), то ошибки выполнения regexp не записываются.

zabbix_agentd записывает ключ элемента данных в случае ошибки выполнения, а zabbix_agent2 записывает ID элемента данных, чтобы помочь определить, у какого элемента данных журнала возникают ошибки выполнения.
При возникновении ошибок выполнения рекомендуется переработать регулярное выражение.

**2 Простая проверка**

Обзор

Простые проверки в основном используются для удалённых безагентных проверок сервисов.

Обратите внимание, что для простых проверок Zabbix агент не требуется. За обработку простых проверок (создание внешних подключений и т.д.) отвечает Zabbix сервер/прокси.

Примеры использования простых проверок:

```
net.tcp.service[ftp,,155]
net.tcp.service[http]
net.tcp.service.perf[http,,8080]
net.udp.service.perf[ntp]
```

**Note:**
Поля *Имя пользователя* и *Пароль* (ограниченные 255 символами) в конфигурации элементов данных простых проверок используются для элементов данных мониторинга VMware; в остальных случаях игнорируются.

Поддерживаемые проверки

Ключи элементов данных перечислены без необязательных параметров и дополнительной информации. Щелкните по ключу элемента данных, чтобы увидеть полные сведения.

См. также ключи элементов данных мониторинга VMware.

| Ключ элемента данных | Описание |
|---|---|
| icmpping | Доступность узла сети по ICMP ping. |
| icmppingloss | Процент потерянных пакетов. |
| icmppingretry | Доступность узла сети по ICMP ping с повторными попытками. |
| icmppingsec | Время отклика ICMP ping. |
| net.tcp.service | Проверяет, запущена ли служба и принимает ли она TCP-соединения. |
| net.tcp.service.perf | Проверяет производительность службы TCP. |
| net.udp.service | Проверяет, запущена ли служба и отвечает ли она на UDP-запросы. |
| net.udp.service.perf | Проверяет производительность службы UDP. |

Подробности ключей элементов данных

Параметры без угловых скобок обязательны. Параметры, обозначенные угловыми скобками **< >**, опциональны.

icmpping[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

<br> Доступность узла сети по ICMP ping.<br> Возвращаемое значение: *0* - ICMP ping не удался; *1* - ICMP ping выполнен успешно.

Параметры:

- **target** - IP-адрес или DNS-имя узла сети;
- **packets** - количество пакетов;
- **interval** - время между последовательными пакетами в миллисекундах;
- **size** - размер пакета в байтах;
- **timeout** - время ожидания в миллисекундах;
- **options** - используется для разрешения перенаправления: если пусто (значение по умолчанию), ответы с перенаправлением считаются признаком недоступности целевого узла сети; если задано *allow_redirect*, ответы с перенаправлением считаются признаком доступности целевого узла сети.

См. также таблицу значений по умолчанию.

Комментарий:

- Если возвращен хотя бы один из пакетов, заданных параметром `packets`, элемент данных возвращает `1`. Если ни один из пакетов не возвращается, элемент данных возвращает `0`.

Пример:

```
icmpping[,4] #                          ,              1.
```

icmppingloss[<цель>,<пакеты>,<интервал>,<размер>,<время ожидания>,<опции>]

<br> Процентное отношение потерянных пакетов.<br> Возвращаемое значение: *Число с плавающей точкой*.

Параметры:

- **цель** — IP или DNS имя хоста;
- **пакеты** — количество пакетов;
- **интервал** — время между последовательными пакетами в миллисекундах;
- **размер** — размер пакета в байтах;
- **время ожидания** — время ожидания в миллисекундах;
- **опции** — используется для разрешения перенаправления: если пусто (значение по умолчанию), перенаправленные ответы рассматриваются как целевой хост выключен; если установлено значение *allow_redirect*, перенаправленные ответы рассматриваются как целевой хост включён.

Смотрите также таблицу значений по умолчанию.

icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]

<br> Доступность узла сети по ICMP ping с повторными попытками. Если первый пакет проходит успешно, остановиться; если пакет не проходит, повторять попытки до достижения максимального числа попыток, заданного в параметре `retries`. Этот элемент данных полезен для уменьшения количества пакетов, отправляемых по сети.<br> Возвращаемое значение: *0* - ICMP ping неудачен; *1* - ICMP ping успешен.

Параметры:

- **target** - IP-адрес или DNS-имя узла сети;
- **retries** - количество повторных попыток ping для цели, не включая первую попытку (0 или больше; по умолчанию 1);
- **backoff** - множитель, на который увеличивается время ожидания при каждом последующем запросе (диапазон 1.0-5.0; по умолчанию 1.0);
- **size** - размер пакета в байтах;
- **timeout** - время ожидания в миллисекундах;
- **options** - используется для разрешения перенаправления: если пусто (значение по умолчанию), ответы с перенаправлением считаются признаком недоступности узла сети; если задано *allow_redirect*, ответы с перенаправлением считаются признаком доступности узла сети.

См. также таблицу значений по умолчанию.

icmppingsec[<цель>,<пакеты>,<интервал>,<размер>,<время ожидания>,<режим>,<опции>]

<br> Время ответа на пинг по ICMP (в секундах).<br> Возвращаемое значение: *Число с плавающей точкой*.

Параметры:

- **цель** — IP или DNS имя хоста;
- **пакеты** — количество пакетов;
- **интервал** — время между последовательными пакетами в миллисекундах;
- **размер** — размер пакета в байтах;
- **время ожидания** — время ожидания в миллисекундах;
- **режим** — один из min, max или avg (по умолчанию);
- **опции** — используется для разрешения перенаправления: если пусто (значение по умолчанию), перенаправленные ответы рассматриваются как целевой хост выключен; если установлено значение *allow_redirect*, перенаправленные ответы рассматриваются как целевой хост включён.

Комментарии:

- Потерянные пакеты либо пакеты с превышенным временем ожидания при вычислениях не учитываются;
- Если хост недоступен (превышено время ожидания), элемент данных вернёт 0;
- Если возвращаемое значение меньше 0.0001 секунд, то оно будет выставлено в 0.0001 секунд;
- Смотрите также таблицу значений по умолчанию.

net.tcp.service[сервис,<ip>,<порт>]

<br> Проверка того, что сервис запущен и принимает TCP подключения.<br> Возвращаемое значение: *0* — сервис недоступен; *1* — сервис работает.

Параметры:

- **сервис** — возможные значения: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (смотрите подробности);
- **ip** — IP или DNS имя хоста (по умолчанию используется IP/DNS узла сети);
- **порт** — номер порта (по умолчанию используется стандартный для сервиса номер порта).

Комментарии:

- Обратите внимание, для сервиса *tcp* обязательно нужно указывать порт;

- Эти проверки могут привести к дополнительным записям в файлах журналов системных демонов (обычно журналируются сессии SMTP и SSH);
- Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. Как обходной вариант решения, пожалуйста, используйте `net.tcp.service[tcp,<ip>,  ]` для подобных проверок.

Пример:

```
net.tcp.service[ftp,,45] #                                    FTP        45     TCP
```

**Attention:**
Если SELinux работает в принудительном режиме (enforced mode), пользовательские простые проверки TCP/UDP могут быть заблокированы политикой. Чтобы проверить и разрешить новое исходящее соединение, просмотрите отказы аудита: `grep denied /var/log/audit/audit.log`

net.tcp.service.perf[сервис,<ip>,<порт>]

<br> Проверка производительности TCP сервиса.<br> Возвращаемое значение: *Число с плавающей точкой*: *0.000000* — сервис недоступен; *секунды* — количество секунд, потребовавшихся для подключения к сервису.

Параметры:

- **сервис** — возможные значения: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (смотрите подробности);
- **ip** — IP или DNS имя (по умолчанию используется IP/DNS узла сети);
- **порт** — номер порта (по умолчанию используется стандартный для сервиса номер порта).

Комментарии:

- Обратите внимание, для сервиса *tcp* обязательно нужно указывать порт;
- Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. Как обходной вариант решения, пожалуйста, используйте `net.tcp.service[tcp,<ip>,  ]` для подобных проверок.

Пример:

```
net.tcp.service.perf[ssh] #                                          SSH      .
```

net.udp.service[сервис,<ip>,<порт>]

<br> Проверка, запущен ли сервис и отвечает ли на UDP запросы.<br> Возвращаемое значение: *0* — сервис недоступен; *1* — сервис работает.

Параметры:

- **сервис** — возможные значения: *ntp* (смотрите подробности);
- **ip** — IP адрес или DNS имя (по умолчанию используется IP/DNS узла сети);
- **порт** — номер порта (по умолчанию используется стандартный для сервиса номер порта).

Пример:

```
net.udp.service[ntp,,45] #                                     NTP        45     UDP.
```

net.udp.service.perf[сервис,<ip>,<порт>]

<br> Проверка производительности UDP сервиса.<br> Возвращаемое значение: *Число с плавающей точкой*: *0.000000* — сервис недоступен; *секунды* — количество секунд, потребовавшихся на ожидание ответа от сервиса.

Параметры:

- **сервис** — возможные значения: *ntp* (смотрите подробности);
- **ip** — IP адрес или DNS имя (по умолчанию используется IP/DNS узла сети);
- **порт** — номер порта (по умолчанию используется стандартный для сервиса номер порта).

Пример:

```
net.udp.service.perf[ntp] #                                          NTP      .
```

**Attention:**
Для поддержки SourceIP в простых проверках LDAP (например в `net.tcp.service[ldap]`) требуется OpenLDAP версии 2.6.1 или новее.

Обработка тайм-аута

Zabbix не будет обрабатывать простой проверочный запрос дольше, чем количество секунд, заданное в поле *Timeout* в форме настройки элемента данных. Для элементов VMware items Zabbix не будет обрабатывать простой проверочный запрос дольше, чем количество секунд `Timeout`, заданное в файле конфигурации Zabbix сервер или прокси. Для элементов icmpping* значения тайм-аута и числа повторов указываются непосредственно в ключе элемента данных и не зависят от глобального параметра `Timeout`. Убедитесь, что эти значения настроены надлежащим образом в ключе элемента данных. Обратите внимание, что для элементов icmpping* применяется жестко заданное максимальное возможное значение тайм-аута 600 секунд (начиная с Zabbix 7.4.6).

ICMP-пинги

Zabbix использует внешнюю утилиту **fping** для обработки ICMP-пингов (**icmpping**, **icmppingloss**, **icmppingretry**, **icmppingsec**).

Установка

fping не входит в состав Zabbix и должен быть установлен отдельно:

- На различных Unix-платформах пакет fping доступен в стандартных репозиториях, но не установлен по умолчанию. В этом случае вы можете использовать менеджер пакетов для установки fping.
- Zabbix предоставляет пакеты fping для RHEL и его производных. Обратите внимание, что эти пакеты предоставляются без официальной поддержки.
- fping также можно собрать из исходного кода.

Конфигурация

Укажите расположение fping в параметре *FpingLocation* файла конфигурации сервера/прокси Zabbix (или в параметре *Fping6Location* для использования адресов IPv6).

fping должен быть доступен для выполнения пользователем, от имени которого запущен сервер/прокси Zabbix, и у этого пользователя должны быть достаточные права.

См. также: Известные проблемы при обработке простых проверок с версиями fping ниже 3.10.

Значения по умолчанию

Значения по умолчанию, ограничения и описание значений для параметров проверки ICMP:

| Параметр | Единица | Описание | Флаг fping | Значение по умолчанию задается | | Допустимые пределы по Zabbix | |
|---|---|---|---|---|---|---|---|
| | | | | **fping** | **Zabbix** | **min** | **max** |
| packets | число | Количество пакетов запроса, отправляемых целевому узлу | -C | | 3 | 1 | 10000 |
| interval | миллисекунды | Время ожидания между последовательными пакетами для отдельного целевого узла | -p | 1000 | | 20 | без ограничений |
| size | байты | Размер пакета в байтах<br>56 байт на x86, 68 байт на x86_64 | -b | 56 or 68 | | 24 | 65507 |
| timeout | миллисекунды | **fping v3.x** - время ожидания после отправки последнего пакета, зависит от флага *-C*<br>**fping v4.x** - индивидуальное время ожидания для каждого пакета | -t | **fping v3.x** - 500<br>**fping v4.x** и новее - наследуется от флага *-p*, но не более 2000 | | 50 | без ограничений |
| retries | число | Количество попыток выполнения ping для целевого узла, не включая первую попытку | -r | 3 | 1 | 0 | без ограничений |
| backoff factor | число | Коэффициент, на который умножается время ожидания при каждом последующем запросе | -B | 1.5 | 1.0 | 1.0 | 5.0 |

Значения по умолчанию могут немного отличаться в зависимости от платформы и версии.

Кроме того, Zabbix использует параметры fping *-i interval ms* (не путайте с параметром элемента данных *interval*, упомянутым в таблице выше, который соответствует параметру fping *-p*) и *-S source IP address* (или *-I* в более старых версиях fping). Эти параметры определяются автоматически путем выполнения проверок с разными комбинациями параметров. Zabbix пытается определить минимальное значение в миллисекундах, которое fping позволяет использовать с *-i*, пробуя 3 значения: 0, 1 и 10. Значение, которое впервые срабатывает успешно, затем используется для последующих проверок ICMP. Этот процесс выполняется каждым процессом ICMP pinger отдельно.

Автоматически определенные параметры fping становятся недействительными каждый час и определяются заново при следующей попытке выполнить проверку ICMP. Установите DebugLevel>=4, чтобы просмотреть подробности этого процесса в файле журнала сервера или прокси.

Zabbix записывает IP-адреса, которые нужно проверить с помощью любого из ключей *icmpping**, во временный файл, который затем передается fping. Если элементы данных имеют разные параметры ключа, в один файл записываются только те, у которых параметры ключа идентичны. Все IP-адреса, записанные в один файл, будут проверяться fping параллельно, поэтому процесс Zabbix ICMP pinger будет затрачивать фиксированное количество времени независимо от числа IP-адресов в файле.

1. Ключи элементов данных для мониторинга VMware

Список ключей элементов данных для мониторинга VMware перенесён в раздел Мониторинг VMware.

**3 SNMP агент**

Обзор

Возможно, вы захотите использовать мониторинг SNMP на таких устройствах, как принтеры, сетевые коммутаторы, маршрутизаторы или ИБП, которые обычно поддерживают SNMP, и на которых было бы непрактично пытаться устанавливать полноценные операционные системы и агенты Zabbix.

Чтобы иметь возможность получать данные, предоставляемые SNMP-агентами на этих устройствах, сервер Zabbix должен быть изначально настроен с поддержкой SNMP путем указания флага `--with-net-snmp`. Также рекомендуется установить файлы MIB, чтобы значения элементов данных отображались в правильном формате. Без файлов MIB могут возникать проблемы с форматированием, например отображение значений в HEX вместо UTF-8 или наоборот.

Проверки SNMP выполняются только по протоколу UDP.

Демоны сервера Zabbix и прокси записывают в журнал строки, подобные следующей, если получают некорректный ответ SNMP:

```
SNMP response from host "gateway" does not contain all of the requested variable bindings
```

Хотя они не охватывают все проблемные случаи, они полезны для выявления отдельных SNMP-устройств, для которых следует отключить объединенные запросы.

Сервер/прокси Zabbix будет повторять попытку до 5 раз для элементов данных SNMP `walk` и `get`. Механизм повторных попыток не применяется к сбоям разрешения DNS.

Для устаревших SNMP-проверок (один OID-номер или строка) сервер/прокси Zabbix выполнит как минимум одну повторную попытку после неудачного запроса: либо через механизм повторов библиотеки SNMP, либо через внутренний механизм объединенной обработки.

**Warning:**
Если вы мониторите устройства SNMPv3, убедитесь, что msgAuthoritativeEngineID (также известный как snmpEngineID или "Engine ID") никогда не используется двумя устройствами одновременно. Согласно RFC 2571 (раздел 3.1.1.1) он должен быть уникальным для каждого устройства.

**Warning:**
RFC3414 требует, чтобы устройства SNMPv3 сохраняли свои engineBoots. Некоторые устройства этого не делают, из-за чего их сообщения SNMP после перезагрузки считаются устаревшими и отклоняются. В такой ситуации кэш SNMP необходимо вручную очистить на сервере/прокси (с помощью -R snmp_cache_reload) либо перезапустить сервер/прокси.

Настройка мониторинга SNMP

Чтобы начать мониторинг устройства через SNMP, необходимо выполнить следующие шаги:

Шаг 1

Узнайте строку SNMP (или OID) элемента данных, который вы хотите отслеживать.

Для получения списка строк SNMP используйте команду **snmpwalk** (часть программного обеспечения net-snmp [en], которое вы должны были установить как часть инсталляции Zabbix) или эквивалентную утилиту:

```
snmpwalk -v 2c -c public <IP    > .
```

«2c» здесь означает версию SNMP, вы также можете заменить его на «1», чтобы использовать на устройстве SNMP версии 1.

Эта команда должна показать вам список SNMP строк и их последние значения. Если это не произойдёт, то возможно, что SNMP «community» отличается от стандартного «public», в этом случае вам необходимо узнать это имя.

Вы можете пройтись по списку, пока не найдёте строку, которую вы хотите отслеживать; например, если вы хотите мониторить количество входящих байт на 3 порту вашего коммутатора, вы могли бы использовать `IF-MIB::ifInOctets.3` из этой строки:

```
IF-MIB::ifHCInOctets.3 = Counter64: 3409739121
```

Теперь вы можете воспользоваться командой **snmpget**, чтобы определить цифровой OID для «IF-MIB::ifInOctets.3»:

```
snmpget -v 2c -c public -On <IP    > IF-MIB::ifHCInOctets.3
```

Обратите внимание, что последнее число в строке — это номер порта, который вы хотите отслеживать. Смотрите также: Динамические индексы.

Вывод команды покажет вам что-то наподобие этого:

```
.1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 3472126941
```

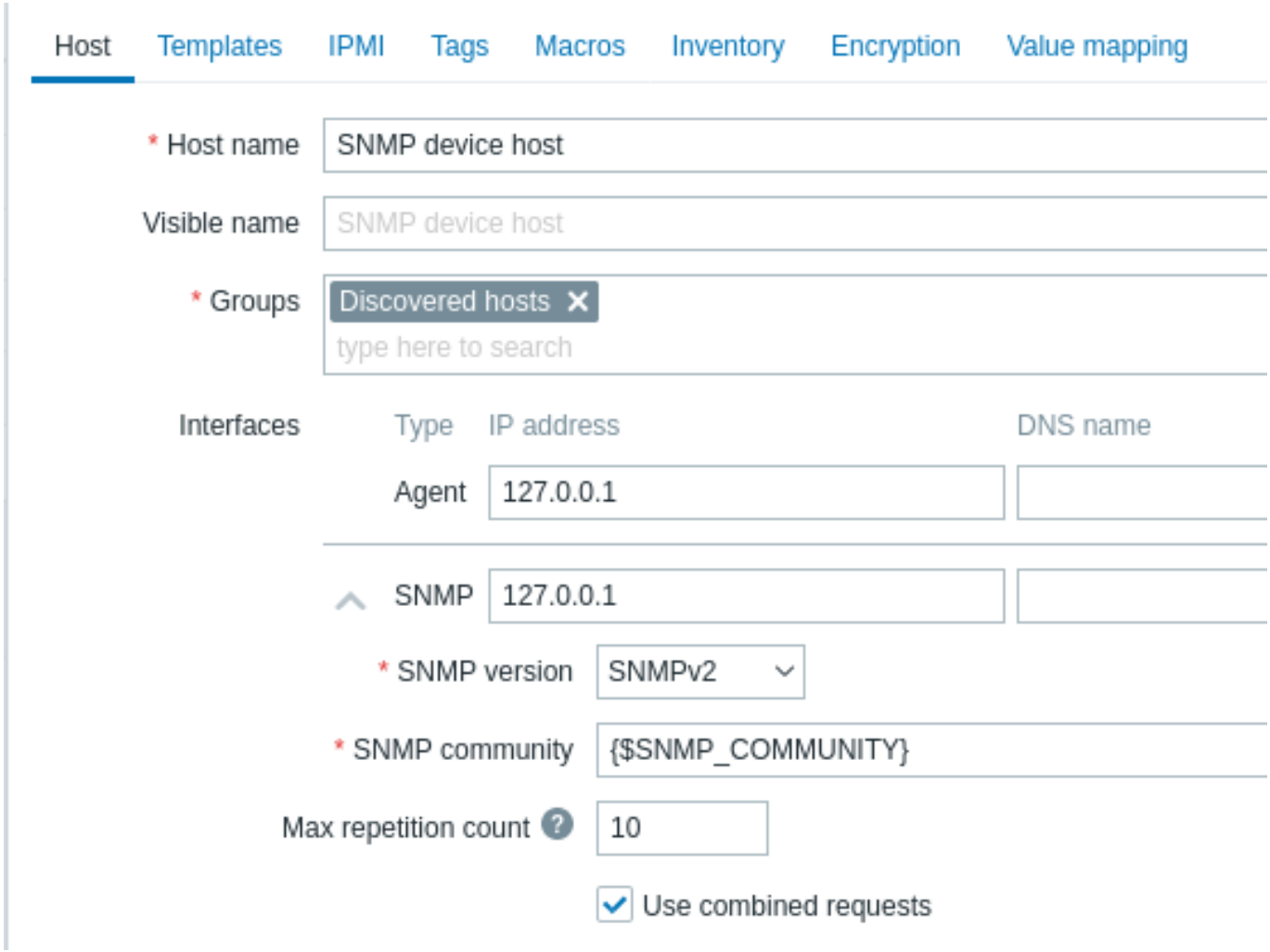
Опять же, последнее число в OID является номером порта.

**Note:**
Некоторые из наиболее часто используемых SNMP OID'ов автоматически конвертируются Zabbix'ом в числовое представление.

В последнем примере выше тип значения — «Counter64» (64-битный счетчик), что внутренне соответствует типу ASN_COUNTER64. Полный список поддерживаемых типов: 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 и ASN_OBJECT_ID (с версий 2.2.8, 2.4.3). Приведённые типы примерно соответствуют «Counter32», «Counter64», «UInteger32», «INTEGER», «Float», «Double», «Timeticks», «Gauge32», «IpAddress», «OCTET STRING», «OBJECT IDENTIFIER» в выводе утилиты **snmpget**, но могут также отображаться как «STRING», «Hex-STRING», «OID» и другие, в зависимости от наличия подсказки.

Шаг 2

Создайте узел сети, соответствующий устройству.




Добавьте SNMP-интерфейс для узла сети:

- Введите IP-адрес/DNS-имя и номер порта.
- Выберите *версию SNMP* в раскрывающемся списке.
- Добавьте учетные данные интерфейса в зависимости от выбранной версии SNMP:
  - SNMPv1, v2 требуют только community (обычно 'public').
  - SNMPv3 требует более специфичных параметров (см. ниже).
- Укажите максимальное значение повторений (по умолчанию: 10) для нативных SNMP bulk-запросов (GetBulkRequest-PDU); только для элементов `discovery[]` и `walk[]` в SNMPv2 и v3. Обратите внимание, что слишком высокое значение может привести к тайм-ауту проверки SNMP-агента.
- Отметьте флажок *Использовать объединенные запросы*, чтобы разрешить объединенную обработку SNMP-запросов (не относится к нативным SNMP bulk-запросам "walk" и "get").

| Параметр SNMPv3 | Описание |
|---|---|
| *Context name* | Введите имя контекста для идентификации элемента в SNMP-подсети.<br>Пользовательские макросы в этом поле будут разрешены. |
| *Security name* | Введите имя безопасности.<br>Пользовательские макросы в этом поле будут разрешены. |
| *Security level* | Выберите уровень безопасности:<br>**noAuthNoPriv** - не используются ни протокол аутентификации, ни протокол конфиденциальности<br>**AuthNoPriv** - используется протокол аутентификации, протокол конфиденциальности не используется<br>**AuthPriv** - используются и протокол аутентификации, и протокол конфиденциальности |
| *Authentication protocol* | Выберите протокол аутентификации - *MD5*, *SHA1*; с net-snmp 5.8 и новее - *SHA224*, *SHA256*, *SHA384* или *SHA512*. |
| *Authentication passphrase* | Введите парольную фразу аутентификации.<br>Пользовательские макросы в этом поле будут разрешены. |

| Параметр SNMPv3 | Описание |
|---|---|
| *Privacy protocol* | Выберите протокол конфиденциальности - *DES*, *AES128*, *AES192*, *AES256*, *AES192C* (Cisco) или *AES256C* (Cisco).<br>См. примечания о поддержке протокола конфиденциальности |
| *Privacy passphrase* | Введите парольную фразу конфиденциальности.<br>Пользовательские макросы в этом поле будут разрешены. |

В случае неверных учетных данных SNMPv3 (security name, authentication protocol/passphrase, privacy protocol):

- Zabbix получает ERROR от net-snmp, за исключением неверной *Privacy passphrase*, в этом случае Zabbix получает ошибку TIMEOUT от net-snmp.
- Доступность SNMP-интерфейса изменится на красный цвет (недоступен).

**Note:**
Изменения в *Authentication protocol*, *Authentication passphrase*, *Privacy protocol* или *Privacy passphrase*, внесенные без изменения *Security name*, обычно применяются автоматически при обновлении соответствующего SNMPv3-интерфейса в Zabbix. Если также изменяется *Security name*, все параметры будут обновлены немедленно.

Вы можете использовать один из предоставленных SNMP-шаблонов, который автоматически добавит набор элементов данных. Перед использованием шаблона убедитесь, что он совместим с узлом сети.

Нажмите *Add*, чтобы сохранить узел сети.

Поддержка протоколов конфиденциальности

В зависимости от вашей операционной системы и конфигурации net-snmp некоторые протоколы конфиденциальности могут быть недоступны:

- В некоторых более новых операционных системах (например, RHEL9) поддержка DES отключена для пакета net-snmp.
- Протоколы шифрования AES192 и более стойкие не поддерживаются из коробки в операционных системах старше RHEL 8, CentOS 8, Oracle Linux 8, Debian 12, Ubuntu LTS 22.04, openSUSE Leap 15.5.

Чтобы проверить, поддерживает ли библиотека net-snmp AES192+, используйте один из следующих вариантов:

1. `net-snmp-config`:

```
net-snmp-config --configure-options
```

Если вывод содержит `--enable-blumenthal-aes`, поддержка AES192+ есть.

Обратите внимание, что net-snmp-config входит в состав пакета разработки для SNMP (libsnmp-dev для Debian/Ubuntu, net-snmp-devel для CentOS/RHEL/OL/SUSE) и может не быть установлен по умолчанию.

2. `snmpget`:

```
snmpget -v 3 -x AES-256
```

Если вывод содержит `Invalid privacy protocol specified after -3x flag: AES-256`, поддержка AES192+ отсутствует. Если вывод содержит `No hostname specified.`, поддержка AES192+ отсутствует.

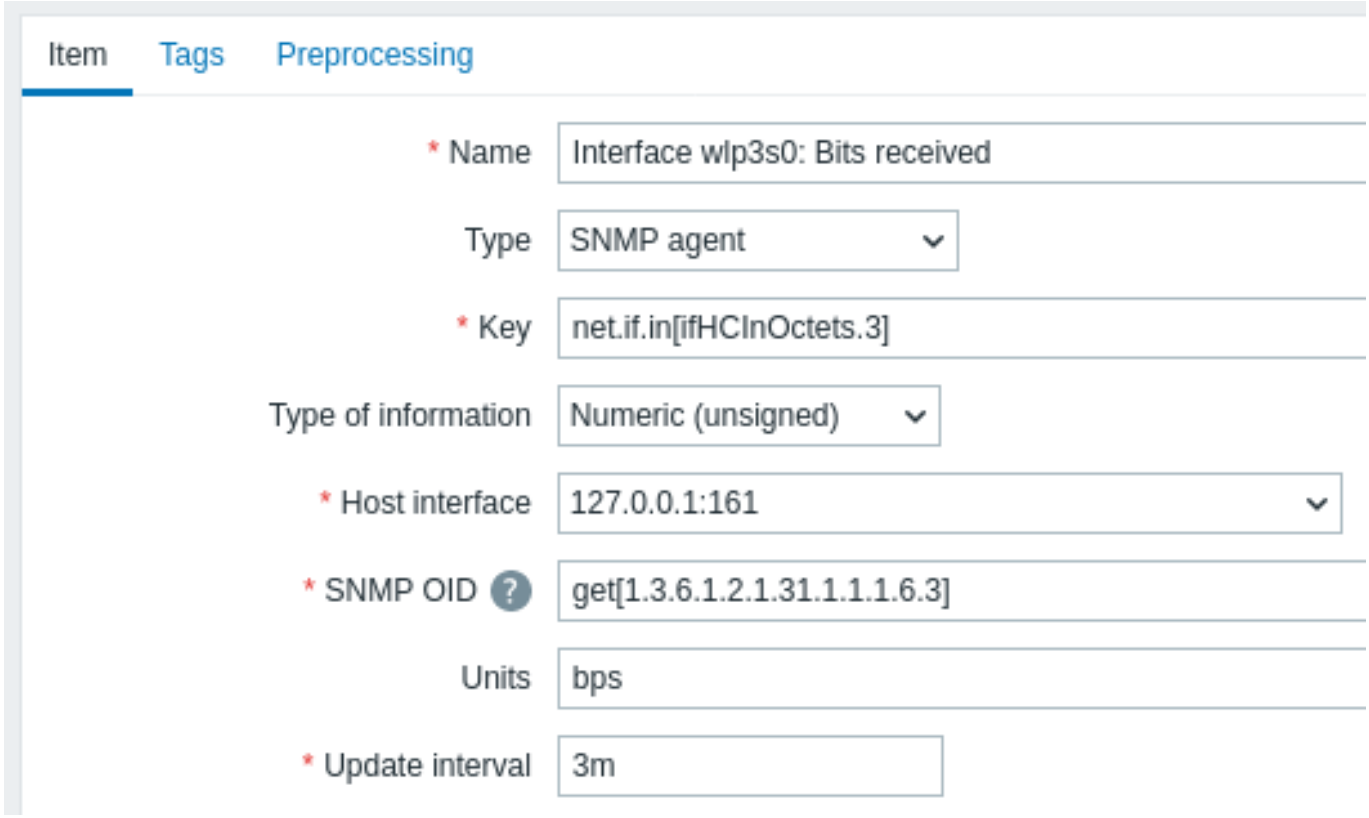
Если ваша библиотека net-snmp не поддерживает протоколы AES192 и выше, пересоберите net-snmp с опцией `--enable-blumenthal-aes`, затем пересоберите сервер Zabbix, указав опцию `--with-net-snmp=/home/user/yourcustomnetsnm`

Шаг 3

Создайте элемент данных для мониторинга.

Итак, теперь вернитесь в Zabbix и нажмите *Items* для SNMP-узла сети, который вы создали ранее. В зависимости от того, использовали ли вы шаблон при создании узла сети, у вас будет либо список SNMP-элементов данных, связанных с вашим узлом сети, либо просто пустой список. Мы будем исходить из того, что вы создадите элемент данных самостоятельно, используя информацию, которую только что получили с помощью snmpwalk и snmpget, поэтому нажмите *Create item*.

Заполните обязательные параметры в новой форме элемента данных:




| Parameter | Description |
|---|---|
| *Name* | Введите имя элемента данных. |
| *Type* | Выберите здесь **SNMP agent**. |
| *Key* | Введите ключ, чтобы он был понятным. |
| *Host interface* | Обязательно выберите SNMP-интерфейс, например вашего коммутатора/маршрутизатора. |

| Parameter | Description |
|---|---|
| *SNMP OID* | Используйте один из поддерживаемых форматов для ввода значения(й) OID:<br><br>**walk[OID1,OID2,...]** - получить поддерево значений.<br>Например: `walk[1.3.6.1.2.1.2.2.1.2,1.3.6.1.2.1.2.2.1.3]`.<br>Этот вариант использует нативные SNMP bulk-запросы (GetBulkRequest-PDUs) асинхронно.<br>Параметры тайм-аута для этого элемента данных можно задать в форме конфигурации элемента данных. Рекомендуется установить небольшое значение тайм-аута, чтобы избежать длительных задержек, если устройство недоступно, так как при истечении тайм-аута или ошибке предыдущих попыток будет выполнено до 5 повторов (например, тайм-аут 3 секунды может привести к ожиданию в 15 секунд).<br>Вы можете использовать его как мастер-элемент данных, а зависимые элементы данных будут извлекать данные из мастер-элемента с помощью предварительной обработки.<br>Можно указать несколько OID в одном snmp walk, например `walk[OID1,OID2,...]`, чтобы асинхронно обрабатывать по одному OID за раз.<br>Если bulk-запрос не возвращает результатов, выполняется попытка получить одну запись без bulk-запроса.<br>В качестве параметров поддерживаются имена MIB; таким образом, `walk[1.3.6.1.2.1.2.2.1.2]` и `walk[ifDescr]` вернут одинаковый результат.<br>Если указано несколько OID/MIB, то есть `walk[ifDescr,ifType,ifPhysAddress]`, вывод будет представлять собой объединенный список.<br>Запросы GetBulk используются с интерфейсами SNMPv2 и v3, а GetNext - с интерфейсами SNMPv1; максимальное число повторений для bulk-запросов настраивается на уровне интерфейса.<br>Параметр max repetitions влияет на bulk-запросы, определяя максимальное количество OID, возвращаемых в одном bulk-ответе.<br>Большее значение приводит к более крупным bulk-ответам, уменьшая число необходимых передач. Однако не все устройства могут поддерживать очень высокие значения, что может вызвать проблемы.<br>Этот элемент данных возвращает вывод утилиты snmpwalk с параметрами -Oe -Ot -On.<br>Вы можете использовать этот элемент данных как мастер-элемент в SNMP discovery.<br><br>**get[OID]** - асинхронно получить *одно* значение.<br>Например: `get[1.3.6.1.2.1.31.1.1.1.6.3]`<br>Параметры тайм-аута для этого элемента данных можно задать в форме конфигурации элемента данных. Рекомендуется установить небольшое значение тайм-аута, чтобы избежать длительных задержек, если устройство недоступно, так как при истечении тайм-аута или ошибке предыдущих попыток будет выполнено до 5 повторов (например, тайм-аут 3 секунды может привести к ожиданию в 15 секунд).<br><br>**OID** - (устаревший) введите один текстовый или числовой OID для получения одного значения синхронно, при необходимости в сочетании с другими значениями.<br>Например: `1.3.6.1.2.1.31.1.1.1.6.3`.<br>Для этого варианта время ожидания проверки элемента данных будет равно значению, заданному в [файле конфигурации] сервера(/manual/appendix/config/zabbix_server#timeout).<br><br>**Рекомендуется** использовать элементы данных `walk[OID]` и `get[OID]` для лучшей производительности. Все элементы данных `walk[OID]` и `get[OID]` выполняются асинхронно - не требуется получать ответ на один запрос, прежде чем будут запущены другие проверки. Разрешение DNS также выполняется асинхронно.<br>Максимальная параллельность асинхронных проверок составляет 1000 (определяется параметром MaxConcurrentChecksPerPoller). Количество асинхронных SNMP-опросчиков определяется параметром StartSNMPPollers.<br><br>Обратите внимание, что для статистики сетевого трафика, возвращаемой любым из методов, на вкладке *Preprocessing* необходимо добавить шаг *Change per second*; в противном случае вы получите накопительное значение от SNMP-устройства вместо последнего изменения. |

Все обязательные поля ввода отмечены красной звездочкой.

Теперь сохраните элемент данных и перейдите в *Monitoring* > *Latest data* для ваших SNMP-данных.

Пример 1

Общий пример:

| Параметр | Описание |
|---|---|
| **OID** | 1.2.3.45.6.7.8.0 (или .1.2.3.45.6.7.8.0) |
| **Ключ** | <Уникальная строка, которая используется как ссылка в триггерах><br>Например, «my_param». |

Обратите внимание, что OID можно задать в числовом или строковом представлении. Тем не менее, в некоторых случаях строковый OID должен быть сконвертирован в числовое представление. Для этого можно использовать утилиту snmpget:

```
snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

Пример 2

Мониторинг времени работы:

| Параметр | Описание |
|---|---|
| **OID** | MIB::sysUpTime.0 |
| **Ключ** | router.uptime |
| **Тип информации** | Числовой (с плавающей точкой) |
| **Единица измерения** | uptime |
| **Множитель** | 0.01 |

Собственные SNMP bulk-запросы

Элемент данных **walk[OID1,OID2,...]** позволяет использовать собственную SNMP-функциональность для bulk-запросов (GetBulkRequest-PDU), доступную в версиях SNMP 2/3.

Запрос GetBulk в SNMP выполняет несколько запросов GetNext и возвращает результат в одном ответе. Это можно использовать как для обычных SNMP-элементов данных, так и для SNMP-обнаружения, чтобы минимизировать число сетевых обменов.

Элемент данных SNMP **walk[OID1,OID2,...]** можно использовать как основной элемент данных, который собирает данные одним запросом, а зависимые элементы данных разбирают ответ по мере необходимости с помощью предварительной обработки.

Обратите внимание, что использование собственных SNMP bulk-запросов не связано с опцией объединения SNMP-запросов, которая является собственным способом Zabbix объединять несколько SNMP-запросов (см. следующий раздел).

Для SNMP bulk-элементов данных будет выполнено до пяти повторных попыток, чтобы избежать сбоя, если один из пакетов будет потерян. Таймаут для SNMP-элементов данных с `get` и `walk` (задается в форме настройки элемента данных) устанавливается на всю сессию. Таймаут применяется независимо от того, были ли данные получены полностью; если данные получены частично (например, успешно собраны данные только для одного из нескольких OID), то элемент данных становится неподдерживаемым с сообщением "Only partial data received". Если таймаут истекает, будет выполнена повторная попытка, таймаут будет сброшен, и последний запрос будет отправлен снова, что позволит продолжить сессию с последнего запроса, если один пакет был потерян или пришел слишком поздно. Рекомендуется установить небольшое значение таймаута, чтобы избежать длительных задержек, если устройство недоступно, поскольку при истечении таймаута или сбое предыдущих попыток будет выполнено до 5 повторных попыток (например, таймаут 3 секунды может привести к ожиданию до 15 секунд).

Внутреннее устройство комбинированной обработки

Zabbix-сервер и прокси-сервер могут запрашивать у SNMP-устройств несколько значений в одном запросе. Это влияет на несколько типов элементов данных SNMP:

- обычные элементы данных SNMP
- элементы данных SNMP с динамическими индексами
- SNMP правила низкоуровневого обнаружения

Все элементы данных SNMP на одном интерфейсе с идентичными параметрами планируются к опросу одновременно. Первые два типа элементов данных обрабатываются поллерами группами, не превышающими 128 элементов, тогда как правила низкоуровневого обнаружения обрабатываются по отдельности, как и раньше.

На нижнем уровне для запроса значений выполняются два вида операций: получение нескольких указанных объектов («получение данных», getting) и обход дерева OID («обход данных», walking).

Для «получения данных» используется GetRequest-PDU с не более чем 128 привязками переменных. Для «обхода данных» используется GetNextRequest-PDU для SNMPv1, а для SNMPv2 и SNMPv3 — GetBulkRequest с полем «max-repetitions» не более 128.

Таким образом, преимущества комбинированной обработки для каждого типа элементов SNMP описаны ниже:

- обычные элементы данных SNMP получают преимущества от улучшений в области «получения данных»;
- элементы данных SNMP с динамическими индексами получают преимущества как в области «получения данных», так и в области «обхода данных»: «получение данных» используется для проверки индекса, а «обход данных» — для создания кэша;
- низкоуровневые правила обнаружения SNMP получают преимущества от улучшений в области «обхода данных».

Однако, существует техническая проблема: не все устройства способны возвращать 128 значений за запрос. Некоторые всегда возвращают корректный ответ, но другие либо возвращают ошибку «tooBig(1)», либо вообще не отвечают, если потенциальный ответ превышает определённый предел.

Чтобы найти оптимальное количество объектов для запроса для данного устройства, Zabbix использует следующую стратегию. Он начинает с осторожного запроса одного значения. Если запрос успешен, Zabbix запрашивает два значения. Если запрос снова успешен, Zabbix запрашивает три значения и продолжает аналогичным образом, умножая количество запрошенных объектов на 1,5. В результате получается следующая последовательность размеров запросов: 1, 2, 3, 4, 6, 9, 13, 19, 28, 42, 63, 94, 128.

Однако, как только устройство отказывается давать корректный ответ (например, для 42 переменных), Zabbix делает два действия.

Во-первых, для текущего пакета элементов данных он вдвое уменьшает количество объектов в одном запросе и запрашивает 21 переменную. Если устройство работает, то запрос должен сработать в подавляющем большинстве случаев, поскольку 28 переменных, как известно, сработали, а 21 — значительно меньше. Однако, если и это не поможет, Zabbix возвращается к поочерёдному запросу значений. Если и на этом этапе не получится, то устройство точно не отвечает, и размер запроса не имеет значения.

Второе действие Zabbix для последующих пакетов элементов данных заключается в том, что он начинает с последнего успешного количества переменных (28 в нашем примере) и продолжает увеличивать размер запроса на 1, пока не будет достигнут предел. Например, если максимальный размер ответа составляет 32 переменные, последующие запросы будут иметь размеры 29, 30, 31, 32 и 33. Последний запрос завершится ошибкой, и Zabbix больше никогда не выполнит запрос размером 33. С этого момента Zabbix будет запрашивать не более 32 переменных для этого устройства.

Если большие запросы с таким количеством переменных завершатся неудачей, это может означать одно из двух. Точные критерии, используемые устройством для ограничения размера ответа, неизвестны, но мы пытаемся приблизительно определить их, используя количество переменных. Итак, первая возможность заключается в том, что это количество переменных приблизительно соответствует фактическому ограничению размера ответа устройства в общем случае: иногда ответ меньше ограничения, иногда больше. Вторая возможность заключается в том, что UDP-пакет в любом направлении просто потерялся. По этим причинам, если Zabbix получает неудавшийся запрос, он уменьшает максимальное количество переменных, чтобы попытаться глубже войти в комфортный диапазон устройства, но не более двух раз.

В приведённом выше примере, если запрос с 32 переменными завершится ошибкой, Zabbix уменьшит количество запросов до 31. Если и это не удастся, Zabbix уменьшит количество запросов до 30. Однако Zabbix не уменьшит количество запросов ниже 30, поскольку будет считать, что дальнейшие сбои связаны с потерей UDP-пакетов, а не с ограничением устройства.

Если же устройство не может корректно обрабатывать объединённые запросы по другим причинам и описанная выше эвристика не работает, для каждого интерфейса предусмотрена настройка «Использование объединённых запросов», которая позволяет отключить объединённые запросы для этого устройства.

Если объединённые запросы вызывают частичные или некорректные ответы, приводящие к некорректным посекундным (дельта) расчётам (например, к заметным скачкам показаний счётчиков интерфейсов), отключите функцию *Использование объединённых запросов (Use combined requests)* для затронутого интерфейса, чтобы принудительно выполнять отдельные запросы по элементам данных; это часто предотвращает ложные скачки. В качестве альтернативы рассмотрите возможность использования асинхронных элементов данных `get[]` или `walk[]`, которые выполняются асинхронно и не подлежат комбинированию в соответствии с настройками интерфейса *Использование объединённых запросов (Use combined requests)* — их можно использовать вместо устаревших синхронных проверок OID, чтобы избежать проблем, связанных с объединёнными запросами. Найдите в журнале сервера/прокси записи, похожие на ту, что показана в разделе Обзор, чтобы помочь идентифицировать затронутые устройства.

Кроме того, если интерфейс часто становится недоступным, может потребоваться увеличить параметр `UnavailableDelay`

в файлах конфигурации Zabbix-сервера или Zabbix-прокси, чтобы уменьшить частоту запросов. Элементы могут перестать поддерживаться, если во время обнаружения или проверки OID получены частичные данные.

1. Динамические индексы

Обзор

Хотя вы можете найти требуемый номер индекса (например, сетевого интерфейса) среди SNMP OID'ов, иногда вы не можете полностью положиться на то, что номер индекса всегда останется таким же.

Индексы могут быть динамическими — они могут изменяться время от времени и, как следствие, ваш элемент данных может перестать работать.

Чтобы избежать этого, имеется возможность указать OID, который принимает во внимание возможность изменения номера индекса.

Например, если вам необходимо получить значение индекса, который относится к **ifInOctets**, которое соответствует интерфейсу **GigabitEthernet0/1** на устройстве Cisco, используйте следующий OID:

```
ifInOctets["index","ifDescr","GigabitEthernet0/1"]
```

Синтаксис

Используется специальный синтаксис для OID:

**<OID данных>["index","<базовый OID индекса>","<строка для поиска>"]**

| Parameter | Description |
|---|---|
| OID of data | Основной OID, используемый для получения данных элемента данных. |
| index | Метод обработки. В настоящее время поддерживается один метод:<br>**index** - поиск индекса и добавление его к OID данных |
| base OID of index | Этот OID будет запрашиваться для получения значения индекса, соответствующего строке. |
| string to search for | Строка, используемая для точного сопоставления со значением при выполнении поиска. Учитывается регистр. |

Пример

Получение использования памяти процессом *apache*.

Если используется этот синтаксис OID:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem["index","HOST-RESOURCES-MIB::hrSWRunPath", "/usr/sbin/apache2"]
```

номер индекса будет искаться здесь:

```
...
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"
...
```

Теперь у нас есть индекс, равный 5388. Индекс будет присоединён к OID'у данных, чтобы получить интересующее нас значение:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem.5388 = INTEGER: 31468 KBytes
```

Кэширование поиска индексов

При запросе динамического индекса элемента данных Zabbix извлекает и кэширует всю таблицу SNMP под базовым OID, даже если совпадение будет найдено ранее. Это делается на случай, если другой элемент данных в дальнейшем будет ссылаться на тот же самый базовый OID — Zabbix будет искать индекс в кэше вместо ещё одного запроса к наблюдаемому устройству. Обратите внимание, что каждый процесс поллера использует свой собственный кэш.

Во всех последующих операциях получения значений проверяется только найденный индекс. Если он не изменился, значение запрашивается. Если он изменился, кэш перестраивается — каждый поллер, который встречает изменённый индекс, снова обходит и кэширует таблицу индексов SNMP.

2. Специальные OID’ы

Некоторые OID’ы, наиболее часто используемые в SNMP, автоматически преобразуются в числовые представления самим Zabbix. Например, **ifIndex** преобразуется в **1.3.6.1.2.1.2.2.1.1**, **ifIndex.0** преобразуется в **1.3.6.1.2.1.2.2.1.1.0**.

В таблице содержится перечень специальных OID’ов.

| Специальный OID | Идентификатор | Описание |
|---|---|---|
| ifIndex | 1.3.6.1.2.1.2.2.1.1 | Уникальное для каждого интерфейса значение. |
| ifDescr | 1.3.6.1.2.1.2.2.1.2 | Текстовая строка, которая содержит информацию об интерфейсе. Эта строка может включать в себя название компании-производителя, имя продукта и версию аппаратного интерфейса. |
| ifType | 1.3.6.1.2.1.2.2.1.3 | Тип интерфейса, выделяется в соответствии с физическим/канальным протоколом(ами) непосредственно «под» сетевым уровнем стека сетевых протоколов. |
| ifMtu | 1.3.6.1.2.1.2.2.1.4 | Размер наибольшей дейтаграммы, которую может отправить/получить интерфейс, указывается в октетах. |
| ifSpeed | 1.3.6.1.2.1.2.2.1.5 | Оценочная скорость текущей пропускной способности интерфейса в битах за секунду. |
| ifPhysAddress | 1.3.6.1.2.1.2.2.1.6 | Физический адрес интерфейса на канальном уровне непосредственно «под» сетевым уровнем в стеке сетевых протоколов. |
| ifAdminStatus | 1.3.6.1.2.1.2.2.1.7 | Текущее административное состояние интерфейса. |
| ifOperStatus | 1.3.6.1.2.1.2.2.1.8 | Текущее рабочее состояние интерфейса. |
| ifInOctets | 1.3.6.1.2.1.2.2.1.10 | Общее число октетов, полученных на интерфейсе, включая символы кадрирования. |
| ifInUcastPkts | 1.3.6.1.2.1.2.2.1.11 | Количество пакетов одноадресной рассылки, доставленных протоколу более высокого уровня. |
| ifInNUcastPkts | 1.3.6.1.2.1.2.2.1.12 | Количество пакетов НЕ одноадресной рассылки (т.е. широковещательных или многоадресных), доставленных протоколу более высокого уровня. |
| ifInDiscards | 1.3.6.1.2.1.2.2.1.13 | Количество входящих, но отвергнутых пакетов, даже если не было обнаружено ошибок, мешающих доставке пакетов протоколу более высокого уровня. Одной из возможных причин для отвержения пакета могло быть освобождение места в буфере. |
| ifInErrors | 1.3.6.1.2.1.2.2.1.14 | Количество входящих пакетов, полученных с ошибкой, из-за которой пакеты не были доставлены протоколу более высокого уровня. |
| ifInUnknownProtos | 1.3.6.1.2.1.2.2.1.15 | Количество пакетов, полученных через интерфейс, но отвергнутых из-за неизвестного или неподдерживаемого протокола. |
| ifOutOctets | 1.3.6.1.2.1.2.2.1.16 | Общее количество отправленных октетов с интерфейса, включая символы кадрирования. |
| ifOutUcastPkts | 1.3.6.1.2.1.2.2.1.17 | Общее количество пакетов, которые пытался отправить верхний уровень стека сетевых протоколов, и которые адресованы НЕ на многоадресный или широковещательный адрес на этом подуровне, включая те, которые были отвергнуты или не отправлены. |
| ifOutNUcastPkts | 1.3.6.1.2.1.2.2.1.18 | Общее количество пакетов, которые пытался отправить верхний уровень стека сетевых протоколов, и которые адресованы на многоадресный или широковещательный адрес на этом подуровне, включая те, которые были отвергнуты или не отправлены. |
| ifOutDiscards | 1.3.6.1.2.1.2.2.1.19 | Количество исходящих пакетов, которые были отвергнуты, даже если не было обнаружено ошибок, мешающих отправке. Одной из возможных причин отвержения пакета могло быть освобождение места в буфере. |
| ifOutErrors | 1.3.6.1.2.1.2.2.1.20 | Количество исходящих пакетов, которые не смогли быть отправлены из-за ошибок. |

| Специальный OID | Идентификатор | Описание |
|---|---|---|
| ifOutQLen | 1.3.6.1.2.1.2.2.1.21 | Длина очереди исходящих пакетов (в пакетах). |

3. MIB файлы

Введение

MIB обозначает Базу Управляющей Информации (Management Information Base). MIB файлы позволяют вам использовать текстовое представление OID'а (Идентификатора Объекта). При мониторинге Zabbix'ом устройств SNMP можно использовать сырые OID'ы, но если для Вас более комфортно использовать текстовое представление, то потребуется установить MIB файлы.

Например,

`ifHCOutOctets`

является текстовым представлением OID'а

`1.3.6.1.2.1.31.1.1.1.10`

Установка MIB файлов

На системах на основе Debian:

```
apt install snmp-mibs-downloader
download-mibs
```

На системах на основе RedHat:

```
dnf install net-snmp-libs
```

Подлючение MIB файлов

На системах на основе RedHat MIB файлы должны быть подключены по умолчанию. На системах на основе Debian вам нужно отредактировать файл `/etc/snmp/snmp.conf` и закомментировать строку, которая содержит `mibs`:

```
# As the snmp packages come without MIB files due to license reasons, loading
# of MIBs is disabled by default. If you added the MIBs you can re-enable
# loading them by commenting out the following line.
mibs :
```

Тестирование MIB файлов

Тестирование SNMP MIB'ов можно выполнить с использованием утилиты `snmpwalk`. Если у вас эта утилита не установлена, воспользуйтесь следующими инструкциями.

На системах на основе Debian:

```
apt install snmp
```

На системах на основе RedHat:

```
dnf install net-snmp-utils
```

После чего следующая команда должна не выдавать ошибку при выполнении запроса к сетевому устройству:

```
$ snmpwalk -v 2c -c public <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
[...]
```

Использование MIB'ов в Zabbix

Самое главное, что следует иметь в виду, — это то, что процессы Zabbix не знают об изменениях, внесённых в MIB файлы. Поэтому после каждого изменения вам необходимо перезапустить Zabbix сервер или прокси, например:

```
systemctl restart zabbix-server
```

После выполнения этой команды изменения, сделанные в MIB файлах, вступят в силу.

Использование пользовательских MIB файлов

Имеются стандартные MIB файлы, которые поставляются с каждым GNU/Linux дистрибутивом. Но некоторые производители устройств поставляют свои собственные файлы.

Допустим, вы хотите использовать CISCO-SMI MIB файл. Следующая инструкция поможет загрузить и установить этот файл:

```
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
```

Теперь вы должны иметь возможность использовать его. Попытайтесь перевести имя объекта *ciscoProducts* из MIB файла в OID:

```
snmptranslate -IR -On CISCO-SMI::ciscoProducts
.1.3.6.1.4.1.9.1
```

Если вы видите ошибки вместо OID, удостоверьтесь, что все предыдущие команды не вернули никаких ошибок.

Когда перевод имени объекта заработает, вы будете готовы использовать пользовательский MIB файл. Обратите внимание, что в запросе используется MIB префикс имени (*CISCO-SMI::*). Вам потребуется его указывать, когда используете инструменты командной строки, а также Zabbix.

Не забудьте перезапустить Zabbix сервер/прокси перед использованием этого MIB файла в Zabbix.

**Attention:**
Имейте в виду, что у MIB файлов могут быть зависимости. То есть, одному MIB может требоваться другой. Чтобы удовлетворить эти зависимости, вам нужно установить все необходимые MIB файлы.

#### 4 SNMP trap

Обзор

Получение SNMP traps — это обратный процесс по сравнению с опросом устройств с поддержкой SNMP.

В этом случае информация отправляется с устройства с поддержкой SNMP в snmptrapd и собирается, или "ловится", сервером Zabbix или прокси Zabbix из файла.

Обычно traps отправляются при изменении некоторого условия, и агент подключается к серверу на порт 162 (в отличие от порта 161 на стороне агента, который используется для запросов). Использование traps может помочь обнаружить некоторые кратковременные проблемы, возникающие в промежутке между опросами, которые могут быть пропущены при получении данных по запросу.

Получение SNMP traps в Zabbix предназначено для работы с **snmptrapd** и одним из механизмов передачи traps в Zabbix — либо Bash-скриптом, либо Perl-скриптом, либо SNMPTT.

**Note:**
Самый простой способ настроить мониторинг traps после конфигурирования Zabbix — использовать решение на Bash-скрипте, поскольку Perl и SNMPTT часто отсутствуют в современных дистрибутивах и требуют более сложной настройки. Однако это решение использует скрипт, настроенный как `traphandle`. Для лучшей производительности в рабочих системах используйте встроенное решение на Perl (либо скрипт с опцией `do perl`, либо SNMPTT).

Последовательность обработки trap:

1. `snmptrapd` получает trap.
2. `snmptrapd` передает trap скрипту-получателю (Bash, Perl) или SNMPTT.
3. Получатель разбирает, форматирует и записывает trap в файл.
4. SNMP trapper Zabbix читает и разбирает файл traps.
5. Для каждого trap Zabbix находит все элементы данных SNMP trapper с интерфейсами узлов сети, соответствующими полученному адресу trap. Обратите внимание, что при сопоставлении используется только выбранный *IP* или *DNS* в интерфейсе узла сети.

6. Для каждого найденного элемента данных trap сравнивается с regexp в `snmptrap[regexp]`. Trap устанавливается как значение **всех** совпавших элементов данных. Если совпадающий элемент данных не найден и существует элемент данных `snmptrap.fallback`, trap устанавливается как его значение.
7. Если trap не был установлен как значение ни одного элемента данных, Zabbix по умолчанию записывает в журнал несопоставленный trap. (Это настраивается параметром *Log unmatched SNMP traps* в *Administration* > *General* > *Other*.)

Замечания о переключении HA

Во время переключения ноды кластера высокой доступности (HA) Zabbix продолжит обработку после последней записи в последней отметки времени ISO 8601; если та же запись не найдена, то для определения последней позиции будет использоваться только отметка времени.

Настройка SNMP traps

Для этого типа элемента данных требуется следующая настройка веб-интерфейса:

Шаг 1: Создайте SNMP-интерфейс для узла сети

1. В *Сбор данных* > *Узлы сети*, создайте узел сети или отредактируйте существующий.
2. В поле *Интерфейсы* выберите тип интерфейса *SNMP*.
3. Укажите IP-адрес/DNS-имя и номер порта.
4. Выберите *версию SNMP* в выпадающем списке: (*SNMPv1*, *SNMPv2* или *SNMPv3*) и добавьте учетные данные интерфейса в зависимости от выбранной версии SNMP.

Адрес каждого полученного trap будет сравниваться с IP- и DNS-адресами всех SNMP-интерфейсов, чтобы найти соответствующие узлы сети.

Шаг 2: Настройте элемент данных

1. Для узла сети создайте элемент данных или отредактируйте существующий.
2. В поле *Ключ* используйте один из ключей SNMP trap:
   - `snmptrap[regexp]` - перехватывает все SNMP trap, которые соответствуют регулярному выражению, указанному в параметре `regexp`; если `regexp` не задан, перехватывает любой trap.<br> В параметре поддерживаются пользовательские макросы и глобальные регулярные выражения.<br> Возвращаемое значение: SNMP trap.<br> Этот элемент данных можно задать только для SNMP-интерфейсов.
   - `snmptrap.fallback` - перехватывает все SNMP trap, которые не были перехвачены ни одним из элементов данных `snmptrap[]` для этого интерфейса.<br> Возвращаемое значение: SNMP trap.<br> Этот элемент данных можно задать только для SNMP-интерфейсов.

**Note:**
В настоящее время сопоставление с регулярным выражением в нескольких строках не поддерживается.

3. Установите для *Типа информации* значение *Log*, чтобы временные метки были разобраны. Другие форматы, такие как *Numeric*, также допустимы, но могут потребовать пользовательский обработчик trap.

Настройка мониторинга SNMP трапов

Настройка сервера/прокси Zabbix

Чтобы читать ловушки, сервер или прокси Zabbix должны быть настроены на запуск процесса SNMP trapper и указание на файл ловушек, который записывается SNMPTT или приемником ловушек Bash/Perl. Для этого отредактируйте файл конфигурации (`zabbix\_server.conf` или `zabbix\_proxy.conf`):

```
StartSNMPTrapper=1
SNMPTrapperFile=[TRAP FILE]
```

**Warning:**
Если используется параметр systemd `PrivateTmp`, этот файл, скорее всего, не будет работать в `/tmp`.

Настройка приемника ловушек Bash

Требования: только snmptrapd.

Скрипт приемника ловушек Bash script можно использовать для передачи ловушек на сервер Zabbix из snmptrapd через файл trapper. Чтобы настроить его, добавьте параметр `traphandle` в файл конфигурации snmptrapd (`snmptrapd.conf`), см. пример.

**Note:**
Может потребоваться перезапустить snmptrapd, чтобы он применил изменения в своей конфигурации.

Настройка Perl-приемника ловушек

Требования: Perl, Net-SNMP, собранный с `--enable-embedded-perl` (по умолчанию это выполняется начиная с Net-SNMP 5.4)

Perl-приемник ловушек (см. `misc/snmptrap/zabbix\_trap\_receiver.pl`) можно использовать для передачи ловушек на сервер Zabbix напрямую из snmptrapd. Чтобы настроить его:

- Добавьте Perl-скрипт в файл конфигурации snmptrapd (`snmptrapd.conf`), например:

```
perl do "[FULL PATH TO PERL RECEIVER SCRIPT]";
```

- настройте приемник, например:

```
$SNMPTrapperFile = '[TRAP FILE]';
$DateTimeFormat = '[DATE TIME FORMAT]';
```

**Note:**
Возможно, потребуется перезапустить snmptrapd, чтобы он применил изменения в своей конфигурации.

**Note:**
Если имя скрипта не заключено в кавычки, snmptrapd откажется запускаться и выведет сообщения, подобные следующим:<br><br>

```
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
```

Настройка SNMPTT

Для начала, snmptrapd необходимо настроить на использование SNMPTT.

**Note:**
Для лучшей производительности SNMPTT нужно сконфигурировать как демон, используя **snmptthandler-embedded** для передачи ему трапов. Смотрите инструкции по настройке SNMPTT [en].

Когда SNMPTT настроен на получение трапов, настройте `snmptt.ini`:

1. включите использование модуля Perl из пакета NET-SNMP:

```
net_snmp_perl_enable = 1
```

2. включите журналирование трапов в файл с трапами, который будет читаться Zabbix'ом:

```
log_enable = 1
log_file = [TRAP    ]
```

3. задайте формат даты/времени:

```
date_time_format = %Y-%m-%dT%H:%M:%S%z
```

**Warning:**
Пакет «net-snmp-perl» удалён в RHEL 8.0-8.2; добавлен заново в RHEL 8.3. Для получения более подробной информации смотрите известные проблемы.

Теперь задайте формат трапов, чтобы они распознавались Zabbix'ом (отредактируйте snmptt.conf):

1. Каждая инструкция FORMAT должна начинаться с "ZBXTRAP [адрес]", где [адрес] будет сравниваться с IP и DNS адресами у SNMP интерфейсов в Zabbix. Например:

```
EVENT coldStart .1.3.6.1.6.3.1.1.5.1 "Status Events" Normal
FORMAT ZBXTRAP $aA Device reinitialized (coldStart)
```

2. Подробнее о формате SNMP трапов смотрите ниже.

**Attention:**
Не используйте неизвестные трапы — Zabbix может их не распознать. Неизвестные трапы могут быть обработаны, если задать общее событие в snmptt.conf:<br><br>

```
EVENT general .* "General event" Normal
```

Формат SNMP trap

Все настроенные получатели trap Perl и конфигурация trap SNMPTT должны форматировать trap следующим образом:

```
[timestamp] [the trap, part 1] ZBXTRAP [address] [the trap, part 2]
```

где

- `[timestamp]` - метка времени в формате `%Y-%m-%dT%H:%M:%S%z`.
- `ZBXTRAP` - заголовок, который указывает, что в этой строке начинается новый trap.
- `[address]` - IP-адрес, используемый для поиска узла сети для этого trap.

Обратите внимание, что `ZBXTRAP` и `[address]` будут удалены из сообщения во время обработки. Если trap отформатирован иначе, Zabbix может неожиданно разобрать trap.

Пример 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
```

Это приведет к следующему trap для интерфейса SNMP с 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
```

Системные требования

**Note:**
Рекомендуется установить файлы MIB, чтобы гарантировать, что значения элементов данных отображаются в правильном формате. Без файлов MIB могут возникнуть проблемы с форматированием, такие как отображение значений в HEX вместо UTF-8 или наоборот.

Поддержка больших файлов

Zabbix поддерживает большие файлы при работе с файлами SNMP трапов. Максимальный размер файла, который Zabbix может прочитать, — это 2^63 (8 эксабайт). Обратите внимание, что файловая система может иметь меньшее ограничение на максимальный размер файлов.

Ротация журнала

Zabbix не предоставляет систему ротации журнала - это должно выполняться пользователем. При ротации журнала сначала следует переименовать старый файл и только потом удалять его, чтобы не потерять ни одного trap:

1. Zabbix открывает trap-файл по последнему известному расположению и переходит к шагу 3.
2. Zabbix проверяет, был ли текущий открытый файл ротирован, сравнивая номер inode с номером inode заданного trap-файла. Если открытого файла нет, Zabbix сбрасывает последнее расположение и переходит к шагу 1.
3. Zabbix считывает данные из текущего открытого файла и задает новое расположение.
4. Новые данные анализируются. Если это был ротированный файл, файл закрывается и выполняется возврат к шагу 2.
5. Если новых данных нет, Zabbix засыпает на 1 секунду и возвращается к шагу 2.

Файловая система

Из-за реализации trap-файла Zabbix требуется, чтобы файловая система поддерживала inode для различения файлов (информация получается с помощью вызова `stat()`).

Примеры настройки с использованием разных версий протокола SNMP

В этом примере используются `snmptrapd` и Bash-скрипт-приемник для передачи trap-сообщений на сервер Zabbix.

Настройка:

1. Настройте Zabbix на запуск SNMP trapper и укажите файл trap-сообщений. Добавьте в `zabbix_server.conf`:

```
StartSNMPTrapper=1
SNMPTrapperFile=/var/lib/zabbix/snmptraps/snmptraps.log
```

2. Скачайте Bash-скрипт в /usr/sbin/zabbix_trap_handler.sh:

```
curl -o /usr/sbin/zabbix_trap_handler.sh https://raw.githubusercontent.com/zabbix/zabbix-docker/7.4/templa
```

Если необходимо, измените переменную ZABBIX_TRAPS_FILE в скрипте. Чтобы использовать значение по умолчанию, сначала создайте родительский каталог:

```
mkdir -p /var/lib/zabbix/snmptraps
```

3. Добавьте следующее в snmtrapd.conf (см. рабочий пример)

```
traphandle default /bin/bash /usr/sbin/zabbix_trap_handler.sh
```

**Note:**
Может потребоваться перезапустить snmptrapd, чтобы он применил изменения в своей конфигурации.

4. Создайте элемент данных SNMP TEST (учтите первоначальные требования к настройке):

   Type: *SNMP trap*<br> Type of information: *Log*<br> Host interface: *SNMP 127.0.0.1*<br> Key: snmptrap["linkUp"]<br> Log time format: yyyyMMdd.hhmmss

Обратите внимание, что используется формат даты и времени ISO 8601.

5. Далее мы настроим snmptrapd для выбранной версии протокола SNMP и отправим тестовые trap-сообщения с помощью утилиты snmptrap.

SNMPv1, SNMPv2

Протоколы SNMPv1 и SNMPv2 используют аутентификацию по "community string". В примере ниже мы будем использовать "secret" в качестве community string. Оно должно быть задано тем же значением на отправителях SNMP trap-сообщений.

Обратите внимание: хотя SNMPv2 по-прежнему широко используется в рабочих средах, он не обеспечивает ни шифрования, ни настоящей аутентификации отправителя. Данные передаются в открытом виде, поэтому эти версии протокола следует использовать только в защищенных средах, например в частной сети, и никогда не использовать в общедоступных или сторонних сетях.

Версия SNMP 1 сейчас практически не используется, поскольку она не поддерживает 64-битные счетчики и считается устаревшим протоколом.

Чтобы включить прием trap-сообщений SNMPv1 или SNMPv2, добавьте следующую строку в snmptrapd.conf. Замените secret на community string SNMP, настроенную на отправителях SNMP trap-сообщений:

```
authCommunity log,execute,net secret
```

Далее можно отправить тестовое trap-сообщение с помощью snmptrap. В этом примере мы используем распространенный OID "link up":

```
snmptrap -v 2c -c secret localhost '' linkUp.0
```

SNMPv3

SNMPv3 устраняет проблемы безопасности SNMPv1/v2 и обеспечивает аутентификацию и шифрование. Можно использовать методы аутентификации MD5 или несколько SHA, а также DES/несколько AES в качестве шифра.

Чтобы включить прием SNMPv3, добавьте следующие строки в snmptrapd.conf:

```
createUser -e 0x8000000001020304 traptest SHA mypassword AES
authuser log,execute traptest
```

**Attention:**
Обратите внимание на ключевое слово "execute", которое позволяет выполнять скрипты для этой модели безопасности пользователя.

```
snmptrap -v 3 -n "" -a SHA -A mypassword -x AES -X mypassword -l authPriv -u traptest -e 0x800000000102030
```

**Warning:**
Если вы хотите использовать надежные методы шифрования, такие как AES192 или AES256, используйте net-snmp начиная с версии 5.8. Возможно, потребуется пересобрать его с опцией configure --enable-blumenthal-aes. Более старые версии net-snmp не поддерживают AES192/AES256. См. также: Strong Authentication or Encryption.

Проверка

В обоих примерах в /var/lib/zabbix/snmptraps/snmptraps.log будут похожие строки:

```
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
```

Значение элемента данных в Zabbix будет таким:

```
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
```

Пример с Perl:

```
2024-01-30T11:42:54+0200 ZBXTRAP 127.0.0.1
PDU INFO:
  receivedfrom                    UDP: [127.0.0.1]:58649->[127.0.0.1]:162
  notificationtype                TRAP
  version                         1
  community                       public
  errorstatus                     0
  transactionid                   1
  requestid                       2101882550
  messageid                       0
  errorindex                      0
VARBINDS:
  DISMAN-EVENT-MIB::sysUpTimeInstance type=67 value=Timeticks: (457671) 1:16:16.71
  SNMPv2-MIB::snmpTrapOID.0       type=6  value=OID: IF-MIB::linkUp.0
```

Смотрите также

- Статья в Zabbix блоге о SNMP трапах [en]
- Настройка snmptrapd (официальная документация net-snmp) [en]
- Настройка snmptrapd на получение SNMPv3 уведомлений (официальная документация net-snmp) [en]

#### 5 Внутренние механизмы Zabbix

Обзор

Внутренние проверки позволяют наблюдать за внутренними процессами Zabbix. Другими словами, вы можете наблюдать, что происходит с Zabbix сервером или Zabbix прокси.

Внутренние проверки вычисляются:

- на стороне Zabbix сервера — если узел сети наблюдается через сервер
- на стороне Zabbix прокси — если узел сети наблюдается через прокси

Внутренние проверки обрабатываются сервером или прокси вне зависимости от состояния обслуживания узла сети.

Для использования этого элемента данных выберите тип элемента данных **Внутренний Zabbix (Zabbix internal)**.

> **Note:**
> Внутренние проверки обрабатываются Zabbix поллерами.

Производительность

Использование некоторых внутренних элементов данных может отрицательно повлиять на производительность. Речь идёт о следующих элементах данных:

- zabbix[host,,items]
- zabbix[host,,items_unsupported]
- zabbix[hosts]
- zabbix[items]
- zabbix[items_unsupported]
- zabbix[queue,,]

- zabbix[requiredperformance]
- zabbix[stats,,,queue,,]
- zabbix[triggers]

Также это влияет на разделы веб-интерфейса Информация о системе (System information) и Очередь (Queue).

Поддерживаемые проверки

Ключи элементов данных перечислены без настраиваемых параметров и дополнительной информации. Щелкните ключ элемента данных, чтобы увидеть полные сведения.

| Ключ элемента данных | Описание |
|---|---|
| zabbix[boottime] | Время запуска процесса сервера Zabbix или прокси Zabbix в секундах. |
| zabbix[cluster,discovery,nodes] | Обнаруживает узлы кластера высокой доступности. |
| zabbix[connector_queue] | Количество значений, поставленных в очередь connector queue. |
| zabbix[discovery_queue] | Количество сетевых проверок, поставленных в очередь discovery queue. |
| zabbix[host„items] | Количество включенных элементов данных (поддерживаемых и неподдерживаемых) на узле сети. |
| zabbix[host„items_unsupported] | Количество включенных неподдерживаемых элементов данных на узле сети. |
| zabbix[host„maintenance] | Текущий статус обслуживания узла сети. |
| zabbix[host,active_agent,available] | Доступность проверок активного агента на узле сети. |
| zabbix[host,discovery,interfaces] | Сведения обо всех настроенных интерфейсах узла сети в веб-интерфейсе Zabbix. |
| zabbix[host„available] | Доступность основного интерфейса определенного типа проверок на узле сети. |
| zabbix[hosts] | Количество контролируемых узлов сети. |
| zabbix[items] | Количество включенных элементов данных (поддерживаемых и неподдерживаемых). |
| zabbix[items_unsupported] | Количество неподдерживаемых элементов данных. |
| zabbix[java„] | Сведения о Java gateway Zabbix. |
| zabbix[lld_queue] | Количество значений, поставленных в очередь обработки low-level discovery. |
| zabbix[preprocessing] | Статистика значений, полученных менеджером предварительной обработки. |
| zabbix[preprocessing_queue] | Количество значений, поставленных в очередь предварительной обработки. |
| zabbix[process„,] | Процент времени, которое определенный процесс Zabbix или группа процессов (идентифицируемые по <type> и <mode>) провели в состоянии <state>. |
| zabbix[proxy„] | Сведения о прокси Zabbix. |
| zabbix[proxy,discovery] | Список прокси Zabbix. |
| zabbix[proxy group„available] | Количество онлайн-прокси в группе прокси. |
| zabbix[proxy group„pavailable] | Процент онлайн-прокси в группе прокси. |
| zabbix[proxy group„proxies] | Список прокси Zabbix в группе прокси. |
| zabbix[proxy group„state] | Состояние группы прокси. |
| zabbix[proxy group,discovery] | Возвращает список групп прокси с данными конфигурации и данными в реальном времени. |
| zabbix[proxy_buffer,buffer,] | Возвращает статистику использования буфера памяти прокси. |
| zabbix[proxy_buffer,state,changes] | Возвращает количество изменений состояния между режимами буфера disk/memory с момента запуска. |
| zabbix[proxy_buffer,state,current] | Возвращает текущее рабочее состояние, в котором сохраняются новые данные. |
| zabbix[proxy_history] | Количество значений в таблице истории прокси, ожидающих отправки на сервер. |
| zabbix[queue„] | Количество контролируемых элементов данных в очереди, задержанных как минимум на <from> секунд, но менее чем на <to> секунд. |
| zabbix[rcache„] | Статистика доступности кэша конфигурации Zabbix. |
| zabbix[requiredperformance] | Требуемая производительность сервера Zabbix или прокси Zabbix, в ожидаемом количестве новых значений в секунду. |
| zabbix[stats„] | Возвращает внутренние метрики сервера Zabbix или прокси. Если указаны <ip> и <port>, метрики извлекаются с удаленного экземпляра; в противном случае - с локального экземпляра. |
| zabbix[stats„,queue„] | Возвращает внутренние метрики очереди сервера Zabbix или прокси. Если указаны <ip> и <port>, метрики извлекаются с удаленного экземпляра; в противном случае - с локального экземпляра. |
| zabbix[tcache„] | Статистика эффективности кэша функций trend Zabbix. |
| zabbix[triggers] | Количество включенных триггеров в базе данных Zabbix, со всеми включенными элементами данных на включенных узлах сети. |

| Ключ элемента данных | Описание |
|---|---|
| zabbix[uptime] | Время работы процесса сервера Zabbix или прокси в секундах. |
| zabbix[vcache,buffer,] | Статистика доступности кэша значений Zabbix. |
| zabbix[vcache,cache,] | Статистика эффективности кэша значений Zabbix. |
| zabbix[version] | Версия сервера Zabbix или прокси. |
| zabbix[vmware,buffer,] | Статистика доступности кэша vmware Zabbix. |
| zabbix[vps,written] | Общее количество значений истории, записанных в базу данных. |
| zabbix[wcache„] | Статистика и доступность кэша записи Zabbix. |

Подробности ключа элемента данных

- Параметры без угловых скобок являются обязательными и должны использоваться *как есть* (например, «host» и «available» в `zabbix[host,< >,available]`).
- Параметры с угловыми скобками **< >** должны заменяться корректными значениями. Если параметр имеет значение по умолчанию, его можно опустить.
- Значения элементов данных и параметров элементов данных с пометкой «не поддерживаются на прокси» можно собирать только на узлах сети, которые наблюдаются через сервер. И наоборот, значения «не поддерживается на сервере» можно собирать, только если узел сети наблюдается через прокси.

zabbix[boottime]

<br> Время запуска процесса Zabbix сервера или Zabbix прокси в секундах.<br> Возвращаемое значение: *Целое число*.

zabbix[cluster,discovery,nodes]

<br> Обнаружение нод кластера высокой доступности.<br> Возвращаемое значение: *Объект JSON*.

Комментарии:

- Этот элемент данных можно использовать в низкоуровневом обнаружении.
- Возвращаемые поля в основном соответствуют свойствам объекта ноды высокой доутспности (High availability node).
- Дополнительные поля:
  - `db_timestamp` — текущее время сервера (Unix timestamp), включённое в ответ
  - `lastaccess_age` — секунд с последнего сигнала heartbeat ноды (db_timestamp - lastaccess)
  - `status` — состояние ноды:
    - 0 — горячий резерв (standby)
    - 1 — остановлен вручную
    - 2 — недоступен
    - 3 — активен

Пример возвращаемых значений:

```
[
  {
    "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]

<br> Количество значений, поставленных в очередь коннектора..<br> Возвращаемое значение: *Целое число*.

zabbix[discovery_queue]

<br> Количество сетевых проверок, поставленных в очередь обнаружения.<br> Возвращаемое значение: *Целое число*.

zabbix[host,,items]

<br> Количество активированных элементов данных (поддерживаемых и неподдерживаемых) у узла сети.<br> Возвращаемое значение: *Целое число*.

zabbix[host,,items_unsupported]

<br> Количество активированных неподдерживаемых элементов данных у узла сети.<br> Возвращаемое значение: *Целое число*.

zabbix[host,,maintenance]

<br> Текущее состояние обслуживания узла сети.<br> Возвращаемое значение: *0* — нормальное состояние; *1* — обслуживание со сбором данных; *2* — обслуживание без сбора данных.

Комментарии:

- Этот элемент данных всегда обрабатывается Zabbix сервером вне зависимости от настроек узла сети (наблюдается через сервер или прокси). Прокси не получает этот элемент данных при получении своей конфигурации.
- Второй параметр должен быть пустым и зарезервирован для использования в будущем.

zabbix[host,active_agent,available]

<br> Доступность активных проверок агента на узле сети.<br> Возвращаемые значения: *0* — неизвестно; *1* — доступно; *2* — недоступно.

zabbix[host,discovery,interfaces]

<br> Сведения по всем настроенным интерфейсам узла сети в веб-интерфейсе Zabbix.<br> Возвращаемое значение: *Объект JSON*.

Комментарии:

- Этот элемент данных можно использовать в низкоуровневом обнаружении.
- Этот элемент данных не поддерживается на Zabbix прокси.

zabbix[host,<тип>,available]

<br> Доступность основного интерфейса определённого типа проверок на узле сети.<br> Возвращаемые значения: *0* — недоступен; *1* — доступен; *2* — неизвестно.

Параметры:

- **тип** — *agent*, *snmp*, *ipmi* или *jmx*.

Комментарии:

- Значение элемента данных вычисляется согласно параметрам конфигурации доступности/недоступности касательно узлов сети.

zabbix[hosts]

<br> Количество наблюдаемых узлов сети.<br> Возвращаемое значение: *Целое число*.

zabbix[items]

<br> Количество активированных элементов данных (поддерживаемых и неподдерживаемых).<br> Возвращаемое значение: *Целое число*.

zabbix[items_unsupported]

<br> Количество неподдерживаемых элементов данных.<br> Возвращаемое значение: *Целое число*.

zabbix[java„<param>]

<br> Информация о Zabbix Java gateway.<br> Возвращаемые значения: *1* - если <param> равно *ping*; *версия Java gateway* - если <param> равно *version* (например: ”7.4.0”).

Параметры:

- **param** - *ping* или *version*.

Комментарии:

- Этот элемент данных можно использовать для проверки доступности Java gateway с помощью функции триггера `nodata()`.
- Второй параметр должен быть пустым и зарезервирован для будущего использования.

zabbix[lld_queue]

<br> Количество значений, помещённых в очередь обработки низкоуровневых обнаружений.<br> Возвращаемое значение: *Целое число*.

Комментарии:

- Этот элемент данных можно использовать для мониторинга размера очереди обработки низкоуровневых обнаружений.

zabbix[preprocessing]

<br> Статистика значений, полученных менеджером предварительной обработки:

- *queued* - количество и размер значений в очереди, требующих предварительной обработки (счетчик)
- *direct* - количество и размер значений в очереди, не требующих предварительной обработки (счетчик)
- *queue* - количество значений, добавленных в очередь предварительной обработки (то же, что и `zabbix[preprocessing_queu`

Возвращаемое значение: *JSON*.

Пример возвращаемых значений:

```
{"data":
    {
    "queued": {
        "count": 106,
        "size": 58620
    },
    "direct": {
        "count": 395,
        "size": 33843
    },
    "queue": 0
    }
}
```

zabbix[preprocessing_queue]

<br> Количество значений, помещённых в очередь предварительной обработки.<br> Возвращаемое значение: *Целое число*.

Комментарии:

- Этот элемент данных можно использовать для мониторинга размера очереди предварительной обработки.

zabbix[process,<type>,<mode>,<state>]

<br> Процент времени, который определенный процесс Zabbix или группа процессов (идентифицируемые по <type> и <mode>) провели в состоянии <state>. Вычисляется только за последнюю минуту.<br> Возвращаемое значение: *Float*.

Параметры:

- **type** - для процессов сервера: *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*;<br>для процессов прокси: *agent poller*, *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* - среднее значение для всех процессов указанного типа (по умолчанию);<br>*count* - возвращает количество fork-процессов для указанного типа процесса, <state> не должен быть указан;<br>*max* - максимальное значение;<br>*min* - минимальное значение;<br><*process number*> - номер процесса (от 1 до количества предварительно созданных экземпляров; например, если запущено 4 trapper, значение находится в диапазоне от 1 до 4);
- **state** - *busy* - процесс находится в состоянии занятости, например, обрабатывает запрос (по умолчанию);<br>*idle* - процесс находится в состоянии простоя и ничего не делает.

Комментарии:

- Если <mode> — это номер процесса Zabbix, который не запущен (например, при 5 запущенных poller значение <mode> указано как 6), такой элемент данных станет неподдерживаемым.
- Минимум и максимум относятся к проценту использования одного процесса. Например, если в группе из 3 poller процент использования по процессам был 2, 18 и 66, min вернет 2, а max вернет 66.
- Процессы сообщают о том, что они делают, в разделяемой памяти, а процесс self-monitoring каждую секунду суммирует эти данные. Изменения состояния (busy/idle) регистрируются в момент изменения - таким образом, процесс, который становится busy, регистрируется как таковой и не меняет и не обновляет состояние, пока не станет idle. Это гарантирует, что даже полностью зависшие процессы будут корректно зарегистрированы как busy на 100%.
- В настоящее время "busy" означает "не спит", но в будущем могут быть введены дополнительные состояния - ожидание блокировок, выполнение запросов к базе данных и т. д. Обратите внимание, что асинхронные poller считаются busy, если они достигли лимита, заданного конфигурационным параметром MaxConcurrentChecksPerPoller сервер/прокси.
- В Linux и большинстве других систем разрешение составляет 1/100 секунды.

Примеры:

```
zabbix[process,poller,avg,busy] #        ,           poller
zabbix[process,"icmp pinger",max,busy] #          ,              ICMP pinger
zabbix[process,"history syncer",2,busy] #   ,          history syncer     2
zabbix[process,trapper,count] #            trapper,
```

zabbix[proxy,<имя>,<параметр>]

<br> Информация о Zabbix прокси.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **имя** — имя прокси;
- **параметр** — *lastaccess* — штамп времени последнего сообщения о доступности от прокси;<br>*delay* — длительность задержки отправки собранных значений, рассчитывается как «задержка прокси» + («текущее время сервера» - «последний доступ прокси»), где «задержка прокси» — это разница между текущим временем прокси и отметкой времени самого старого неотправленного значения на прокси.

Комментарии:

- Этот элемент данных всегда обрабатывается Zabbix сервером, независимо от расположения узла сети (на сервере или прокси).
- Можно использовать функцию `fuzzytime()` для проверки доступности прокси.

Примеры:

```
zabbix[proxy,"Germany",lastaccess] #                                ,              «Germany»
```

zabbix[proxy,discovery]

<br> Список прокси-серверов Zabbix с именем, режимом, шифрованием, сжатием, версией, последним контактом, количеством узлов сети, количеством элементов данных, требуемыми значениями в секунду (vps), статусом версии (текущая/устаревшая/неподдерживаемая), тайм-аутами по типам элементов данных, именем группы прокси (если прокси принадлежит группе), состоянием (неизвестно/офлайн/онлайн).<br> Возвращаемое значение: *Объект JSON*.

zabbix[proxy group,<имя>,available]

<br> Количество онлайн-прокси в группе прокси.<br> Возвращаемое значение: *Целое число*

Параметры:

- **имя** — имя группы прокси.

zabbix[proxy group,<имя>,pavailable]

<br> Процент онлайн-прокси в группе прокси.<br> Возвращаемое значение: *Число с плавающей точкой*.

Параметры:

- **имя** — имя группы прокси.

zabbix[proxy group,<имя>,proxies]

<br> Список прокси-серверов Zabbix в группе прокси с именем, режимом, шифрованием, сжатием, версией, последним контактом, количеством узлов сети, количеством элементов данных, требуемыми значениями в секунду (vps), статусом версии (текущая/устаревшая/неподдерживаемая), тайм-аутами, именем группы прокси, состоянием (неизвестно/офлайн/онлайн).<b
Возвращаемое значение: *Объект JSON*.

Параметры:

- **имя** — имя группы прокси.

zabbix[proxy group,<имя>,state]

<br> Состояние группы прокси.<br> Возвращаемое значение: *0* — неизвестно; *1* — офлайн; *2* — восстановление; *3* — онлайн; *4* — деградация.

Параметры:

- **имя** — имя группы прокси.

zabbix[proxy group,discovery]

<br> Возвращает список групп прокси с данными конфигурации и данными в реальном времени. Данные конфигурации включают имя группы прокси, задержку failover и минимальное количество онлайн-прокси, необходимое для работы. Данные в реальном времени включают состояние группы прокси (подробности см. в комментариях), количество онлайн-прокси и процент онлайн-прокси.<br> Возвращаемое значение: *JSON*.

Комментарии:

- Этот элемент данных не возвращает прокси без группы.
- Если для "failover_delay" или "min_online" указано недопустимое значение, то для его обозначения возвращается специальное значение *-1*. Недопустимые значения могут возникать, если для конфигурации используются макросы, а макросы не могут быть раскрыты в допустимое значение.
- Состояние группы прокси возвращается как целое число: *0* - неизвестно; *1* - офлайн; *2* - восстановление; *3* - онлайн; *4* - ухудшение.

Пример возвращаемых значений:

```
{
    "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,<режим>]

<br> Статистика использования буфера памяти прокси.<br> Возвращаемое значение: *Целое число* (для размера); *Число с плавающей точкой* (для процентов).

Параметры:

- **режим**: *total* — общий размер буфера (можно использовать для проверки, включён ли буфер памяти);<br>*free* — размер свободного буфера;<br>*pfree* — процент свободного буфера;<br>*used* — размер используемого буфера;<br>*pused* — процент используемого буфера.

Комментарии:

- Возвращает ошибку «Буфер памяти прокси отключён» (Proxy memory buffer is disabled), когда буфер памяти отключён;<br>
- Этот элемент данных не поддерживается на сервере Zabbix.

zabbix[proxy_buffer,state,changes]

<br> Возвращает количество изменений состояния между режимами буфера диск/память с момента запуска.<br> Возвращаемое значение: *Целое число*; *0* - буфер памяти отключён.

Комментарии:

- Частые изменения состояния указывают на то, что необходимо увеличить размер или возраст буфера памяти.
- Если состояние буфера памяти отслеживается нечасто (например, раз в минуту), то буфер может изменить своё состояние, не отметив этого.

zabbix[proxy_buffer,state,current]

<br> Возвращает текущее рабочее состояние, в котором сохраняются новые данные.<br> Возвращаемые значения: *0* - диск; *1* - память.

Комментарии:

- ”0” также возвращается, когда буфер памяти отключен.

zabbix[proxy_history]

<br> Количество значений в таблице истории прокси, ожидающих отправку на сервер.<br> Возвращаемое значение: *Целое число*.

Комментарии:

- Этот элемент данных не поддерживается на сервере Zabbix.

zabbix[queue,<от>,<до>]

<br> Количество наблюдаемых элементов данных в очереди, которые задерживаются минимум <от>, но менее <до> секунд.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **от** — задерживаются минимум (по умолчанию: 6 секунд);
- **до** — задерживаются максимум (по умолчанию: бесконечно).

Комментарии:

- В этих параметрах поддерживаются суффиксы времени (s,m,h,d,w).

**Attention:**
Прокси-сервер Zabbix не знает о периодах обслуживания; подробности смотрите в Расчёт очередей во время обслуживания.

zabbix[rcache,<кэш>,<режим>]

<br> Статистика доступности кэша конфигурации Zabbix.<br> Возвращаемое значение: *Целое число* (для размера); *Число с плавающей точкой* (для процентов).

Параметры:

- **кэш** — *buffer*;
- **режим** — *total* — полный размер буфера;<br>*free* — размер свободного места в буфере;<br>*pfree* — процент свободного места в буфере;<br>*used* — размер использованного места в буфере;<br>*pused* — процент использования буфера.

zabbix[requiredperformance]

<br> Требуемое быстродействие Zabbix сервера или прокси, ожидаемых новых значений в секунду.<br> Возвращаемое значение: *Число с плавающей точкой*.

Комментарии:

- Приблизительно соответствует значению «Требуемое быстродействие сервера, новые значения в секунду (Required server performance, new values per second)» со страницы *Отчёты* → *Информация о системе* *(Reports → System information)*.

zabbix[stats,<ip>,<порт>]

<br> Возвращает внутренние метрики Zabbix сервера или прокси. Если указаны <ip> и <порт>, метрики извлекаются с удалённого экземпляра; в противном случае — с локального.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **ip** — список IP/DNS/сетевой маски опрашиваемых удалённо серверов/прокси (по умолчанию 127.0.0.1);
- **порт** — порт опрашиваемого удалённо сервера/прокси (по умолчанию 10051).

Комментарии:

- Запрос статистики будет приниматься только с адресов, которые перечислены в параметре «StatsAllowedIP» сервера/прокси на удалённом экземпляре.
- Этим элементом данных возвращается выбранный список внутренних метрик. Для получения более подробных сведений смотрите удалённый мониторинг статистики Zabbix.

zabbix[stats,<ip>,<порт>,queue,<от>,<до>]

<br> Возвращает метрики внутренней очереди Zabbix сервера или прокси (смотрите `zabbix[queue,< >,< >]`). Если указаны <ip> и <порт>, метрики извлекаются с удалённого экземпляра; в противном случае — с локального.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **ip** — список IP/DNS/сетевой маски опрашиваемых удалённо серверов/прокси (по умолчанию 127.0.0.1);
- **порт** — порт опрашиваемого удалённо сервера/прокси (по умолчанию 10051);
- **от** — с задержкой по крайней мере (по умолчанию от 6 секунд);
- **до** — с задержкой не более (по умолчанию до бесконечности).

Комментарии:

- Запрос статистики будет приниматься только с адресов, которые перечислены в параметре «StatsAllowedIP» сервера/прокси на целевом экземпляре.
- Этим элементом данных возвращается выбранный список внутренних метрик. Для получения более подробных сведений смотрите удалённый мониторинг статистики Zabbix.

**Attention:**
Прокси-сервер Zabbix не знает о периодах обслуживания; подробности смотрите в Расчёт очередей во время обслуживания.

zabbix[tcache,cache,<параметр>]

<br> Статистика эффективности кэша Zabbix функций динамики изменений.<br> Возвращаемое значение: *Целое число* (для размера); *Число с плавающей точкой* (для процентов).

Параметры:

- **параметр** — *all* — общее количество запросов кэша (по умолчанию);<br>*hits* — количество попаданий в кэш;<br>*phits* — процент попаданий в кэш;<br>*misses* — количество непопаданий в кэш;<br>*pmisses* — процент непопаданий в кэш;<br>*items* — количество элементов данных в кэше;<br>*requests* — количество кэшируемых запросов;<br>*pitems* — процент элементов данных в кэше из кэшируемых элементов данных + запросов. Низкий процент зачастую означает, что размер кэша можно уменьшить.

Комментарии:

- Этот элемент данных не поддерживается на Zabbix прокси.

zabbix[triggers]

<br> Количество активированных триггеров в базе данных Zabbix, со всеми активированными элементами данных, которые упомянуты в выражениях триггеров на активированных узлах сети.<br> Возвращаемое значение: *Целое число*.

Комментарии:

- Этот элемент данных не поддерживается на Zabbix прокси.

zabbix[uptime]

<br> Время непрерывной работы процесса Zabbix сервера или прокси в секундах.<br> Возвращаемое значение: *Целое число*.

zabbix[vcache,buffer,<режим>]

<br> Статистика доступности кэша значений Zabbix.<br> Возвращаемое значение: *Целое число* (для размера); *Число с плавающей точкой* (для процентов).

Параметры:

- **режим** — *total* — полный размер буфера;<br>*free* — размер свободного места в буфере;<br>*pfree* — процент свободного места в буфере;<br>*used* — размер использованного места в буфере;<br>*pused* — процент использованного места в буфере.

Комментарии:

- Этот элемент данных не поддерживается на Zabbix прокси.

zabbix[vcache,cache,<параметр>]

<br> Статистика эффективности кэша значений Zabbix.<br> Возвращаемое значение: *Целое число*. Если <параметр> имеет значение *mode*, то возвращается: *0* — нормальный режим; *1* — режим нехватки памяти.

Параметры:

- **параметр** — *requests* — общее количество запросов;<br>*hits* — количество попаданий в кэш (значения истории берутся из кэша);<br>*misses* — количество непопаданий в кэш (значения истории берутся из базы данных);<br>*mode* — режим работы кэша значений.

Комментарии:

- После переключения в режим нехватки памяти, кэш значений будет оставаться в этом состоянии в течение 24 часов — даже если проблема, вызвавшая этот режим, будет решена раньше.
- Вы можете использовать этот ключ с шагом предобработки *Изменение в секунду* при необходимости получения статистики значений в секунду.
- Этот элемент данных не поддерживается на Zabbix прокси.

zabbixversion

<br> Версия сервера или прокси Zabbix.<br> Возвращаемое значение: *String*. Например: `7.4.0`.

zabbix[vmware,buffer,<режим>]

<br> Статистика доступности Zabbix кэша VMware.<br> Возвращаемое значение: *Целое число* (для размера); *Число с плавающей точкой* (для процентов).

Параметры:

- **режим** — *total* — полный размер буфера;<br>*free* — размер свободного места в буфере;<br>*pfree* — процент свободного места в буфере;<br>*used* — размер использованного места в буфере;<br>*pused* — процент использованного места в буфере.

zabbix[vps,written]

<br> Общее количество значений истории, записанных в базу данных.<br> Возвращаемое значение: *Целое число*.

zabbix[wcache,<кэш>,<режим>]

<br> Статистика и доступность кэша записи Zabbix.<br> Возвращаемое значение: *Целое число* (для количества/размера); *Число с плавающей точкой* (для процентов).

Параметры:

- **кэш** — *values*, *history*, *index* или *trend*;
- **режим** — (для *values*) *all* (по умолчанию) — общее количество значений, обработанных Zabbix сервером или Zabbix прокси, исключая неподдерживаемые элементы данных (счётчик);<br>*float* — количество обработанных значений с плавающей точкой (счётчик);<br>*uint* — количество обработанных целочисленных значений (счётчик);<br>*str* — количество обработанных символьных/строковых значений (счётчик);<br>*log* — количество обработанных значений из файлов журналов (счётчик);<br>*text* — количество обработанных текстовых значений (счётчик);<br>*bin* — количество обработанных бинарных значений (счётчик);<br>*not supported* — количество раз, когда обработка элементов данных приводила к тому, что элемент данных становился неподдерживаемым или сохранял это состояние (счётчик);<br>(для параметра кэша *history*, *index*, *trend*) *pfree* (по умолчанию) — свободное место в буфере в процентах;<br>*total* — полный размер буфера;<br>*free* — размер свободного места в буфере;<br>*used* — размер занятого места в буфере;<br>*pused* — занятое место в буфере в процентах.

Комментарии:

- Параметр кэша `trend` не поддерживается на Zabbix прокси.
- Кэш истории используется для хранения значений элементов данных. Низкое количество свободного места отражает проблемы с производительностью на стороне базы данных.

- Кэш индексов истории используется для индексирования значений, записываемых в кэш истории.
- После того, как кэш истории будет заполнен и затем очищен, в кэше индекса истории все равно будут находиться некоторые данные. Такое поведение ожидаемо и помогает системе работать более эффективно, избегая дополнительной обработки, необходимой для постоянного изменения размера памяти.
- Кэш динамики изменений агрегируется за текущий час для всех элементов данных, которые получают данные.
- Вы можете использовать ключ zabbix[wcache,values] с шагом предобработки *Изменение в секунду* при необходимости получения статистики значений в секунду.

### 6 Zabbix trapper

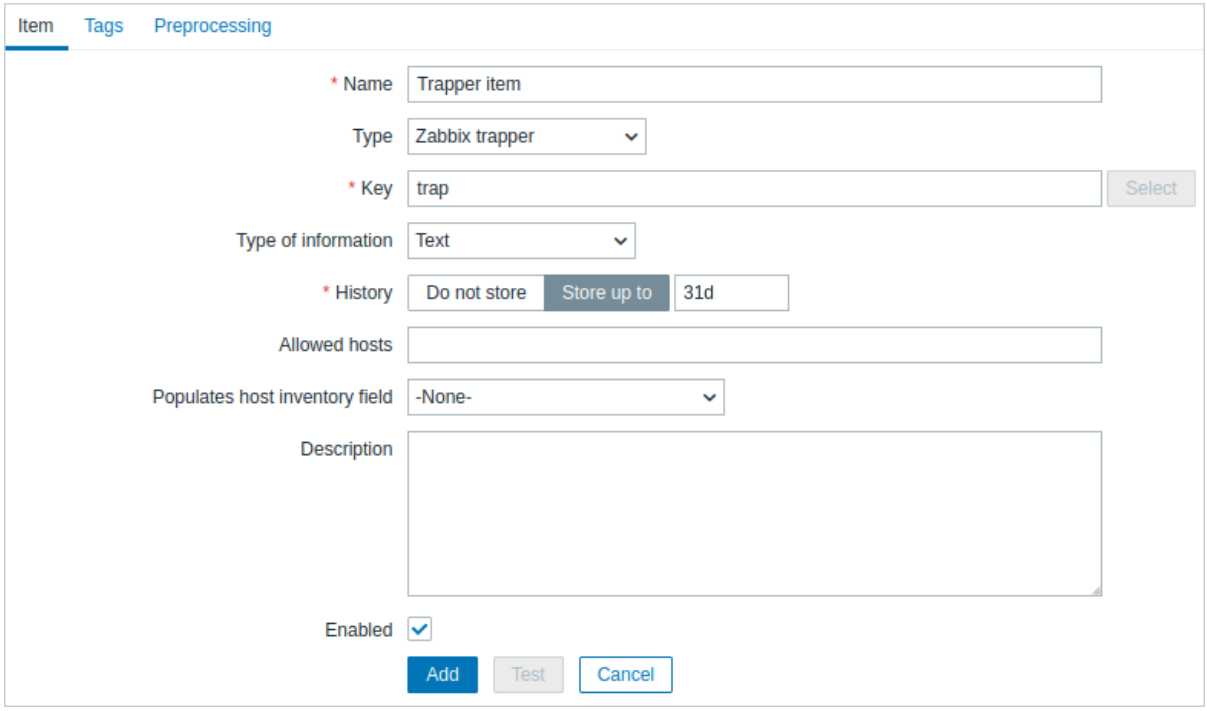
Обзор

Траппер элементы данных принимают входящие данные вместо опроса этих данных. Это полезно для любых данных, которые вы хотите отправить в Zabbix.

Настройка

Чтобы настроить траппер элемент данных:

1. Перейдите в *Сбор данных* → *Узлы сети*.
2. Нажмите *Элементы данных* в строке узла сети.
3. Нажмите *Создать элемент данных*.
4. Введите параметры элемента данных в диалоге настроек.




Все обязательные поля ввода отмечены красной звездочкой.

Поля, которые требуют указания информации, специфичной для траппер элемента данных:

| | |
|---|---|
| *Тип (Type)* | Здесь выберите «Zabbix траппер (Zabbix trapper)». |
| *Ключ (Key)* | Введите ключ, который будет использован для распознавания элемента данных при отправке данных. |
| *Тип информации (Type of information)* | Выберите тип информации, который будет соответствовать формату отсылаемых данных. |

| | |
|---|---|
| *Разрешённые хосты (Allowed hosts)* | Список разделённых запятыми IP адресов (опционально в CIDR представлении) или DNS имён.<br><br>Если задано, входящие подключения будут приниматься только с перечисленных здесь хостов.<br>Если включена поддержка IPv6, тогда «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» обрабатываются одинаково, а «::/0» разрешает любой адрес IPv4 или IPv6. «0.0.0.0/0» можно использовать, чтобы разрешить любой адрес IPv4.<br>Обратите внимание, что «IPv4-совместимые IPv6 адреса» (0000::/96 префикс) поддерживаются, но являются устаревшими согласно RFC4291 [en].<br><br>Пример: 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}<br><br>Поддерживаются пробелы, пользовательские макросы и макросы узлов сети {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN}. |

**Note:**
Перед отправкой значений, возможно, придется подождать до 60 секунд после сохранения элемента данных, пока сервер Zabbix подхватит изменения из обновлённого кэша конфигурации.

Отправка данных

Отправка данных на Zabbix сервер или прокси возможна с помощью утилиты Zabbix sender или протокола Zabbix sender. Отправка данных на сервер Zabbix также возможна с использованием метода API `history.push`.

**Zabbix sender**

Для отправки данных на Zabbix сервер или прокси с помощью утилиты Zabbix sender вы можете запустить следующую команду, чтобы отправить некоторое «тестовое значение»:

```
zabbix_sender -z <IP          > -p 10051 -s "New host" -k trap -o "          "
```

Чтобы отправить «тестовое значение», используются следующие опции команды:

- `-z` — чтобы указать IP адрес Zabbix сервера
- `-p` — чтобы указать номер порта Zabbix сервера (по умолчанию 10051)
- `-s` — чтобы указать узел сети (убедитесь, что используете здесь «техническое» имя узла сети вместо «видимого» имени)
- `-k` — чтобы указать ключ элемента данных, настроенного в траппер элементе данных
- `-o` — чтобы указать отправляемое значение

**Attention:**
Процесс Zabbix траппера не раскрывает макросы, которые используются в ключе элемента данных при проверке наличия соответствующего ключа элемента данных на целевом узле сети.

Более подробную информацию о коммуникации между Zabbix sender'ом и Zabbix сервером или прокси смотрите в *протоколе Zabbix sender'а*.

**history.push**

Для отправки данных на сервер Zabbix с помощью метода API `history.push` вы можете сделать следующий запрос HTTP POST, содержащий некоторые тестовые значения:

```
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
```

Если запрос корректный, то ответ, возвращаемый API, может выглядеть следующим образом:

```
{
    "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
}
```


Ошибки в данных ответа указывают на то, что отправка данных для определённых элементов данных не прошла проверку сервером Zabbix. Это может произойти по следующим причинам:

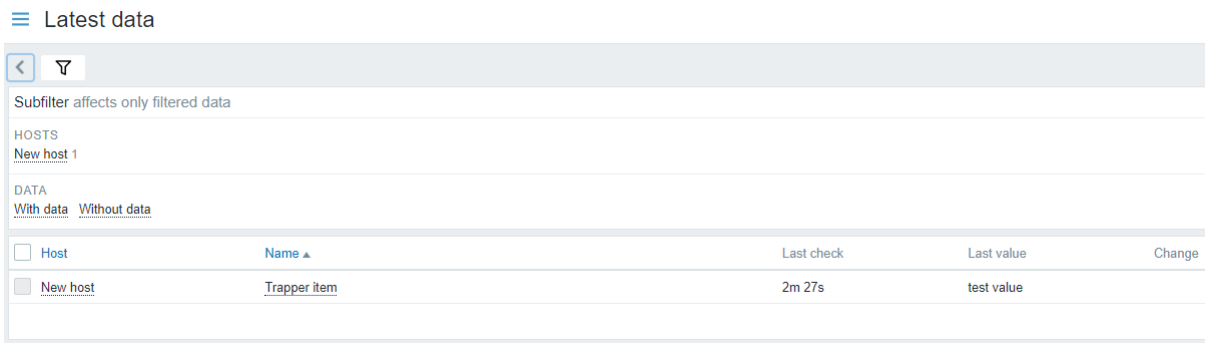
- пользователь, отправляющий данные, не имеет права *чтения* на узел сети элемента данных;
- узел отключён или находится в состоянии обслуживания без сбора данных;
- элемент данных не существует или ещё не включён в кэш конфигурации сервера;
- элемент данных отключён или его тип отличается от Zabbix траппер или HTTP агент (с включённой опцией «Активировать трапы»);
- IP или DNS пользователя не указаны в списке *Разрешённые хосты* элемента данных;
- другой элемент данных имеет значение с дублирующейся временной меткой на уровне наносекунд.

Отсутствие ошибок указывает на то, что отправленные значения были приняты для обработки, которая включает предобработку (если таковая имеется), обработку триггера и сохранение в базе данных. Обратите внимание, что обработка принятого значения также может завершиться ошибкой (например, во время предобработки), в результате чего значение будет отброшено.

Более подробную информацию о работе с API Zabbix смотрите в API.

Отображение данных

После отправки данных вы можете перейти к *Мониторинг* → *Последние данные*, чтобы увидеть результат:




**Note:**
Если отправлено одно числовое значение, на графике отобразится горизонтальная линия с левого до правого края от точки времени значения.

### 7 Внешняя проверка

Обзор

Внешняя проверка — это проверка, выполняемая сервером Zabbix путем запуска shell-скрипта или бинарного файла. Однако, если узлы сети контролируются через прокси Zabbix, внешние проверки выполняются прокси.

Внешние проверки не требуют запущенного агента на контролируемом узле сети.

Синтаксис ключа элемента данных:

`script[<parameter1>,<parameter2>,...]`

Где:

| ARGUMENT | DEFINITION |
|---|---|
| **script** | Имя shell-скрипта или бинарного файла. |
| **parameter(s)** | Необязательные параметры командной строки. |

Если вы не хотите передавать какие-либо параметры скрипту, можно использовать:

```
script[] or
script
```

Сервер Zabbix или прокси выполнит поиск в каталоге, указанном для внешних скриптов, и запустит команду (см. параметр `ExternalScripts` в файле конфигурации Zabbix server/proxy). Команда будет выполнена от имени того же пользователя, что и сервер Zabbix/прокси, поэтому при необходимости любые права доступа или переменные окружения следует обрабатывать в обертке-скрипте. Права доступа к команде также должны позволять этому пользователю выполнять ее. Для выполнения доступны только команды в указанном каталоге.

**Warning:**
Не злоупотребляйте внешними проверками, поскольку для каждого скрипта серверу Zabbix/прокси требуется запускать fork-процесс, и выполнение большого количества скриптов может значительно снизить производительность Zabbix.

Пример использования

Выполнение скрипта **check_oracle.sh** с первым параметром «-h». Второй параметр будет заменён IP адресом или DNS именем узла сети, в зависимости от выбранного в настройках узла сети.

```
check_oracle.sh["-h","{HOST.CONN}"]
```

Предположим, что узел сети настроен на использование IP адреса, тогда Zabbix сервер/прокси выполнит:

```
check_oracle.sh '-h' '192.168.1.4'
```

Результат внешней проверки

Возвращаемым значением внешней проверки является стандартный вывод вместе со стандартным выводом ошибок, производимые проверкой.

**Attention:**
Элемент данных, возвращающий текст (символ, журнал или текстовый типы данных), не станет неподдерживаемым в случае появления чего-либо в стандартном выводе ошибок.

Возвращаемое значение ограничено 16 МБ (включая конечные пробелы, которые усекаются); также применяются ограничения базы данных.

В случае, если выполняемый скрипт не найден или Zabbix сервер/прокси не имеет необходимых прав на его запуск, элемент данных станет неподдерживаемым и отобразится соответствующее сообщение об ошибке.

В случае превышения времени ожидания элемент данных также станет неподдерживаемым, будет отображено соответствующее сообщение об ошибке, а процесс, порождённый для этого скрипта, будет ликвидирован.

## 8 Монитор базы данных

Обзор

Элемент данных типа *Database monitor* в веб-интерфейсе Zabbix используется для мониторинга через ODBC (проверки ODBC).

ODBC — это API промежуточного программного обеспечения на языке программирования C для доступа к системам управления базами данных (DBMS). Концепция ODBC была разработана Microsoft и позже перенесена на другие платформы.

Zabbix может выполнять запросы к любой базе данных, поддерживаемой ODBC. Для этого Zabbix не подключается к базам данных напрямую, а использует интерфейс ODBC и драйверы, настроенные в ODBC. Это позволяет более эффективно выполнять мониторинг различных баз данных для разных целей (например, проверять определенные очереди базы данных, статистику использования и т. д.).

Zabbix поддерживает unixODBC, одну из наиболее часто используемых реализаций API ODBC с открытым исходным кодом.

**Attention:**
См. также: известные проблемы для проверок ODBC.

Установка unixODBC

Рекомендуемый способ установки unixODBC — использовать стандартные репозитории пакетов операционной системы Linux. В большинстве популярных дистрибутивов Linux unixODBC по умолчанию включен в репозиторий пакетов. Если пакеты недоступны, исходные файлы можно получить на домашней странице unixODBC: http://www.unixodbc.org/download.html.

Чтобы установить unixODBC, используйте менеджер пакетов для выбранной вами системы:

```
#####        Ubuntu/Debian:
apt install unixodbc unixodbc-dev

#####             RedHat/Fedora:
dnf install unixODBC unixODBC-devel

#####             SUSE:
zypper in unixODBC-devel
```

**Attention:**
Пакет `unixodbc-dev` или `unixODBC-devel` необходим для компиляции Zabbix с поддержкой unixODBC. Чтобы включить поддержку ODBC, Zabbix следует собирать со следующей опцией конфигурации: <br><br>
`--with-unixodbc[=ARG] #              ODBC         unixODBC.`

Установка драйверов unixODBC

Драйвер unixODBC базы данных должен быть установлен для базы данных, которая будет наблюдаться. Для списка поддерживаемых баз данных и драйверов обратитесь к домашней странице unixODBC: http://www.unixodbc.org/drivers.html.

**Note:**
В некоторых дистрибутивах Linux драйвера баз данных включены в репозитории пакетов.

MySQL

Чтобы установить драйвер unixODBC базы данных MySQL, используйте менеджер пакетов для выбранной вами системы:

```
#####        Ubuntu/Debian:
apt install odbc-mariadb

#####             RedHat/Fedora:
dnf install mariadb-connector-odbc

#####             SUSE:
zypper install mariadb-connector-odbc
```

Чтобы установить драйвер базы данных без менеджера пакетов, обратитесь к документации MySQL [en] для `mysql-connector-odbc`, или документации MariaDB [en] для `mariadb-connector-odbc`.

PostgreSQL

Чтобы установить драйвер unixODBC базы данных PostgreSQL, используйте менеджер пакетов для выбранной вами системы:

```
#####        Ubuntu/Debian:
apt install odbc-postgresql

#####             RedHat/Fedora:
dnf install postgresql-odbc

#####             SUSE:
zypper install psqlODBC
```

Чтобы установить драйвер базы данных без менеджера пакетов, обратитесь к документации PostgreSQL [en].

Oracle

Чтобы установить драйвер базы данных unixODBC, обратитесь к документации Oracle [en].

MSSQL

Чтобы установить драйвер unixODBC базы данных MSSQL, используйте менеджер пакетов для выбранной вами системы:

```
#####        Ubuntu/Debian:
apt install tdsodbc

#####             RedHat/Fedora (    EPEL: https://docs.fedoraproject.org/en-US/epel/):
dnf install epel-release
dnf install freetds

#####             SUSE:
zypper install libtdsodbc0
```

Чтобы установить драйвер базы данных без менеджера пакетов, обратитесь к руководству пользователя FreeTDS [en].

Настройка unixODBC

Чтобы настроить unixODBC, необходимо отредактировать файлы `odbcinst.ini` и `odbc.ini`. Вы можете проверить расположение этих файлов, выполнив следующую команду:

```
odbcinst -j
```

Результат команды должен содержать информацию наподобие следующей:

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

odbcinst.ini

Файл `odbcinst.ini` содержит список установленных драйверов баз данных ODBC. Если `odbcinst.ini` отсутствует, его необходимо создать вручную.

```
[TEST_MYSQL]
Description=ODBC for MySQL
Driver=/usr/lib/libmyodbc5.so
FileUsage=1
```

| Параметр | Описание |
|---|---|
| TEST_MYSQL | Имя драйвера базы данных. |
| Description | Описание драйвера базы данных. |
| Driver | Местоположение библиотеки драйвера базы данных. |
| FileUsage | Определяет, поддерживает ли драйвер базы данных подключение к серверу базы данных без поддержки доступа к локальным файлам (0); поддерживает чтение данных из файлов (1); поддерживает запись данных в файлы (2). |
| Threading | Уровень сериализации потоков. Поддерживается для PostgreSQL.<br>Начиная с версии 1.6, если менеджер драйверов собран с поддержкой потоков, вы можете добавить еще одну запись драйвера. |

odbc.ini

Файл `odbc.ini` используется для настройки источников данных. Обратите внимание, что список поддерживаемых параметров зависит от драйвера базы данных (например, базы данных Oracle могут использовать ServerName вместо Server и т. д.).

```
[TEST_MYSQL]
Description=MySQL Test Database
Driver=mysql
Server=127.0.0.1
User=root
Password=
```

```
Port=3306
Socket=
Database=zabbix
```

| Параметр | Описание |
|---|---|
| TEST_MYSQL | Имя источника данных (Data source name, DSN). |
| Description | Описание источника данных. |
| Driver | Имя драйвера базы данных (как указано в `odbcinst.ini`). |
| Server | IP/DNS сервера базы данных. |
| User | Пользователь базы данных для подключения. |
| Password | Пароль пользователя базы данных. |
| Port | Порт подключения к базе данных. |
| Socket | Сокет подключения к базе данных. |
| Database | Имя базы данных. |

Другие возможные параметры конфигурации смотрите в документации MySQL [en].

Файл `odbc.ini` для PostgreSQL может содержать дополнительные параметры:

```
[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=7.4+
ShowOidColumn=No
FakeOidIndex=No
RowVersioning=No
ShowSystemTables=No
Fetch=Yes
BoolsAsChar=Yes
SSLmode=Require
ConnSettings=
```

| Параметр | Описание |
|---|---|
| ReadOnly | Указывает, разрешает ли подключение к базе данных только операции чтения (запросы `SELECT`) и ограничивает ли изменения (операторы `INSERT`, `UPDATE` и `DELETE`); полезно для сценариев, где данные должны оставаться неизменными. |
| Protocol | Версия внутреннего протокола PostgreSQL (игнорируется при использовании SSL-соединений). |
| ShowOidColumn | Указывает, следует ли включать идентификатор объекта (OID) в SQLColumns. |
| FakeOidIndex | Указывает, следует ли создавать фиктивный уникальный индекс для OID. |
| RowVersioning | Указывает, следует ли разрешить приложениям определять, были ли данные изменены другими пользователями при попытке обновления строки. Обратите внимание, что этот параметр может ускорить процесс обновления, поскольку для обновления строки не нужно указывать каждый отдельный столбец в условии `WHERE`. |
| ShowSystemTables | Указывает, должен ли драйвер базы данных обрабатывать системные таблицы как обычные таблицы в SQLTables; полезно для обеспечения доступности, позволяя просматривать системные таблицы. |
| Fetch | Указывает, должен ли драйвер автоматически использовать декларацию курсора/выборки для обработки операторов `SELECT` и поддерживать кэш в 100 строк. |
| BoolsAsChar | Управляет отображением логических типов.<br>Если установлено значение «Yes», логические типы отображаются на `SQL_CHAR`; в противном случае они отображаются на `SQL_BIT`. |
| SSLmode | Указывает режим SSL для соединения. |
| ConnSettings | Дополнительные настройки, отправляемые на бэкенд при соединении. |

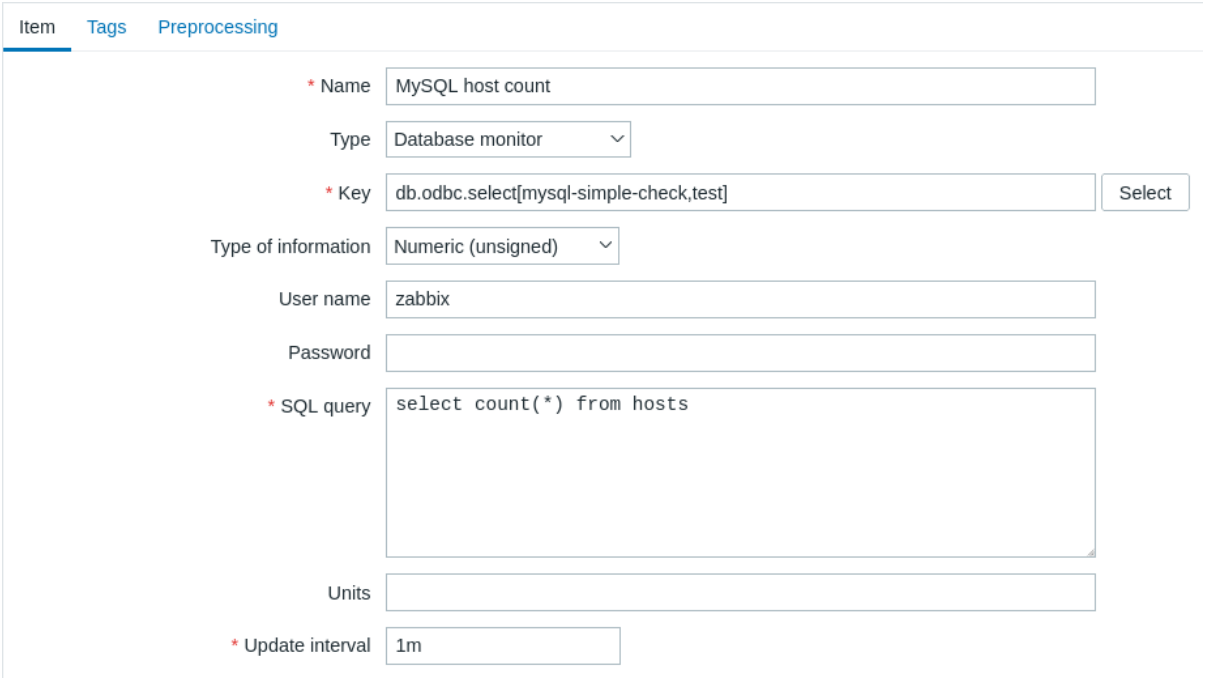
Тестирование соединения ODBC

Чтобы проверить, работает ли соединение ODBC успешно, можно использовать утилиту `isql` (входит в пакет `unixODBC`):

```
isql test
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
|                                       |
+---------------------------------------+
```

Настройка элемента данных в веб-интерфейсе Zabbix

Настройте элемент данных для **мониторинга базы данных**.




Все обязательные поля ввода отмечены красной звездочкой.

Для элементов данных мониторинга базы данных необходимо указать:

| | |
|---|---|
| *Type* | Выберите здесь "Database monitor". |
| *Key* | Введите один из поддерживаемых ключей элемента данных:<br>**db.odbc.select[]** - этот элемент данных возвращает одно значение (первый столбец первой строки результата SQL-запроса);<br>**db.odbc.get[]** - этот элемент данных возвращает несколько строк/столбцов в формате JSON;<br>**db.odbc.discovery[]** - этот элемент данных возвращает данные низкоуровневого обнаружения. |
| *User name* | Введите имя пользователя базы данных (до 255 символов).<br>Этот параметр необязателен, если имя пользователя базы данных указано в файле `odbc.ini`.<br>Если используется строка подключения и поле *User name* не пустое, то оно добавляется к строке подключения как `UID=<user>`. |
| *Password* | Введите пароль пользователя базы данных (до 255 символов).<br>Этот параметр необязателен, если пароль указан в файле `odbc.ini`.<br>Если используется строка подключения и поле *Password* не пустое, то оно добавляется к строке подключения как `PWD=<password>`.<br>В этом поле поддерживаются специальные символы.<br>Пароль будет добавлен к строке подключения после имени пользователя, например `UID=<username>;PWD=P?;)*word`.<br>Чтобы проверить итоговую строку, можно выполнить следующую команду:<br>`isql -v -k 'Driver=libmaodbc.so;Database=zabbix;UID=zabbix;PWD=P?;)*word'` |

| | |
|---|---|
| *SQL query* | Введите SQL-запрос.<br>Обратите внимание, что для `db.odbc.select[]` запрос должен возвращать только одно значение. |
| *Type of information* | Выберите здесь тип информации, который будет возвращен запросом.<br>Если тип информации выбран неверно, элемент данных станет неподдерживаемым. |

**Важные замечания**

- Элементы данных монитора баз данных станут неподдерживаемыми, если процессы *odbc poller* не запускаются в конфигурации сервера или прокси. Чтобы включить ODBC поллеры, задайте параметр `StartODBCPollers` в файле конфигурации Zabbix сервера или, если проверки выполняются на стороне прокси, в файле конфигурации Zabbix прокси.
- Значение параметра *Timeout* из диалога настроек элемента данных используется как время ожидания выполнения входа в ODBC и время ожидания выполнения запроса. Обратите внимание, что эти настройки времени ожидания могут быть проигнорированы, если установленный драйвер ODBC их не поддерживает.
- SQL команда должна вернуть набор данных, как и любой запрос с использованием оператора `select`. Синтаксис запроса будет зависеть от РСУБД, которая будет его обрабатывать. Синтаксис запроса к хранимой процедуре должен начинаться с ключевого слова `call`.

Подробности ключей элементов данных

Параметры без угловых скобок обязательны. Параметры, обозначенные угловыми скобками **< >**, опциональны.

db.odbc.select[<уникальное короткое описание>,<dsn>,<строка подключения>]

<br> Возвращает одно значение, то есть первый столбец первой строки результата SQL запроса.<br> Возвращаемое значение: Зависит от SQL запроса.

Параметры:

- **уникальное короткое описание** — уникальное короткое описание, чтобы идентифицировать элемент данных (для использования в триггерах и т. д.);
- **dsn** — имя источника данных (как указано в `odbc.ini`);
- **строка подключения** — строка подключения (может содержать специфичные для драйверов аргументы).

Комментарии:

- Хотя `dsn` и — опциональные параметры, по крайней мере один из них должен присутствовать; если заданы оба, то `dsn` будет игнорироваться.
- Если запрос возвращает более одного столбца, будет прочитан только первый столбец. Если запрос возвращает более одной строки, будет прочитана только первая строка.

db.odbc.get[<уникальное короткое описание>,<dsn>,<строка подключения>]

<br> Преобразует результат SQL-запроса в массив JSON.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **уникальное короткое описание** — уникальное короткое описание, чтобы идентифицировать элемент данных (для использования в триггерах и т. д.);
- **dsn** — имя источника данных (как указано в `odbc.ini`);
- **строка подключения** — строка подключения (может содержать специфичные для драйверов аргументы).

Комментарии:

- Хотя `dsn` и — опциональные параметры, по крайней мере один из них должен присутствовать; если заданы оба, то `dsn` будет игнорироваться.
- Может быть возвращено несколько строк/столбцов в формате JSON. Этот элемент данных можно использовать в качестве основного элемента данных, который собирает все данные за один системный вызов, а предварительную обработку JSONPath можно использовать в зависимых элементах данных для извлечения отдельных значений. Для получения более подробных сведений смотрите пример возвращаемого формата, использованного в низкоуровневом обнаружении.

Пример:

```
#                    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[<уникальное короткое описание>,<dsn>,<строка подключения>]

<br> Преобразует результат SQL-запроса в массив JSON, используемый для низкоуровневого обнаружения. Имена столбцов из результата запроса преобразуются в имена макросов низкоуровневого обнаружения, спаренных со значениями обнаруженных полей. Эти макросы можно использовать при создании прототипов элементов данных, триггеров и т. д.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **уникальное короткое описание** — уникальное короткое описание, чтобы идентифицировать элемент данных (для использования в триггерах и т. д.);
- **dsn** — имя источника данных (как указано в `odbc.ini`);
- **строка подключения** — строка подключения (может содержать специфичные для драйверов аргументы).

Комментарии:

- Хотя `dsn` и — опциональные параметры, по крайней мере один из них должен присутствовать; если заданы оба, то `dsn` будет игнорироваться.

Сообщения об ошибках

Сообщения об ошибках ODBC структурированы по полям для предоставления подробной информации. Например, сообщение об ошибке может выглядеть следующим образом:

```
Cannot execute ODBC query: [SQL_ERROR]:[42601][7][ERROR: syntax error at or near ";"; Error while executin
```

- «`Cannot execute ODBC query`» — сообщение Zabbix
- «`[SQL_ERROR]`» — код, возвращаемый ODBC
- «`[42601]`» — SQLState
- «`[7]`» — исходный код ошибки
- «`[ERROR: syntax error at or near ";"; Error while executing the query]`» — исходное сообщение об ошибке

Обратите внимание, что длина сообщения об ошибке ограничена 2048 байтами, поэтому сообщение может быть усечено. Если имеется более одной диагностической записи ODBC, Zabbix пытается их скомпоновать (разделяя символом `|`), насколько позволяет ограничение по длине.

### 9 HTTP агент

Обзор

Этот тип элемента данных позволяет получать данные с использованием протокола HTTP/HTTPS. Трапы также возможны с использованием утилиты Zabbix sender или протокола Zabbix sender (для отсылки данных на Zabbix сервер или прокси), либо с использованием метода API `history.push` (для отсылки данных на Zabbix сервер).

Проверка HTTP элемента данных выполняется на стороне Zabbix сервера. Однако, если узлы сети наблюдаются через Zabbix прокси, проверки HTTP элементов данных выполняются этими прокси.

Проверки HTTP элементов данных не требуют наличия какого-либо агента, запущенного на наблюдаемом узле сети.

HTTP агент поддерживает как HTTP, так и HTTPS. Zabbix будет опционально следовать перенаправлениям (смотрите ниже опцию *Следовать перенаправлениям*). Максимальное количество перенаправлений вшито в код и равно 10 (используется cURL опция CURLOPT_MAXREDIRS).

> **Attention:**
> Zabbix сервер/прокси должны быть изначально сконфигурированы с поддержкой cURL (libcurl).

Проверки HTTP выполняются асинхронно — не требуется получать ответ на один запрос перед началом других проверок. Разрешение DNS также выполняется асинхронно.

Максимальное количество параллельных асинхронных проверок составляет 1000 (определяется параметром MaxConcurrentChecksPerPoller).

Количество асинхронных поллеров HTTP агента определяется параметром StartHTTPAgentPollers.

Функция постоянных соединений (persistent connections) cURL была добавлена в проверки HTTP-агента, начиная с версии Zabbix 7.0.

Конфигурация

Чтобы настроить элемент данных HTTP:

1. Перейдите в *Сбор данных* > *Узлы сети*.

2. Нажмите *Элементы данных* в строке узла сети.
3. Нажмите *Создать элемент данных*.
4. Введите параметры элемента данных в форме.

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 | Action |
|---|---|---|---|
| Flexible \| Scheduling | 50s | 1-7,00:00-24:00 | Remove |

Add

* Timeout: Global | Override 3 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: ☐

Populates host inventory field: -None-

Description:

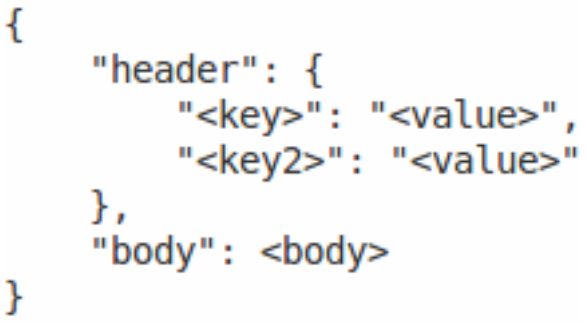
Enabled: ☑

Add | Test | Cancel

Все обязательные поля ввода отмечены красной звездочкой.

Поля, для которых требуется специальная информация для элементов данных HTTP:

| Parameter | Description |
|---|---|
| *Type* | Выберите здесь **HTTP agent**. |
| *Key* | Введите уникальный ключ элемента данных. |
| *URL* | URL для подключения и получения данных. Например:<br>https://www.example.com<br>http://www.example.com/download<br>Доменные имена можно указывать в символах Unicode. При выполнении HTTP-проверки они автоматически преобразуются в punycode и ASCII.<br>Кнопка *Parse* может использоваться для отделения необязательных полей запроса (например, ?name=Admin&password=mypassword) от URL, при этом атрибуты и значения перемещаются в *Query fields* для автоматического URL-кодирования.<br>Ограничено 2048 символами.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_URL. |
| *Query fields* | Переменные для URL (см. выше).<br>Указываются в виде пар атрибутов и значений.<br>Значения автоматически URL-кодируются. Значения из макросов сначала подставляются, а затем автоматически URL-кодируются.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_URL. |
| *Request type* | Выберите тип метода запроса: *GET*, *POST*, *PUT* или *HEAD* |
| *Request body type* | Выберите тип тела запроса:<br>**Raw data** - произвольное тело HTTP-запроса, макросы подставляются, но кодирование не выполняется<br>**JSON data** - тело HTTP-запроса в формате JSON. Макросы можно использовать как строку, число, true и false; макросы, используемые как строки, должны быть заключены в двойные кавычки. Значения из макросов сначала подставляются, а затем автоматически экранируются. Если в заголовках не указан ”Content-Type”, по умолчанию будет использоваться ”Content-Type: application/json”<br>**XML data** - тело HTTP-запроса в формате XML. Макросы можно использовать как текстовый узел, атрибут или секцию CDATA. Значения из макросов сначала подставляются, а затем автоматически экранируются в текстовом узле и атрибуте. Если в заголовках не указан ”Content-Type”, по умолчанию будет использоваться ”Content-Type: application/xml”<br>*Note* что выбор *XML data* требует libxml2. |
| *Request body* | Введите тело запроса.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения. |
| *Headers* | Пользовательские HTTP-заголовки, которые будут отправлены при выполнении запроса.<br>Указываются в виде пар атрибутов и значений.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_HTTPHEADER. |
| *Required status codes* | Список ожидаемых HTTP-кодов состояния. Если Zabbix получает код, которого нет в списке, элемент данных станет неподдерживаемым. Если поле пустое, проверка не выполняется.<br>Например: 200,201,210-299<br>Поддерживаемые макросы в списке: пользовательские макросы, макросы низкоуровневого обнаружения.<br>Использует опцию cURL CURLINFO_RESPONSE_CODE. |
| *Follow redirects* | Установите флажок, чтобы следовать HTTP-перенаправлениям.<br>Это задает опцию cURL CURLOPT_FOLLOWLOCATION. |

| Parameter | Description |
|---|---|
| *Retrieve mode* | Выберите часть ответа, которую нужно получить:<br>**Body** - только тело<br>**Headers** - только заголовки<br>**Body and headers** - тело и заголовки |
| *Convert to JSON* | Заголовки сохраняются как пары атрибутов и значений под ключом ”header”, если для *Retrieve mode* не задано значение *Body*.<br>Если встречается ’Content-Type: application/json’, тело сохраняется как объект, в противном случае оно хранится как строка, например:<br>  <br> |
| *HTTP proxy* | Можно указать HTTP-прокси для использования в формате `[protocol://][username[:password]@]proxy.example.com[:port]`.<br>Необязательный префикс `protocol://` можно использовать для указания альтернативных протоколов прокси (например, https, socks4, socks5; см. документацию; поддержка префикса протокола была добавлена в cURL 7.21.7). Если протокол не указан, прокси будет считаться HTTP-прокси. Если указан неверный протокол, соединение завершится ошибкой, и элемент данных станет неподдерживаемым.<br>По умолчанию будет использоваться порт 1080.<br>Если значение указано, прокси переопределит связанные с прокси переменные окружения, такие как http_proxy, HTTPS_PROXY. Если не указано, прокси не будет переопределять связанные с прокси переменные окружения. Введенное значение передается ”как есть”, без проверки корректности.<br>*Note* что с HTTP-прокси поддерживается только простая аутентификация.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_PROXY. |
| *HTTP authentication* | Выберите вариант аутентификации:<br>**None** - аутентификация не используется;<br>**Basic** - используется базовая аутентификация;<br>**NTLM** - используется аутентификация NTLM (Windows NT LAN Manager);<br>**Kerberos** - используется аутентификация Kerberos (см. также: Настройка Kerberos в Zabbix);<br>**Digest** - используется аутентификация Digest.<br>Это задает опцию cURL CURLOPT_HTTPAUTH. |
| *User name* | Введите имя пользователя (до 255 символов).<br>Это поле доступно, если для *HTTP authentication* выбрано Basic, NTLM, Kerberos или Digest.<br>Поддерживаются пользовательские макросы и макросы низкоуровневого обнаружения. |
| *Password* | Введите пароль пользователя (до 255 символов).<br>Это поле доступно, если для *HTTP authentication* выбрано Basic, NTLM, Kerberos или Digest.<br>Поддерживаются пользовательские макросы и макросы низкоуровневого обнаружения. |
| *SSL verify peer* | Установите флажок, чтобы проверить SSL-сертификат веб-сервера. Сертификат сервера будет автоматически взят из системного расположения центра сертификации (CA). Вы можете переопределить расположение файлов CA с помощью параметра конфигурации Zabbix server или proxy SSLCALocation.<br>Это задает опцию cURL CURLOPT_SSL_VERIFYPEER. |
| *SSL verify host* | Установите флажок, чтобы проверить соответствие полю Common Name или полю Subject Alternate Name сертификата веб-сервера.<br>Это задает опцию cURL CURLOPT_SSL_VERIFYHOST. |

| Parameter | Description |
|---|---|
| *SSL certificate file* | Имя файла SSL-сертификата, используемого для аутентификации клиента. Файл сертификата должен быть в формате PEM[1]. Если файл сертификата также содержит закрытый ключ, оставьте поле SSL key file пустым. Если ключ зашифрован, укажите пароль в поле SSL key password. Каталог, содержащий этот файл, задается параметром конфигурации Zabbix server или proxy SSLCertLocation.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_SSLCERT. |
| *SSL key file* | Имя файла закрытого SSL-ключа, используемого для аутентификации клиента. Файл закрытого ключа должен быть в формате PEM[1]. Каталог, содержащий этот файл, задается параметром конфигурации Zabbix server или proxy SSLKeyLocation.<br>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_SSLKEY. |
| *SSL key password* | Пароль файла закрытого SSL-ключа.<br>Поддерживаемые макросы: пользовательские макросы, макросы низкоуровневого обнаружения.<br>Это задает опцию cURL CURLOPT_KEYPASSWD. |
| *Timeout* | Zabbix не будет тратить на обработку URL больше заданного времени (1-600 секунд). Фактически этот параметр определяет максимальное время для подключения к URL и максимальное время для выполнения HTTP-запроса. Поэтому Zabbix не будет тратить на одну проверку больше, чем 2 x *Timeout* секунд.<br>Это задает опцию cURL CURLOPT_TIMEOUT.<br>Дополнительные сведения о параметре *Timeout* см. в разделе общие атрибуты элемента данных. |
| *Enable trapping* | Если этот флажок установлен, элемент данных также будет работать как элемент данных типа trapper и будет принимать данные, отправленные на сервер или прокси Zabbix с помощью утилиты Zabbix sender или протокола Zabbix sender protocol, либо будет принимать данные, отправленные на сервер Zabbix с помощью метода API `history.push`.<br>Дополнительные сведения об отправке данных см. в разделе: Элементы данных trapper. |
| *Allowed hosts* | Отображается только если установлен флажок *Enable trapping*.<br>Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен.<br>Если указано, входящие соединения будут приниматься только от перечисленных здесь узлов.<br>Если включена поддержка IPv6, то '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' считаются эквивалентными, а '::/0' разрешает любой IPv4- или IPv6-адрес.<br>'0.0.0.0/0' можно использовать, чтобы разрешить любой IPv4-адрес.<br>Обратите внимание, что "IPv4-compatible IPv6 addresses" (префикс 0000::/96) поддерживаются, но считаются устаревшими согласно RFC4291.<br>Пример: 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}<br>В этом поле разрешены пробелы и пользовательские макросы.<br>В этом поле разрешены макросы узла сети: {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN}. |

**Note:**
Если поле *HTTP proxy* оставлено пустым, другой способ использовать HTTP-прокси — задать связанные с прокси переменные окружения.
Для HTTP - задайте переменную окружения `http_proxy` для пользователя сервера Zabbix. Например:<br> `http_proxy=http://proxy_ip:proxy_port`.
Для HTTPS - задайте переменную окружения `HTTPS_PROXY`. Например:<br> `HTTPS_PROXY=http://proxy_ip:proxy_port`.
Дополнительные сведения можно получить, выполнив команду оболочки: *# man curl*.

::: noteimportant [1] Zabbix поддерживает файлы сертификатов и закрытых ключей только в формате PEM. Если у вас есть данные сертификата и закрытого ключа в файле формата PKCS #12 (обычно с расширением *.p12 или *.pfx), вы можете создать из него файл PEM с помощью следующих команд:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
```

```
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes  -out ssl-cert.key
```

Примеры

Пример 1

Отправка простых GET запросов на получение данных с таких сервисов, как Elasticsearch:

- Создайте GET элемент данных с URL: `localhost:9200/?pretty`
- Обратите внимание на ответ:

```
{
  "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"
}
```

- Теперь извлеките номер версии, используя шаг предварительной обработки JSONPath: `$.version.number`

Пример 2

Отправка простых POST запросов на получение данных с таких сервисов, как Elasticsearch:

- Создайте POST элемент данных с URL: `http://localhost:9200/str/values/_search?scroll=10s`
- Настройте следующее тело POST, чтобы получать загрузку процессора (среднее значение за 1 минуту по каждому ядру)

```
{
    "query": {
        "bool": {
            "must": [{
                "match": {
                    "itemid": 28275
                }
            }],
            "filter": [{
                "range": {
                    "clock": {
                        "gt": 1517565836,
                        "lte": 1517566137
                    }
                }
            }]
        }
    }
}
```

- Получено:

```
{
    "_scroll_id": "DnF1ZXJ5VGhlbkZldGNoBQAAAAAAAAAkFllRMlZBWS1UU1pxTmdEeGVwQjRBTFEAAAAAAAAAJRZZUTJWQVk
    "took": 18,
    "timed_out": false,
    "_shards": {
        "total": 5,
        "successful": 5,
        "skipped": 0,
        "failed": 0
    },
```

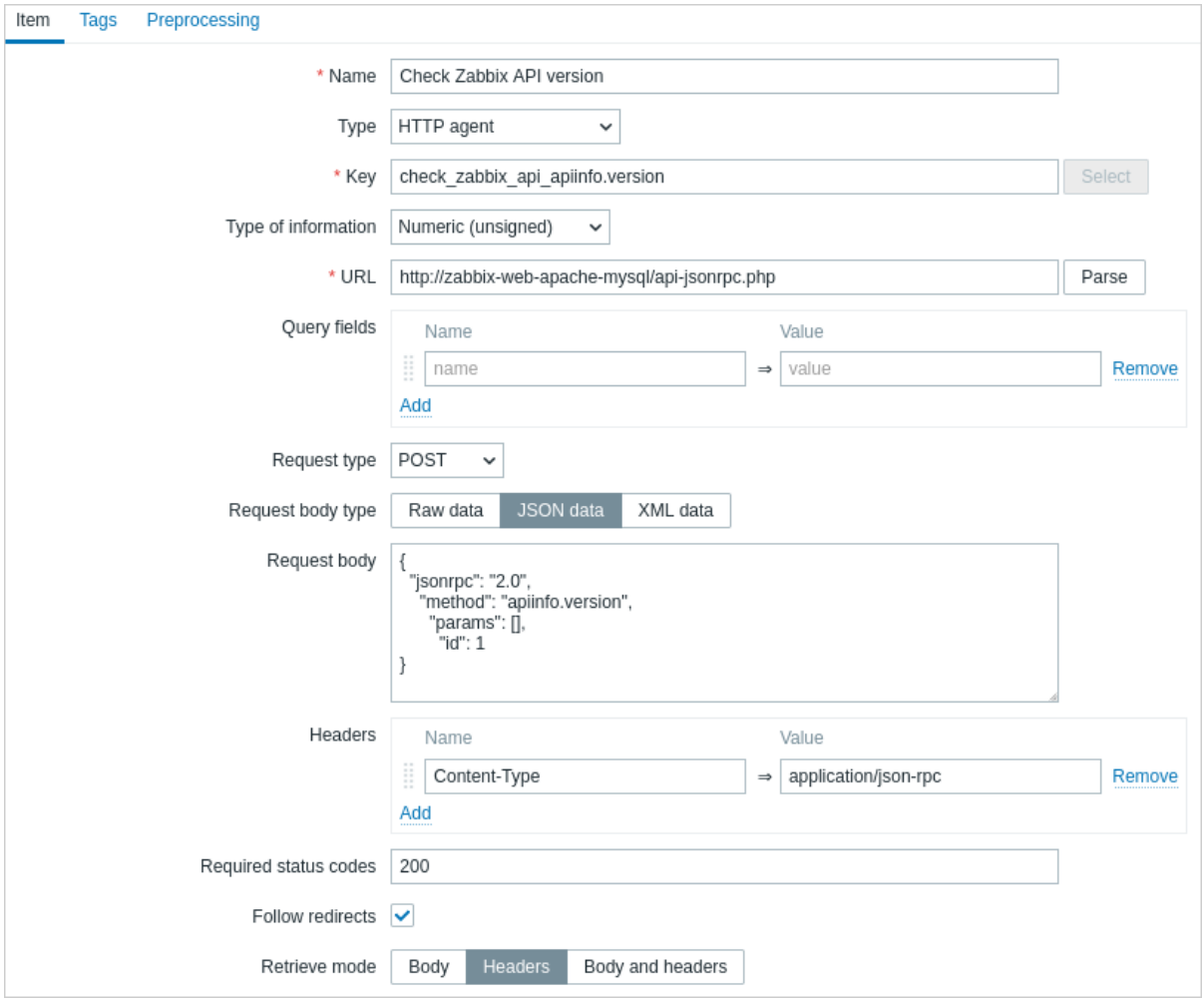
```
        "hits": {
            "total": 1,
            "max_score": 1.0,
            "hits": [{
                "_index": "dbl",
                "_type": "values",
                "_id": "dqX9VWEBV6sEKSMyk6sw",
                "_score": 1.0,
                "_source": {
                    "itemid": 28275,
                    "value": "0.138750",
                    "clock": 1517566136,
                    "ns": 25388713,
                    "ttl": 604800
                }
            }]
        }
    }
```


- Теперь используйте шаг предварительной обработки JSONPath, чтобы получить значение элемента данных: `$.hits.hits[0]._source.value`

Пример 3

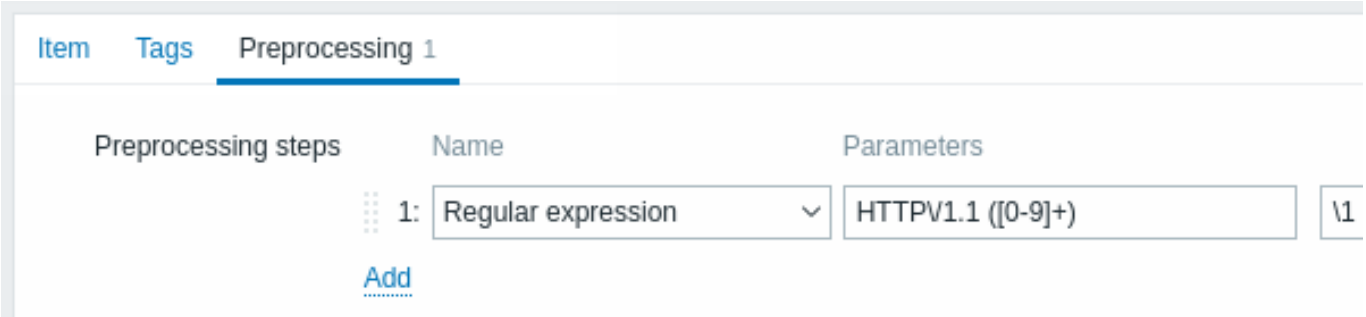
Проверка, доступен ли Zabbix API, используя apiinfo.version.

- Настройка элемента данных:

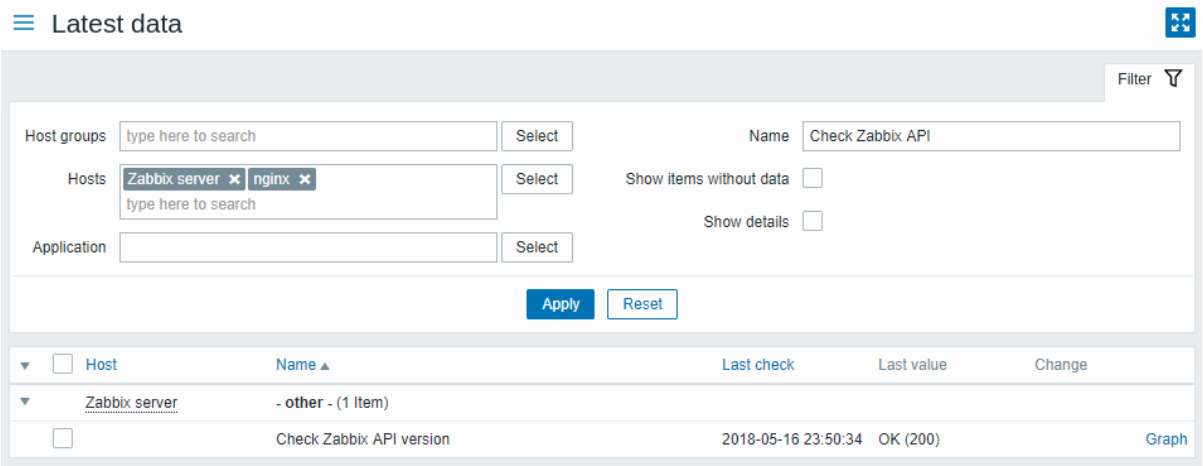



Обратите внимание на использование метода POST с данными JSON, установку заголовков запроса и запрос на возврат только заголовков:

- Предварительная обработка значений элемента данных при помощи регулярного выражения, чтобы получить HTTP код:



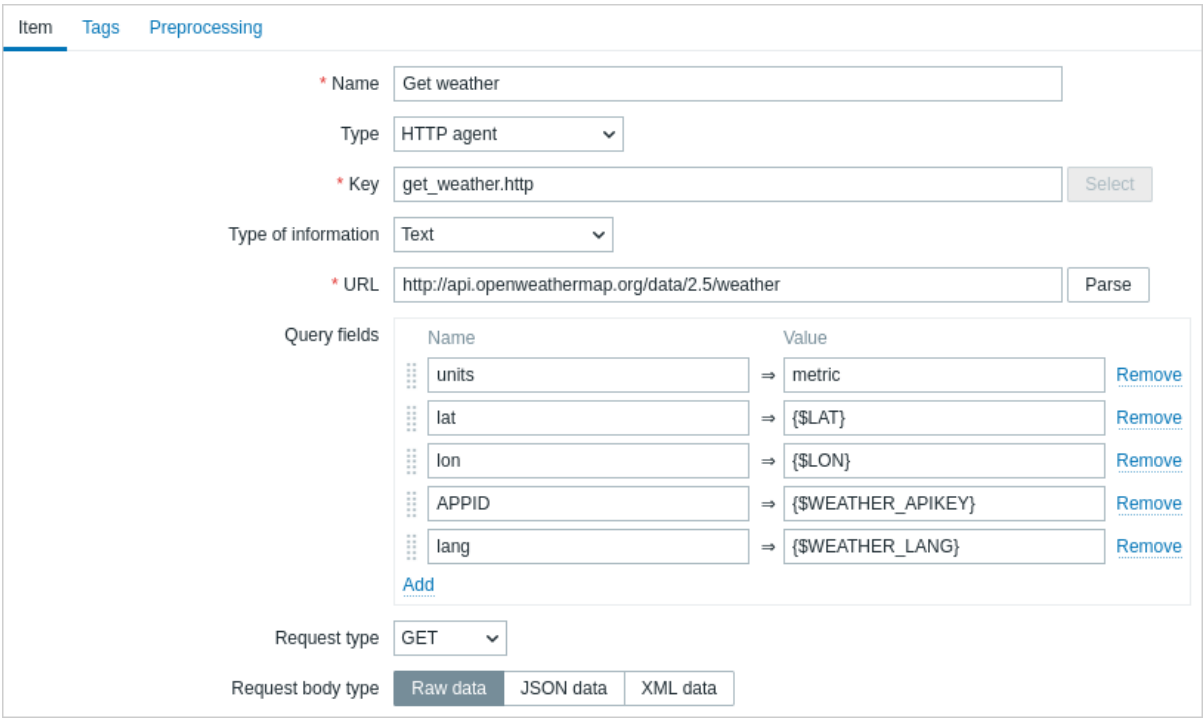

- Проверка результата в *Последние данные*:




Пример 4

Получение информации о погоде с публичного сервиса Openweathermap.

- Настройка основного элемента данных для массового сбора данных в одном JSON:




Обратите внимание на использование макросов в полях запроса. Обратитесь к Openweathermap API [en] для получения подробных сведений о том, чем их заполнять.

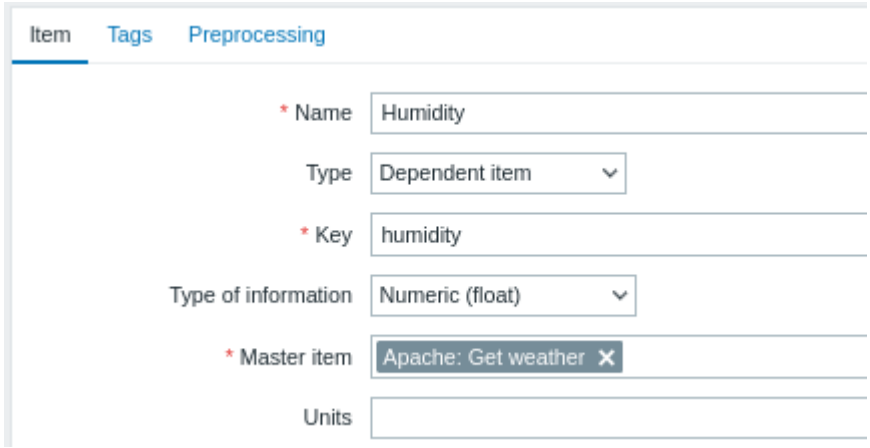
Пример JSON, который вернулся в ответе HTTP агенту:

```
{
    "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
    }
}
```

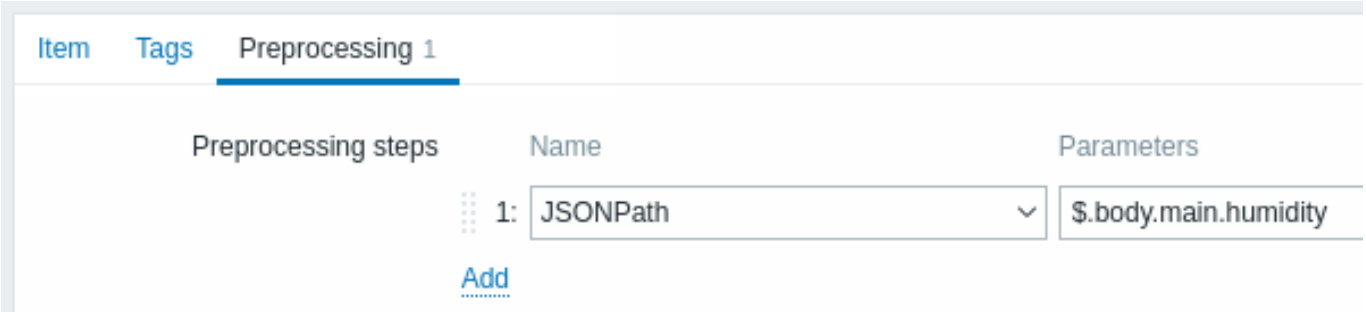

Следующая задача — настройка зависимых элементов данных, которые извлекут данные из JSON.

- Настройка примера зависимого элемента данных для влажности:

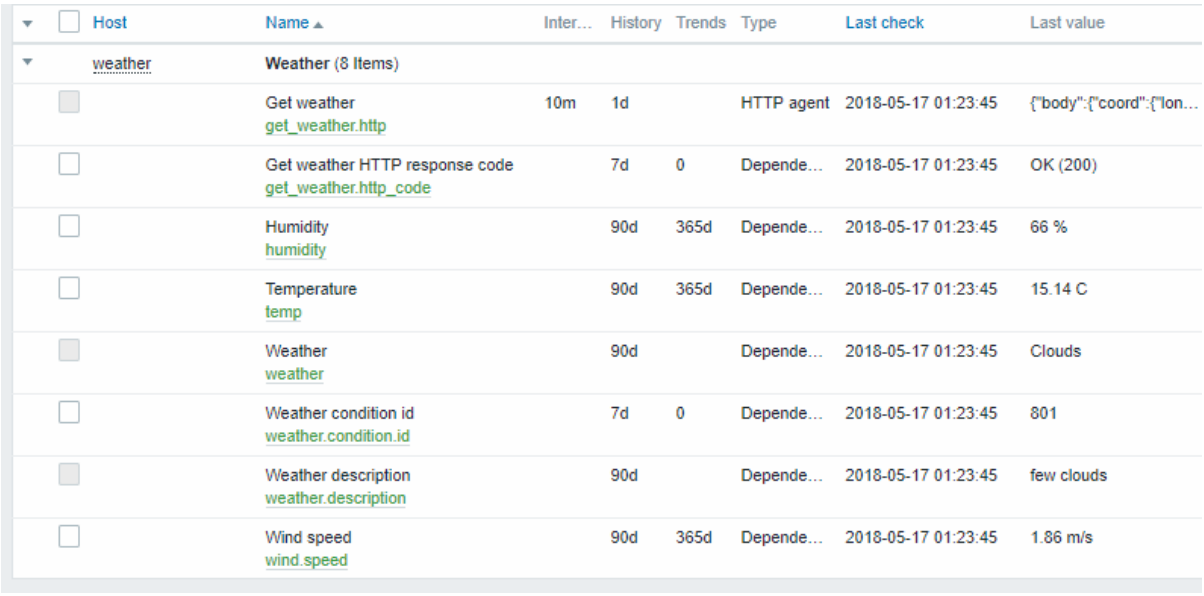



Другие метрики погоды, такие как «Температура», добавляются тем же способом.

- Пример предобработки значений зависимого элемента данных с использованием JSONPath:




- Проверка результата данных о погоде в *Последние данные*:

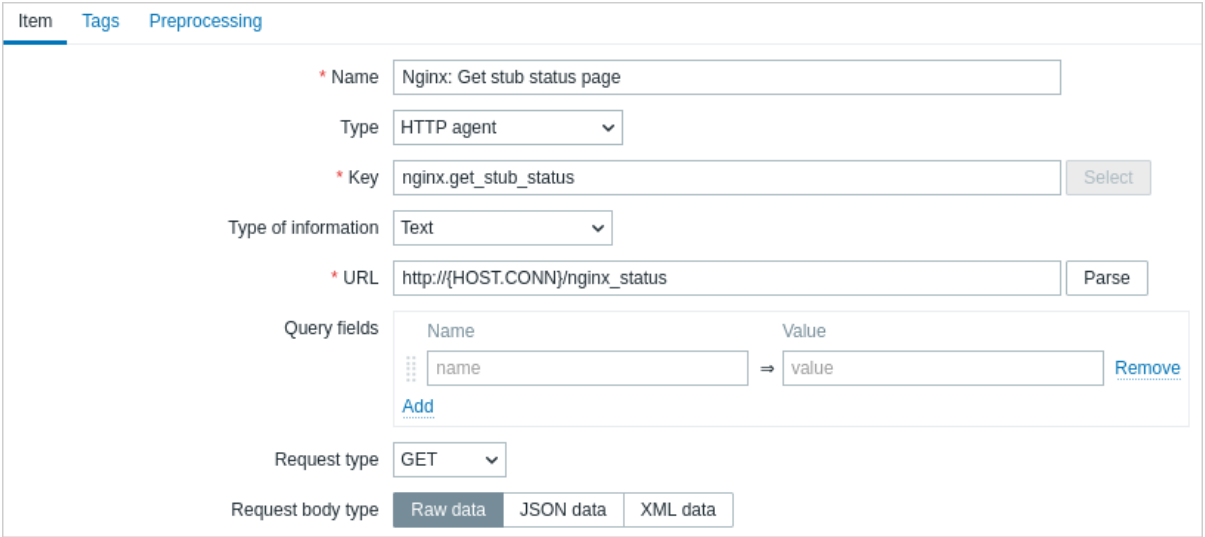


| Host | Name | Inter... | History | Trends | Type | Last check | Last value |
|---|---|---|---|---|---|---|---|
| weather | **Weather** (8 Items) | | | | | | |
| | Get weather<br>get_weather.http | 10m | 1d | | HTTP agent | 2018-05-17 01:23:45 | {"body":{"coord":{"lon... |
| | Get weather HTTP response code<br>get_weather.http_code | | 7d | 0 | Depende... | 2018-05-17 01:23:45 | OK (200) |
| | Humidity<br>humidity | | 90d | 365d | Depende... | 2018-05-17 01:23:45 | 66 % |
| | Temperature<br>temp | | 90d | 365d | Depende... | 2018-05-17 01:23:45 | 15.14 C |
| | Weather<br>weather | | 90d | | Depende... | 2018-05-17 01:23:45 | Clouds |
| | Weather condition id<br>weather.condition.id | | 7d | 0 | Depende... | 2018-05-17 01:23:45 | 801 |
| | Weather description<br>weather.description | | 90d | | Depende... | 2018-05-17 01:23:45 | few clouds |
| | Wind speed<br>wind.speed | | 90d | 365d | Depende... | 2018-05-17 01:23:45 | 1.86 m/s |

Пример 5

Подключение в странице состояния Nginx и получение его метрик за один запрос.

- Настройте Nginx, следуя официальному руководству.
- Настройте основной элемент данных для массового сбора данных:

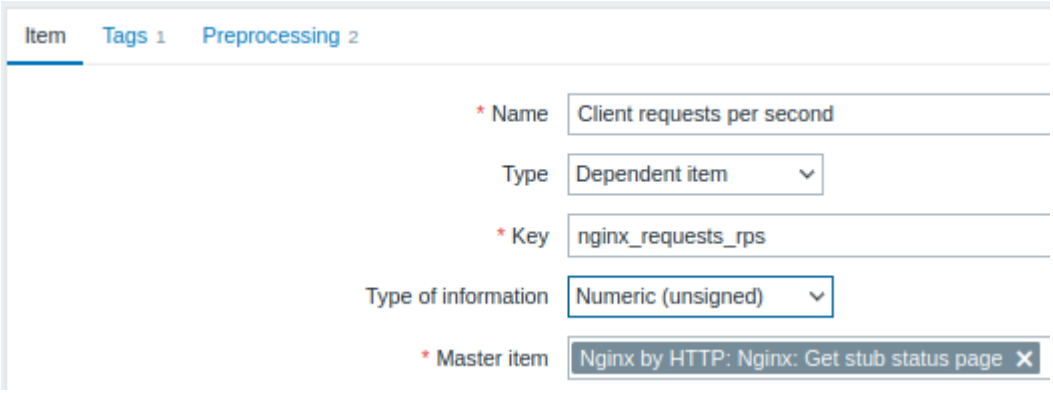



Пример вывода состояния Nginx stub:

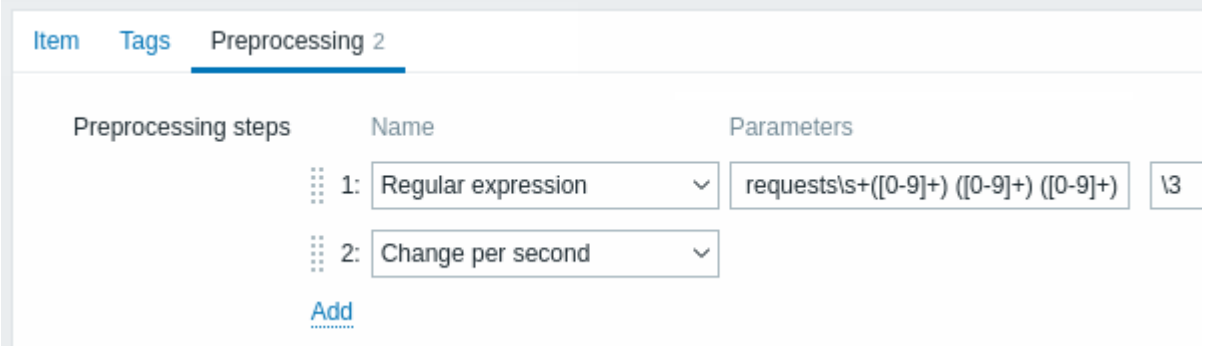
```
Active connections: 1 Active connections:
server accepts handled requests
 52 52 52
Reading: 0 Writing: 1 Waiting: 0
```

Следующая задача — настроить зависимые элементы данных, которые извлекают необходимые данные.

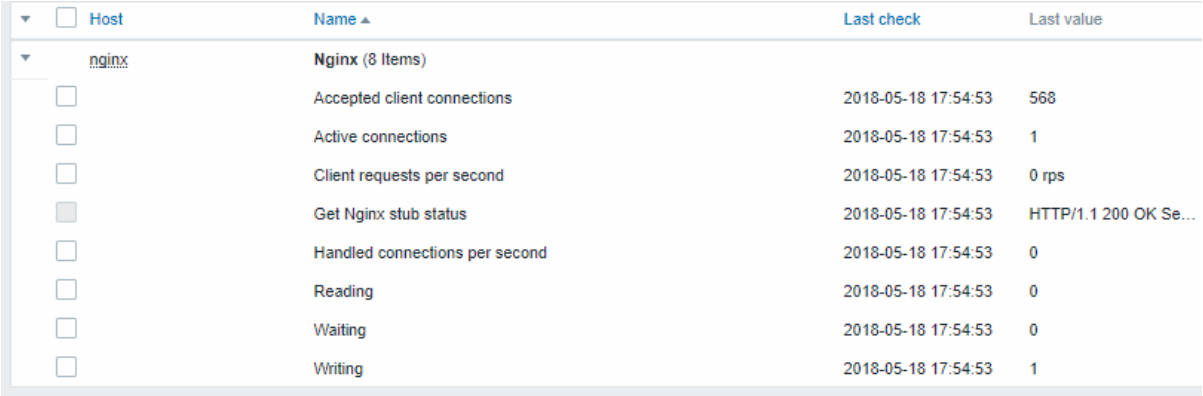
- Настройка примера зависимого элемента данных для запросов в секунду:




- Настройка предварительной обработки значений зависимого элемента данных при помощи регулярного выражения `server accepts handled requests\s+([0-9]+) ([0-9]+) ([0-9]+)`:




- Проверьте полученный результат от модуля stub в *Последние данные*:




1 Проверка Prometheus

Обзор

Zabbix может запрашивать метрики, предоставляемые в формате строк Prometheus.

Для начала сбора данных Prometheus требуется два шага:

- HTTP master item, указывающий на соответствующую конечную точку данных, например `https://<prometheus host>/metrics`
- зависимые элементы данных, использующие опцию предварительной обработки Prometheus для запроса необходимых данных из метрик, собранных master item

Существует две опции предварительной обработки данных Prometheus:

- *Prometheus pattern* - используется в обычных элементах данных для запроса данных Prometheus
- *Prometheus to JSON* - используется в обычных элементах данных и для низкоуровневого обнаружения. В этом случае запрашиваемые данные Prometheus возвращаются в формате JSON.

Пакетная обработка

Пакетная обработка поддерживается для зависимых элементов данных. Чтобы включить кэширование и индексирование, предварительная обработка *Prometheus pattern* должна быть **первым** шагом предварительной обработки. Когда

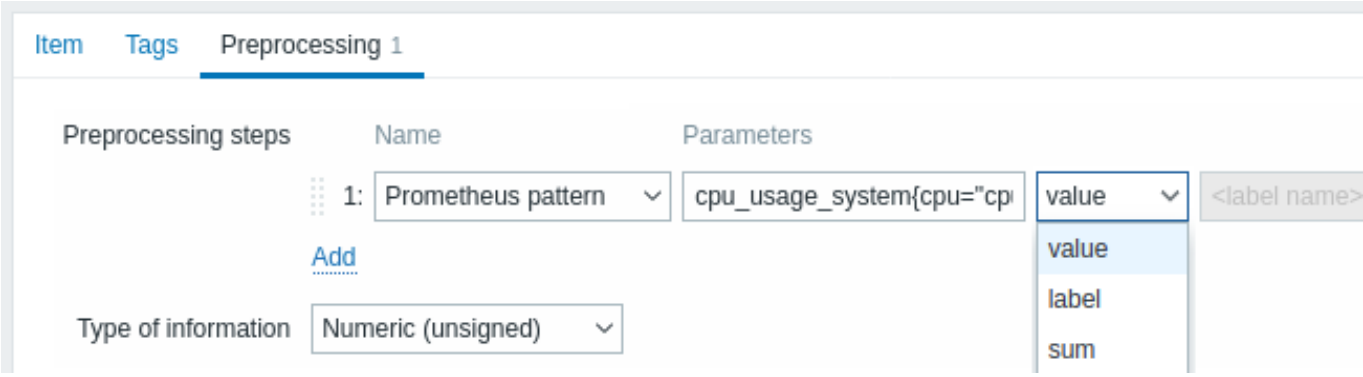
*Prometheus pattern* является первым шагом предварительной обработки, разобранные данные Prometheus кэшируются и индексируются по первому условию <label>==<value> в шаге предварительной обработки *Prometheus pattern*. Этот кэш повторно используется при обработке других зависимых элементов данных в этой партии. Для оптимальной производительности первым должен быть тот label, у которого больше всего различных значений.

Если перед первым шагом требуется выполнить другую предварительную обработку, ее следует перенести либо в мастер-элемент, либо в новый зависимый элемент данных, который будет использоваться как мастер-элемент для зависимых элементов данных.

Конфигурация

Если у вас настроен HTTP master-элемент данных, необходимо создать зависимый элемент данных, который использует шаг предварительной обработки Prometheus:

- Укажите общие параметры зависимого элемента данных в форме конфигурации
- Перейдите на вкладку Предварительная обработка
- Выберите вариант предварительной обработки Prometheus (*Prometheus pattern* или *Prometheus to JSON*)




Следующие параметры относятся только к варианту предварительной обработки *Prometheus pattern*:

| Parameter | Description | Examples |
|---|---|---|
| *Pattern* | Для определения требуемого шаблона данных можно использовать язык запросов, похожий на язык запросов Prometheus (см. сравнительную таблицу), например:<br><metric name> - выбор по имени метрики<br>{__name__="<metric name>"} - выбор по имени метрики<br>{__name__=~"<regex>"} - выбор по имени метрики, соответствующему регулярному выражению<br>{<label name>="<label value>",...} - выбор по имени метки<br>{<label name>=~"<regex>",...} - выбор по имени метки, соответствующему регулярному выражению<br>{__name__=~".*"}==<value> - выбор по значению метрики<br>Или комбинацию перечисленного выше:<br><metric name>{<label1 name>="<label1 value>",<label2 name>=~"<regex>",...}==<value><br><br>Значение метки может быть любой последовательностью символов UTF-8, но обратную косую черту, двойную кавычку и символ перевода строки необходимо экранировать как \\, \" и \n соответственно; остальные символы экранировать не нужно. | *wmi_os_physical_memory_free_bytes*<br>*cpu_usage_system{cpu="cpu-total"}*<br>*cpu_usage_system{cpu=~".*"}*<br>*cpu_usage_system{cpu="cpu-total",host=~".*"}*<br>*wmi_service_state{name="dhcp"}==1*<br>*wmi_os_timezone{timezone=~".*"}==1* |

| Parameter | Description | Examples |
|---|---|---|
| *Result processing* | Укажите, нужно ли вернуть значение, метку или применить соответствующую функцию (если шаблон совпадает с несколькими строками и результат нужно агрегировать):<br>**value** - вернуть значение метрики (ошибка, если совпало несколько строк)<br>**label** - вернуть значение метки, указанной в поле *Label* (ошибка, если совпало несколько метрик)<br>**sum** - вернуть сумму значений<br>**min** - вернуть минимальное значение<br>**max** - вернуть максимальное значение<br>**avg** - вернуть среднее значение<br>**count** - вернуть количество значений<br>Это поле доступно только для варианта *Prometheus pattern*. | См. также примеры использования параметров ниже. |
| *Output* | Определите имя метки (необязательно). В этом случае возвращается значение, соответствующее имени метки.<br>Это поле доступно только для варианта *Prometheus pattern*, если в поле *Result processing* выбрано 'Label'. | |

**Примеры использования параметров**

1. Наиболее распространенный вариант использования - вернуть **value**. Чтобы вернуть значение `/var/db` из:

   `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`

используйте следующие параметры:

- *Pattern* - `node_disk_usage_bytes{path="/var/db"}`
- *Result processing* - выберите 'value'

2. Вас также может интересовать **среднее** значение всех параметров `node_disk_usage_bytes`:

- *Pattern* - `node_disk_usage_bytes`
- *Result processing* - выберите 'avg'

3. Хотя Prometheus поддерживает только числовые данные, часто используют обходной прием, который позволяет также возвращать соответствующее текстовое описание. Это можно сделать с помощью фильтра и указания метки. Так, чтобы вернуть значение метки 'color' из

   `elasticsearch_cluster_health_status{cluster="elasticsearch",color="green"} 1<br> elasticsearch_cluster 0<br> elasticsearch_cluster_health_status{cluster="elasticsearch",color="yellow"} 0`

используйте следующие параметры:

- *Pattern* - `elasticsearch_cluster_health_status {cluster="elasticsearch"} == 1`
- *Result processing* - выберите 'label'
- *Label* - укажите 'color'

Фильтр (на основе числового значения '1') сопоставит нужную строку, а метка вернет описание состояния здоровья (в настоящее время 'green'; но потенциально также 'red' или 'yellow').

Prometheus в JSON

Данные из Prometheus можно использовать для низкоуровневого обнаружения. В этом случае требуются данные в формате JSON, и опция предварительной обработки *Prometheus to JSON* вернет именно их.

Подробнее см. Обнаружение с использованием данных Prometheus.

Сравнение языков запросов

В следующей таблице перечислены различия и сходства между PromQL и языком запросов предварительной обработки Prometheus в Zabbix.

| Селектор мгновенного вектора PromQL | | Предварительная обработка Prometheus в Zabbix |
|---|---|---|
| **Различия** | | |
| Целевой объект запроса | сервер Prometheus | Обычный текст в формате экспозиции Prometheus |
| Возвращает | Мгновенный вектор | Значение метрики или метки (шаблон Prometheus)<br>Массив метрик для одного значения в JSON (Prometheus to JSON) |
| Операторы сопоставления меток | **=**, **!=**, **=~**, **!~** | **=**, **!=**, **=~**, **!~** |
| Регулярное выражение, используемое при сопоставлении имени метки или метрики | RE2 | PCRE |
| Операторы сравнения | См. список | Для фильтрации значений поддерживается только **==** (равно) |
| **Сходства** | | |
| Выбор по имени метрики, равному строке | <имя метрики> или {**name**=”<имя метрики>”} | <имя метрики> или {**name**=”<имя метрики>”} |
| Выбор по имени метрики, соответствующему регулярному выражению | {**name**=~”<regex>”} | {**name**=~”<regex>”} |
| Выбор по значению <имя метки>, равному строке | {<имя метки>=”<значение метки>”,...} | {<имя метки>=”<значение метки>”,...} |
| Выбор по значению <имя метки>, соответствующему регулярному выражению | {<имя метки>=~”<regex>”,...} | {<имя метки>=~”<regex>”,...} |
| Выбор по значению, равному строке | {**name**=~”.*”} == <value> | {**name**=~”.*”} == <value> |

### 10 IPMI агент

Обзор

Вы можете наблюдать за состоянием и доступностью устройств Intelligent Platform Management Interface (IPMI) в Zabbix. Для выполнения проверок по IPMI Zabbix сервер должен быть изначально сконфигурирован с поддержкой IPMI.

IPMI — стандартизованный интерфейс для удаленного «lights-out» или «out-of-band» управления компьютерными системами. Он позволяет наблюдать за состоянием оборудования напрямую с так называемых карт управления «out-of-band», независимо от операционной системы или же от наличия питания на машине.

Zabbix IPMI мониторинг работает только с устройствами, имеющими поддержку IPMI (HP iLO, DELL DRAC, IBM RSA, Sun SSP и т.п.).

Процесс IPMI менеджер выполняет планирование проверок IPMI для IPMI поллеров. Узел сети всегда опрашивается только одним IPMI поллером за раз, уменьшая количество открытых подключений к контроллерам BMC. Благодаря этим изменениям можно безопасно увеличивать количество IPMI поллеров, не беспокоясь о перегрузке BMC контроллеров. Процесс IPMI менеджер автоматически запускается, если запускается хотя бы один IPMI поллер.

Смотрите также известные проблемы по IPMI проверкам.

Настройка

Настройка узла сети

Узел сети необходимо настроить для обработки проверок IPMI. Нужно добавить интерфейс IPMI, с соответствующими IP адресом и номером порта, а также задать параметры аутентификации IPMI.

Смотрите настройку узлов сети для получения более подробных сведений.

Настройка сервера

По умолчанию, Zabbix сервер не запускает IPMI поллеры, поэтому любые добавленные элементы данных IPMI не будут работать. Чтобы изменить это, откройте файл конфигурации (zabbix_server.conf) Zabbix сервера из-под root и найдите следующую строку:

```
# StartIPMIPollers=0
```

Раскомментируйте эту строку и задайте количество поллеров, скажем, равное 3 так, чтобы строка была такой:

```
StartIPMIPollers=3
```

Сохраните файл, после чего перезапустите zabbix_server.

Настройка элемента данных

Для настройки элемента данных на уровне узла сети:

- В поле *Тип* выберите «IPMI агент»
- Введите ключ элемента данных, уникальный в пределах узла сети (скажем, ipmi.fan.rpm)
- В поле *Интерфейс узла сети* выберите подходящий IPMI интерфейс (IP и порт). Обратите внимание, IPMI интерфейс должен уже существовать на узле сети.
- Укажите *IPMI датчик*, с которого забирать метрику (например, «FAN MOD 1A RPM» на Dell Poweredge). По умолчанию, необходимо указать ID датчика. Также имеется возможность использования префиксов до самого значения:
  - `id:` — чтобы указать ID датчика;
  - `name:` — чтобы указать полное имя датчика. Эта опция может быть полезна в ситуациях, когда датчики можно отличить, только указав полное имя.
- Выберите соответствующий тип информации («Числовой (с плавающей точкой)» в данном случае; для дискретных датчиков — «Числовой (целое положительное)»), единицы измерения (скорее всего, «rpm») и любые другие требуемые атрибуты элемента данных.

Поддерживаемые проверки

IPMI агент поддерживает встроенный элемент данных **ipmi.get**, который возвращает информацию, связанную с датчиком IPMI, и может использоваться для обнаружения датчиков IPMI.<br> Возвращаемое значение: *объект JSON*

Время ожидания и завершение сессии

Время ожидания IPMI сообщений и количества попыток определены в библиотеке OpenIPMI. В связи с текущим дизайном OpenIPMI, невозможно сделать эти значения настраиваемыми из Zabbix, ни на уровне интерфейса, ни на уровне элемента данных.

Время ожидания неактивности IPMI сессии для LAN равняется 60 +/-3 секунд. В настоящее время невозможно реализовать периодическую отправку команды Активации Сессии в OpenIPMI. Если проверки IPMI элементов данных от Zabbix к конкретному BMC не выполняются в течение времени, большего, чем время ожидания сессии, настроенное в BMC, то следующая проверка IPMI после истечения времени ожидания приведёт к ошибкам из-за превышения времени ожидания отдельного сообщения, повторных попыток или ошибке при получении. После этого открывается новая сессия и инициируется полное повторное сканирование BMC. Для управления новой сессией может быть открыт новый порт UDP. Неактивность определяется как отсутствие и исходящих запросов, и входящих ответов. Если вы хотите избежать лишнего сканирования BMC, рекомендуется установить интервал опроса элементов данных IPMI ниже времени ожидания неактивности IPMI сессии, настроенного в BMC.

Примечания о дискретных датчиках IPMI

Для поиска датчиков на узле сети запустите Zabbix сервер с включённым **DebugLevel=4**. Подождите пару минут и поищите записи об обнаруженных датчиках в журнале Zabbix сервера:

```
$ 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
```

Для расшифровки типов датчиков IPMI и их состояний доступна копия спецификации IPMI 2.0 [en] (пожалуйста, обратите внимание, что не планируется дальнейших обновлений [en] к спецификации IPMI).

Начнем с параметра «тип_чтения» (reading_type). Для расшифровки кода «reading_type» используйте раздел «Table 42-1, Event/Reading Type Code Ranges» из спецификации. Большинство датчиков из нашего примера имеют «reading_type:0x1», означающих «порог» датчика. «Table 42-3, Sensor Type Codes» показывает, что «type:0x1» — датчик температуры, «type:0x2» — датчик напряжения, «type:0x4» — датчик частоты вращения вентилятора системы охлаждения и так далее. Пороговые датчики иногда называют «аналоговыми» датчиками, так как они измеряют непрерывные параметры, такие как температуру, напряжение, частоту вращения в минуту.

Другой пример — датчик с «reading_type:0x3». «Table 42-1, Event/Reading Type Code Ranges» говорит, что коды типов чтения 02h-0Ch означают «Общий Дискретный» датчик. Дискретные датчики имеют до 15 возможных состояний (другими словами — до 15 значащих бит). К примеру, для датчика «CATERR» с «type:0x7» «Table 42-3, Sensor Type Codes» показывает, что этот тип обозначает «Процессор» и значение отдельных битов: 00h (наименьший значащий бит) — IERR (внутренняя ошибка процессора), 01h — Перегрев процессора и т.д.

В нашем примере есть несколько датчиков с «reading_type:0x6f». Для этих датчиков «Table 42-1, Event/Reading Type Code Ranges» советует использовать «Table 42-3, Sensor Type Codes» для расшифровки значений битов. Например, датчик «Power Unit Stat» имеет тип «type:0x9», который означает «Блок питания». Смещение 00h означает «Выключено/Обесточено». Другими словами, если младший значащий бит равен 1, то сервер выключен. Для проверки этого бита можно воспользоваться функцией `bitand` с маской «1». Выражение триггера для предупреждения о выключенном сервере может выглядеть следующим образом:

```
bitand(last(/www.example.com/Power Unit Stat,#1),1)=1
```

Заметки об именах дискретных датчиков в OpenIPMI-2.0.16, 2.0.17, 2.0.18 и 2.0.19

Имена дискретных датчиков в OpenIPMI-2.0.16, 2.0.17 и 2.0.18 зачастую имеют дополнительный символ «0» (или какую-то другую цифру или символ), добавленный в конце имени. Например, тогда как `ipmitool` и OpenIPMI-2.0.19 отображают имена датчиков как «`PhysicalSecurity`» или «`CATERR`», в OpenIPMI-2.0.16, 2.0.17 и 2.0.18 эти имена — «`PhysicalSecurity0`» или «`CATERR0`» соответственно.

При настройке элемента данных IPMI для Zabbix сервера, использующего OpenIPMI-2.0.16, 2.0.17 и 2.0.18, добавьте к их именам «0» в поле *IPMI датчик* для элементов данных IPMI агента. Когда ваш Zabbix сервер будет обновлён в новом Linux дистрибутиве, использующем OpenIPMI-2.0.19 (или более позднюю), элементы данных с такими IPMI дискретными датчиками перейдут в состояние «НЕ ПОДДЕРЖИВАЕТСЯ». Вам потребуется изменить их имена *IPMI датчик* (удалить «0» в конце) и подождать некоторое время, пока они станут «Активированными» снова.

Заметки об одновременной доступности пороговых и дискретных датчиков

Некоторые IPMI агенты предоставляют одновременно пороговые и дискретные датчики под одним именем. Предпочтение всегда отдаётся пороговому датчику.

Примечания о завершении соединений

Если IPMI проверки не выполняются (по любой из причин: все элементы данных IPMI деактивированы / не поддерживаются на узле сети, сам узел сети деактивирован/удалён, узел сети находится в режиме обслуживания и так далее), то соединение будет разорвано со стороны Zabbix сервера и прокси через 3 — 4 часа, в зависимости от времени запуска Zabbix сервера/прокси.

### 11 SSH агент

Обзор

Проверки SSH выполняются как мониторинг без агента. Для проверок SSH агент Zabbix не требуется.

Для выполнения проверок SSH сервер Zabbix должен быть изначально настроен с поддержкой SSH2 (libssh или libssh2). См. также: Требования.

> **Attention:**
> Начиная с RHEL 8 поддерживается только libssh. Для других дистрибутивов рекомендуется использовать libssh вместо libssh2.

Настройка

Аутентификация по паролю

Проверки SSH предоставляют два метода аутентификации: пара логин пользователя/пароль и на основе файла ключа.

Если вы не собираетесь использовать ключи, то (помимо привязки библиотеки libssh2/libssh к Zabbix при компиляции из исходных кодов) никакой дополнительной настройки не требуется.

Аутентификация по ключу

Чтобы для элементов данных SSH использовать аутентификацию по ключу, необходимо произвести некоторые изменения в конфигурации сервера.

Откройте файл конфигурации сервера Zabbix (*zabbix_server.conf*) из-под `root` и найдите следующую строку:

```
##### SSHKeyLocation=
```

Раскомментируйте её и укажите полный путь к папке, где размещены открытые и закрытые ключи (public and private keys):

```
SSHKeyLocation=/home/zabbix/.ssh
```

Затем сохраните файл и перезапустите сервер Zabbix.

Здесь */home/zabbix* — домашняя папка для учётной записи пользователя *zabbix*, а *.ssh* — папка внутри неё, куда по умолчанию с помощью команды ssh-keygen [en] будут сгенерированы открытые и закрытые ключи.

Обычно при установке сервера Zabbix из пакетов на разных дистрибутивах ОС создается аккаунт пользователя *zabbix* с домашней папкой в не очень стандартных местах, например, */var/lib/zabbix* (как для системных аккаунтов).

До начала создания ключей рассмотрите вариант перемещения домашней папки в более известное место (интуитивно ожидаемое). Этот вариант будет соответствовать параметру `SSHKeyLocation` конфигурации Zabbix сервера, упомянутому выше.

> **Note:**
> Эти шаги можно пропустить, если учётная запись *zabbix* добавлена вручную в соответствии с разделом установки, потому что в этом случае домашняя папка, скорее всего, уже расположена в */home/zabbix*.

Для изменения этой настройки у учётной записи пользователя *zabbix* все работающие процессы, которые его используют, потребуется остановить:

```
systemctl stop zabbix-agent
systemctl stop zabbix-server
```

Чтобы изменить местоположение домашней папки с попыткой переместить её (если папка существует), вы должны выполнить команду:

```
usermod -m -d /home/zabbix zabbix
```

Вполне возможно, что домашняя папка не существует в старом месте, поэтому в новом месте её необходимо создать. Безопасная попытка, чтобы сделать это:

```
test -d /home/zabbix || mkdir /home/zabbix
```

Чтобы быть уверенным, что всё безопасно, можно выполнить дополнительные команды для установки разрешений к домашней папке:

```
chown zabbix:zabbix /home/zabbix
chmod 700 /home/zabbix
```

Теперь можно снова запустить ранее остановленные процессы:

```
systemctl start zabbix-agent
systemctl start zabbix-server
```

Теперь шаги генерирования открытых и закрытых ключей можно выполнить следующими командами (для лучшей читабельности запросы команд закомментированы):

```
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):
<        >
##### Enter same passphrase again:
<        >
##### 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:**
Открытые и закрытые ключи (*id_rsa.pub* и *id_rsa* соответственно) будут сгенерированы по умолчанию в папке */home/zabbix/.ssh*, которая соответствует параметру конфигурации `SSHKeyLocation` Zabbix сервера.

**Attention:**
Утилитой ssh-keygen и сервером SSH могут поддерживаться ключи и других типов, нежели «rsa»; но они могут не поддерживаться используемой Zabbix библиотекой libssh2.

Диалог настройки командной строки

Этот шаг необходимо выполнить только один раз для каждого хоста, который будет наблюдаться с использованием SSH проверок.

При использовании следующей команды файл **открытого** ключа может быть установлен на удалённый хост *10.10.10.10*, для того чтобы потом можно было выполнять SSH проверки при помощи аккаунта *root* (для лучшей читабельности подсказки команды закомментированы):

```
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:
<               root>
#####                          "ssh 'root@10.10.10.10'",
#####       ,         ,                              .
```

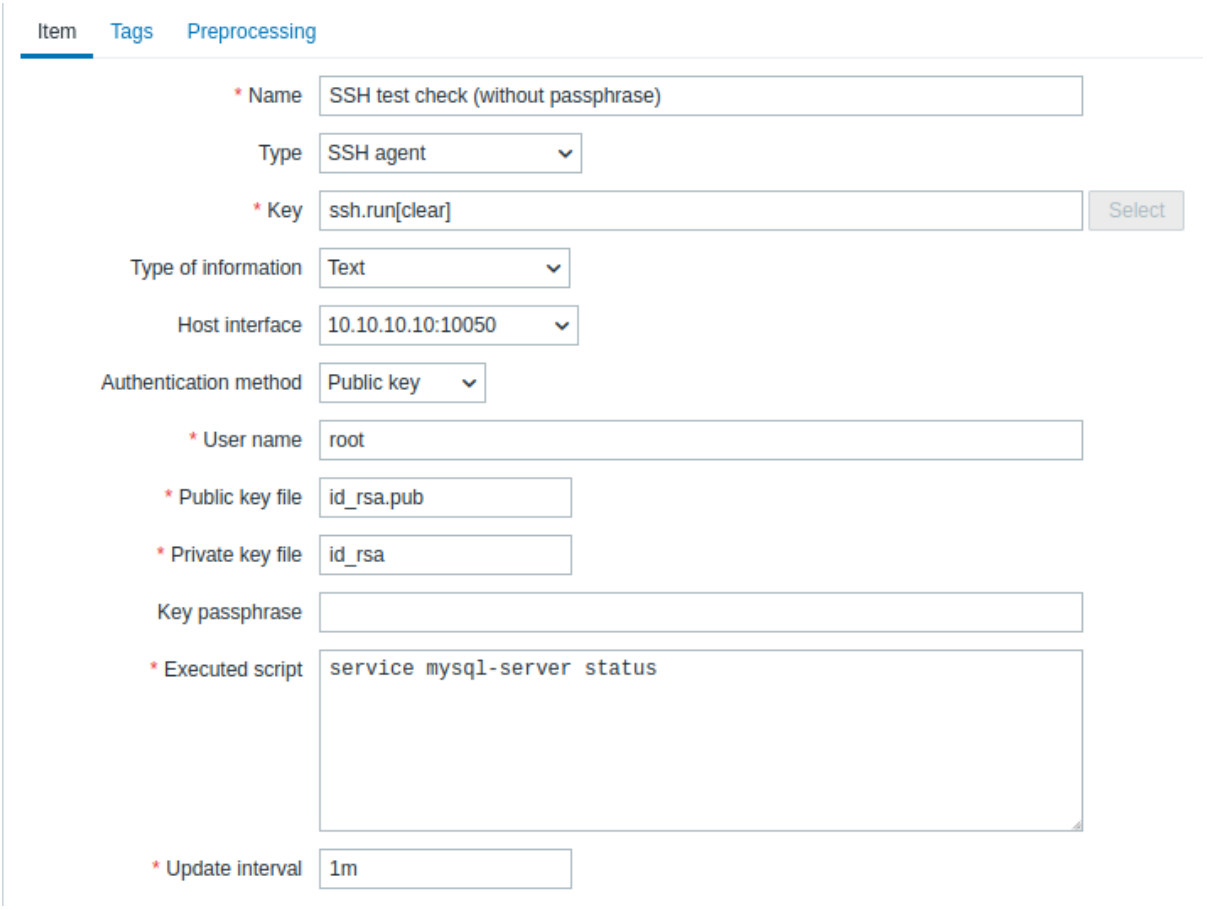
Теперь можно проверить вход по SSH с использованием закрытого ключа по умолчанию (*/home/zabbix/.ssh/id_rsa*) учётной записи пользователя *zabbix*:

```
sudo -u zabbix ssh root@10.10.10.10
```

Если вход успешен, то часть настройки в командной строке завершена и удалённую сессию по SSH можно закрыть.

Конфигурация элемента данных

Фактические команды, которые необходимо выполнить, должны быть указаны в поле *Executed script* в конфигурации элемента данных. Несколько команд можно выполнять одну за другой, размещая их на новой строке. В этом случае возвращаемые значения также будут форматироваться как многострочные.




Все обязательные поля ввода отмечены красной звездочкой.

Поля, для которых для элементов данных SSH требуется конкретная информация:

| Parameter | Description | Comments |
|---|---|---|
| *Type* | Выберите здесь **SSH agent**. | |

| Parameter | Description | Comments |
|---|---|---|
| *Key* | Уникальный ключ элемента данных (для каждого узла сети) в формате **ssh.run[unique short description,<ip>,<port>,<encoding>,<ssh options>,<subsystem>]** | **unique short description** обязателен и должен быть уникальным для каждого элемента данных SSH на каждом узле сети.<br><br>Порт по умолчанию — 22, а не порт, указанный в интерфейсе, к которому назначен этот элемент данных.<br><br>**ssh options** позволяют передавать дополнительные параметры SSH в формате *key1=value1;key2=value2,value3*. Несколько значений для одного ключа можно передавать через запятую (в этом случае параметр должен быть в кавычках); несколько ключей параметров можно передавать через точку с запятой.<br><br>Поддерживаются следующие ключи параметров: `KexAlgorithms`, `HostkeyAlgorithms`, `Ciphers`, `MACs`, `PubkeyAcceptedKeyTypes`. Поддержка ключа и значения зависит от библиотеки SSH (например, `PubkeyAcceptedKeyTypes` поддерживается только в libssh); если параметр не поддерживается, будет возвращена ошибка, и элемент данных станет неподдерживаемым.<br><br>Обратите внимание, что знак "+" для добавления настроек шифров и "!" для отключения определенных настроек шифров (как в GnuTLS и OpenSSL) не поддерживаются.<br><br>Примеры:<br>=><br>`ssh.run[KexAlgorithms,127.0.0.1,,,Ciphers=aes128`<br>=><br>`ssh.run[KexAlgorithms,,,,"KexAlgorithms=diffie-he`<br>=><br>`ssh.run[PubkeyAcceptedKeyTypes,127.0.0.1,,,Pubkey`<br><br>**subsystem** позволяет передать SSH-подсистему, ограничивая SSH-подключение конкретными операциями, разрешенными этой подсистемой (например, передачей файлов с помощью SFTP или управлением сетевым устройством с помощью NETCONF). Обратите внимание, что использование подсистемы может также потребовать применения определенного синтаксиса сценария в параметре *Executed script*.<br><br>Примеры:<br>=><br>`ssh.run[SFTPBackup,192.0.2.18,,,,sftp]`<br>=><br>`ssh.run[Cisco1234,192.0.2.18,,,,netconf]` |
| *Authentication method* | Один из вариантов: "Password" или "Public key". | |

| Parameter | Description | Comments |
|---|---|---|
| *User name* | Имя пользователя (до 255 символов) для аутентификации на удаленном узле сети.<br>Обязательно. | |
| *Public key file* | Имя файла открытого ключа, если *Authentication method* — "Public key".<br>Обязательно. | Пример: *id_rsa.pub* — имя файла открытого ключа по умолчанию, создаваемого командой ssh-keygen. |
| *Private key file* | Имя файла закрытого ключа, если *Authentication method* — "Public key".<br>Обязательно. | Пример: *id_rsa* — имя файла закрытого ключа по умолчанию. |
| *Password* or *Key passphrase* | Пароль (до 255 символов) для аутентификации или<br>парольная фраза **если** она использовалась для закрытого ключа. | Оставьте поле *Key passphrase* пустым, если парольная фраза не использовалась.<br>См. также известные проблемы, связанные с использованием парольной фразы. |
| *Executed script* | Выполняемые команды оболочки через удаленный сеанс SSH. | Возвращаемое значение выполняемых команд оболочки ограничено 16 МБ (включая завершающие пробелы, которые обрезаются); также действуют ограничения базы данных.<br><br>Обратите внимание, что библиотека libssh2 может обрезать исполняемые сценарии примерно до ~32 КБ.<br><br>Примеры:<br>`date +%s`<br>`systemctl status mysql-server`<br>`ps auxww \| grep httpd \| wc -l`<br><br>Пример (для подсистемы NETCONF):<br>`<rpc>`<br>`<get-software-information/>`<br>`</rpc>`<br>`]]>]]>`<br>`<rpc>`<br>`<close-session/>`<br>`</rpc>`<br>`]]>]]>` |

### 12 Telnet агент

Обзор

Telnet проверки выполняются без наличия какого-либо агента. Zabbix агент не требуется для проверок через Telnet.

Настраиваемые поля

Фактические команда(ы) для выполнения необходимо поместить в поле **Выполняемый скрипт** при настройке элемента данных.<br> Можно выполнять несколько команд одну за другой, размещая каждую на новой строке. В этом случае значение результата также будет отформатировано как многострочное.

Поддерживаемые символы, которыми может заканчиваться строка приглашения командной оболочки:

- $
- #
- >
- %

**Note:**
Строка telnet с запросом, которая заканчивается одним из этих символов, будет удалена из возвращаемого значения, но только для первой команды из списка команд, то есть, только в начале telnet сессии.

| Ключ | Описание |
|---|---|
| **telnet.run[<уникальное короткое описание>,<ip>,<порт>,<кодировка>]** | Выполнение команды на удалённом устройстве с использованием telnet соединения |

**Attention:**
Если telnet проверка возвращает значение с не-ASCII символами и не в UTF8 кодировке, то необходимо указать корректный параметр ключа *<кодировка>*. Смотрите страницу о кодировке возвращаемых значений для получения более подробных сведений.

### 13 JMX агент

Обзор

Мониторинг JMX можно использовать для мониторинга счётчиков JMX в Java приложениях.

Мониторинг JMX имеет в Zabbix встроенную поддержку в виде демона Zabbix, называемого «Zabbix Java gateway».

Когда Zabbix сервер хочет узнать значение конкретного счетчика JMX у узла сети, он опрашивает Zabbix **Java gateway**, который, в свою очередь, использует API управления JMX [en], чтобы удалённо опросить интересующее приложение.

Для получения более подробных сведений, включая настройку, смотрите раздел Zabbix Java gateway.

**Warning:**
Подключения между Java gateway и наблюдаемым JMX приложением не должны быть закрыты брандмауэром.

Включение удалённого JMX мониторинга для Java приложений

Приложению Java не требуется какое-либо дополнительно установленное программное обеспечение, но для поддержки удалённого мониторинга JMX приложение необходимо запустить с указанными ниже параметрами командной строки.

Как минимум, если вы просто хотите начать мониторинг простого приложения Java на локальном хосте без какой-либо защиты, запустите его со следующими параметрами:

```
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 /   /  /     .jar
```

С этими аргументами Java будет слушать входящие соединения JMX на порту 12345, только с локального хоста, без необходимости в аутентификации или SSL.

Если вы хотите разрешить подключения с другого интерфейса, укажите параметр -Djava.rmi.server.hostname равным IP адресу этого интерфейса.

Если вы хотите иметь более строгую проверку в плане безопасности, вам доступно множество других опций Java. Скажем, следующий пример запускает приложение с более мудрёным набором опций и открывает это приложение для более широкой сети, а не только для локального компьютера.

```
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=$  _    _    \
-Djavax.net.ssl.keyStorePassword=$  _   _ _    _    \
-Djavax.net.ssl.trustStore=$  _    _     \
-Djavax.net.ssl.trustStorePassword=$  _   _ _     _     \
```

```
-Dcom.sun.management.jmxremote.ssl.need.client.auth=true \
-jar /   /   /      .jar
```

Большинство (если не все) из этих настроек можно указать в `$JRE/lib/management/management.properties` (или там, где этот файл расположен на вашем компьютере).

Обратите внимание, что если вы хотите использовать SSL, вам необходимо изменить скрипт startup.sh, добавив в него опции Java gateway `-Djavax.net.ssl.*`, чтобы он знал, где найти хранилище ключей и доверенное хранилище.

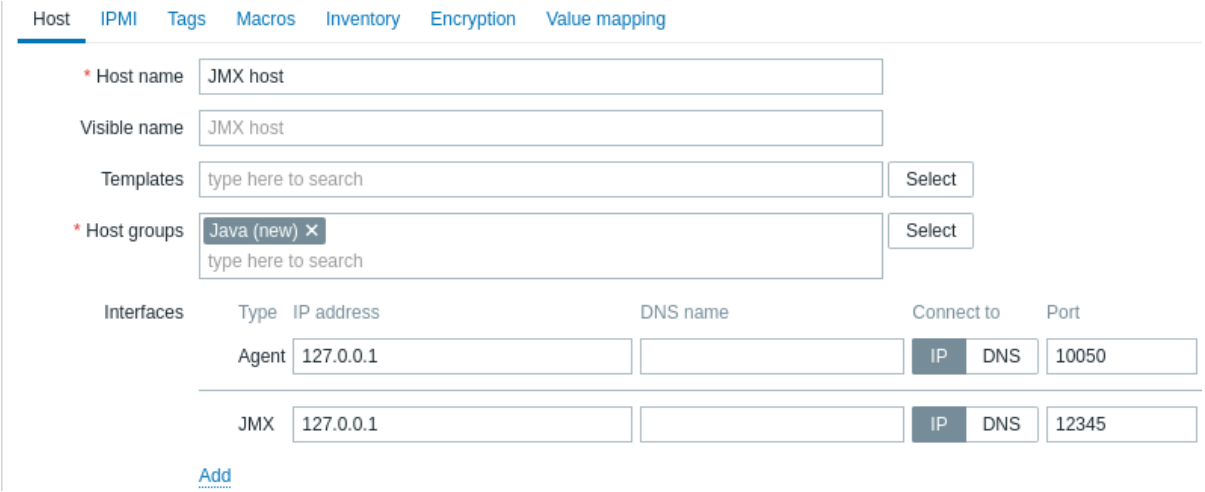
Смотрите Мониторинг и Управление с использованием JMX [en] для получения более подробной информации.

Настройка JMX интерфейсов и элементов данных в веб-интерфейсе Zabbix

Когда Java Gateway запущен, сервер знает, где его искать, и Java приложение запущено с поддержкой удалённого JMX мониторинга, самое время настроить интерфейсы и элементы данных в веб-интерфейсе Zabbix.

Настройка JMX интерфейса

Начнём с создания интерфейса типа JMX у интересующего узла сети.

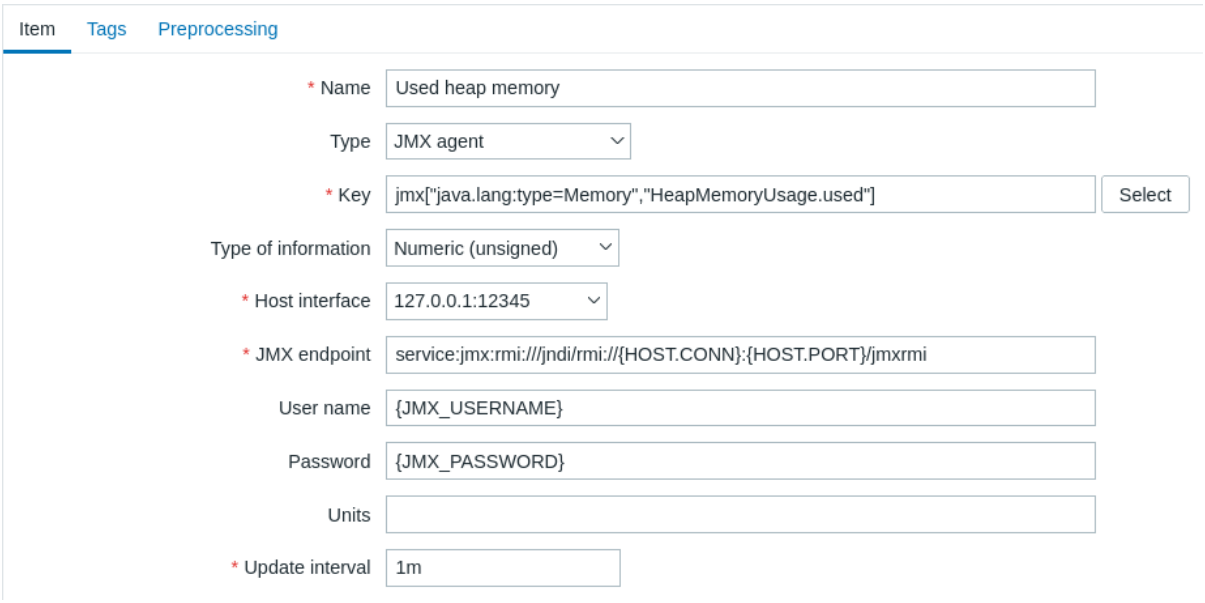



Все обязательные поля ввода отмечены красной звёздочкой.

Добавление элемента данных JMX агента

Для каждого интересующего вас счётчика JMX необходимо добавить элемент данных с типом **JMX агент**, присоединённый к этому интерфейсу.

Ключ на снимке экрана ниже имеет следующий вид: `jmx["java.lang:type=Memory","HeapMemoryUsage.used"]`.




Все обязательные поля ввода отмечены красной звёздочкой.

Поля, требующие информации, специфичной для JMX элементов данных:

| | |
|---|---|
| *Тип (Type)* | Укажите здесь **JMX агент**. |

| | |
|---|---|
| *Ключ (Key)* | Ключ элемента данных `jmx[]` содержит три параметра:<br>**имя объекта** — имя объекта MBean;<br>**имя атрибута** — имя атрибута MBean с опциональными составными именами полей данных, разделёнными точками.<br>**уникальное короткое описание** — уникальное описание, которое позволит создать на одном узле сети несколько JMX элементов данных с одинаковыми именем объекта и именем атрибута (опционально)<br>Подробнее о ключах JMX элементов данных смотрите ниже.<br>Вы можете обнаруживать MBeans и MBean атрибуты, используя элемент данных низкоуровневого обнаружения `jmx.discovery[]`. |
| *JMX endpoint* | Вы можете указать пользовательский JMX endpoint. Убедитесь, что параметры подключения JMX endpoint совпадают с JMX интерфейсом. Это можно сделать при помощи макросов {HOST.*}, как уже сделано в JMX endpoint по умолчанию.<br>Поддерживаются макросы {HOST.*} и пользовательские макросы. |
| *Имя пользователя (User name)* | Укажите имя пользователя (до 255 символов), если вы настроили аутентификацию у вашего Java приложения.<br>Поддерживаются пользовательские макросы. |
| *Пароль (Password)* | Укажите пароль (до 255 символов), если вы настроили аутентификацию у вашего Java приложения.<br>Поддерживаются пользовательские макросы. |

Если вы хотите наблюдать за логическим счётчиком, который может иметь значение «true» или «false», укажите тип информации «Числовой (целое положительное)» и добавьте шаг предварительной обработки «Логический в десятичный» на вкладке Предобработка. Сервер будет сохранять логические значения как 1 или 0, соответственно.

Более подробно о ключах элементов данных JMX

Простые атрибуты

Имя объекта MBean — это не что иное, как строка, которую вы определили в вашем Java приложении. Имя атрибута, с другой стороны, может быть более сложным. В случае, если атрибут возвращает простой тип данных (целое число, строку и т.п.), то беспокоиться не о чем, ключ будет выглядеть примерно так:

`jmx[com.example:Type=Hello,weight]`

В этом примере именем объекта является «com.example:Type=Hello», именем атрибута будет «weight», а тип возвращаемого значения, скорее всего, должен быть «Числовой (с плавающей точкой)».

Атрибуты, возвращающие составные данные

Ключ становится более сложным, когда ваш атрибут возвращает составные данные. Например: именем вашего атрибута является «apple», и он возвращает хеш представляющих его параметров, таких как «weight», «color» и прочее. Тогда ваш ключ может выглядеть примерно так:

`jmx[com.example:Type=Hello,apple.weight]`

Этот пример показывает, как разделяются с помощью точки имя атрибута и ключ хеша. Точно также, если атрибут возвращает вложенные составные данные, их части снова нужно разделять точкой:

`jmx[com.example:Type=Hello,fruits.apple.weight]`

Атрибуты, возвращающие табличные данные

Атрибуты с табличными данными состоят из одного или нескольких составных атрибутов. Если такой атрибут указан в параметре имени атрибута, то значение этого элемента данных вернёт всю структуру атрибута в формате JSON. Значения отдельных элементов из табличных данных можно извлечь с помощью предобработки.

Пример атрибута с табличными данными:

`jmx[com.example:type=Hello,foodinfo]`

Значение элемента данных:

```
[
  {
    "a": "apple",
    "b": "banana",
    "c": "cherry"
  },
```

```
  {
    "a": "potato",
    "b": "lettuce",
    "c": "onion"
  }
]
```

Проблема с точками

Пока все хорошо. Но что, если имя атрибута или ключ хеша содержит символ точки? Вот пример:

```
jmx[com.example:Type=Hello,all.fruits.apple.weight]
```

Это проблема. Как сказать Zabbix'у, что имя атрибута — «all.fruits», а не просто «all»? Как отличить точку, которая является частью имени, от точки, которая разделяет имя атрибута и ключи хешей?

Это возможно; всё, что вам требуется сделать — экранировать точки, которые являются частью имени, обратной косой чертой:

```
jmx[com.example:Type=Hello,all\.fruits.apple.weight]
```

Аналогично, если ваш ключ хеша содержит точку, вам необходимо её экранировать:

```
jmx[com.example:Type=Hello,all\.fruits.apple.total\.weight]
```

Другие проблемы

Символ обратной косой черты в имени атрибута также необходимо экранировать:

```
jmx[com.example:type=Hello,c:\\documents]
```

Для обработки любых других символов в ключе JMX элемента данных, пожалуйста, обратитесь к разделу формата ключа элементов данных.

На самом деле это всё, что нужно сделать. Успешного мониторинга JMX!

Непримитивные типы данных

Имеется возможность работы с пользовательскими MBean, возвращающими непримитивные типы данных, которые переопределяют метод **toString()**.

Использование пользовательского endpoint с JBoss EAP 6.4

Пользовательские endpoint'ы позволяют работать с различными транспортными протоколами, отличными от протокола по умолчанию RMI.

Для иллюстрации этой возможности давайте попытаемся настроить мониторинг JBoss EAP 6.4 в качестве примера. Во-первых, давайте сделаем некоторые предположения:

- У вас уже имеется установленный Zabbix Java gateway. Если нет, тогда вы можете его установить, руководствуясь документацией.
- Zabbix сервер и Java gateway установлены с префиксом `/usr/local/`.
- JBoss уже установлен в `/opt/jboss-eap-6.4/` и запущен в автономном режиме.
- Мы предположим, что все эти компоненты работают на одном и том же узле сети.
- Брандмауэр и SELinux отключены (или настроены должным образом).

Давайте выполним некоторые простые настройки в zabbix_server.conf:

```
JavaGateway=127.0.0.1
StartJavaPollers=5
```

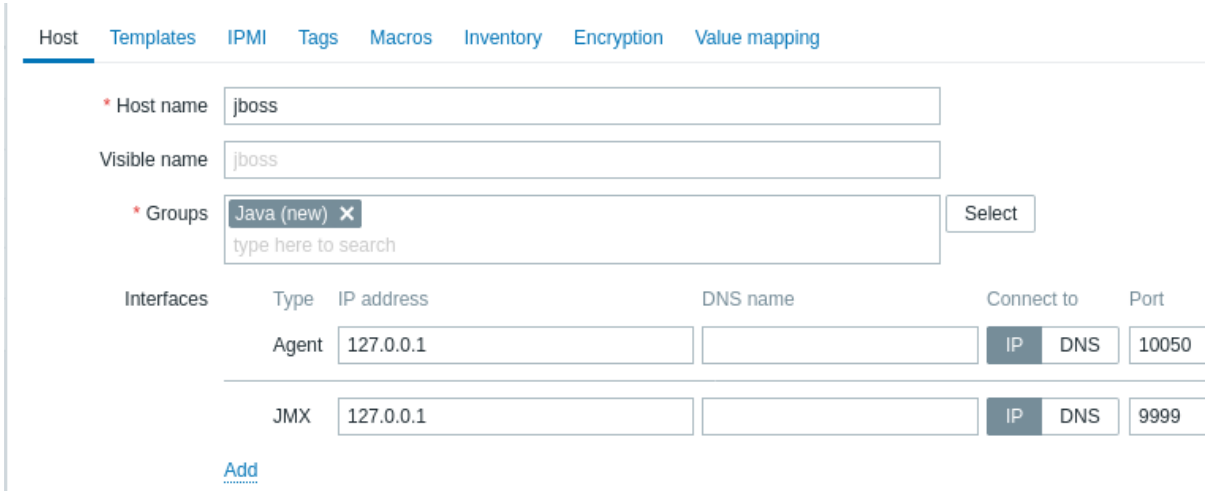
И в файле конфигурации `zabbix_java/settings.sh` (или `zabbix_java_gateway.conf`):

```
START_POLLERS=5
```

Удостоверьтесь, что JBoss слушает свой стандартный порт управления:

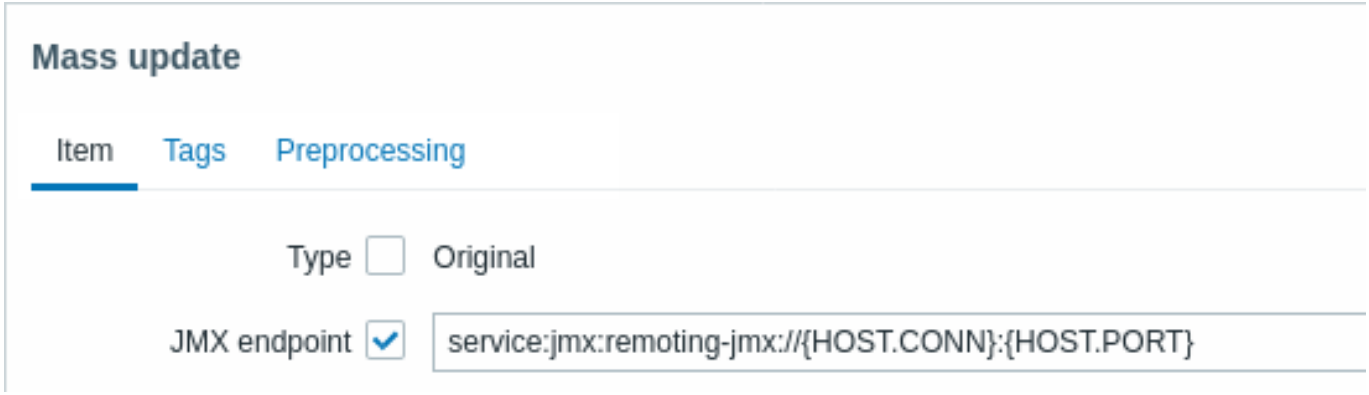
```
$ netstat -natp | grep 9999
tcp        0      0 127.0.0.1:9999          0.0.0.0:*               LISTEN      10148/java
```

Теперь давайте создадим в Zabbix узел сети с JMX интерфейсом 127.0.0.1:9999.




Поскольку мы знаем, что эта версия JBoss использует протокол JBoss Remoting вместо RMI, то мы можем использовать массовое обновление параметра JMX endpoint в нашем шаблоне JMX соответствующим образом:

```
service:jmx:remoting-jmx://{HOST.CONN}:{HOST.PORT}
```




Давайте обновим кэш конфигурации:

```
/usr/local/sbin/zabbix_server -R config_cache_reload
```

Обратите внимание, что сначала может возникнуть ошибка.

```
3. mc [root@centos7-dev]:/home/vagrant/zabbix-3.2.6/src/zabbix_java (ssh)
com.zabbix.gateway.ZabbixException: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at com.zabbix.gateway.JMXItemChecker.getValues(JMXItemChecker.java:97) ~[zabbix-java-gateway-3.4.2.jar:na]
        at com.zabbix.gateway.SocketProcessor.run(SocketProcessor.java:63) ~[zabbix-java-gateway-3.4.2.jar:na]
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [na:1.8.0_144]
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [na:1.8.0_144]
        at java.lang.Thread.run(Thread.java:748) [na:1.8.0_144]
Caused by: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359) ~[na:1.8.0_144]
        at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269) ~[na:1.8.0_144]
        at com.zabbix.gateway.ZabbixJMXConnectorFactory$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gatewa
-3.4.2.jar:na]
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) ~[na:1.8.0_144]
        at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[na:1.8.0_144]
        ... 3 common frames omitted
2017-11-07 13:52:12.644 [pool-1-thread-1] WARN  com.zabbix.gateway.SocketProcessor - error processing request
com.zabbix.gateway.ZabbixException: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at com.zabbix.gateway.JMXItemChecker.getValues(JMXItemChecker.java:97) ~[zabbix-java-gateway-3.4.2.jar:na]
        at com.zabbix.gateway.SocketProcessor.run(SocketProcessor.java:63) ~[zabbix-java-gateway-3.4.2.jar:na]
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [na:1.8.0_144]
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [na:1.8.0_144]
        at java.lang.Thread.run(Thread.java:748) [na:1.8.0_144]
Caused by: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
        at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359) ~[na:1.8.0_144]
        at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269) ~[na:1.8.0_144]
        at com.zabbix.gateway.ZabbixJMXConnectorFactory$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gatewa
-3.4.2.jar:na]
        at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) ~[na:1.8.0_144]
        at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[na:1.8.0_144]
        ... 3 common frames omitted
2017-11-07 13:52:14.889 [Thread-0] INFO  com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885)
as stopped
2017-11-07 13:52:26.167 [main] INFO  com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885) has
tarted
```

«Unsupported protocol: remoting-jmx» означает, что Java gateway не знает, как работать с указанным протоколом. Эту ошибку можно исправить, создав файл `~/needed_modules.txt` со следующим содержимым:

```
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
```

и затем выполнив эту команду:

```
for i in $(cat ~/needed_modules.txt); do find /opt/jboss-eap-6.4 -iname "${i}*.jar" -exec cp '{}' /usr/loc
```

Таким образом, Java gateway будет иметь все необходимые модули для работы с jmx-remoting. Осталось только перезапустить Java gateway, немного подождать и, если вы всё сделали правильно, вы увидите, что данные JMX мониторинга начинают поступать в Zabbix (смотрите также: Последние данные).

#### 14 Вычисляемый элемент данных

Обзор

Вычисляемые элементы данных позволяют выполнять подсчёты на основании значений некоторых существующих элементов данных. Например, вы можете захотеть рассчитать среднее почасовое значение некоторого элемента данных или рассчитать сумму значений для группы элементов данных. Для этого и нужен вычисляемый элемент данных.

Вычисления могут использовать:

- единичные значения отдельных элементов данных
- сложные фильтры для выбора нескольких элементов данных для последующей агрегации (смотрите агрегированные вычисления для получения более подробных сведений)

Вычисляемые элементы данных — это способ создания виртуальных источников данных. Все вычисления выполняются только на стороне Zabbix сервера. Значения периодически рассчитываются на основе использованного арифметического выражения.

Полученные данные записываются в базу данных Zabbix так же, как и по любым другим элементам данных; хранятся значения истории и динамики изменений, можно сгенерировать графики.

> **Note:**
> Если результат вычисления — значение числа с плавающей точкой, оно будет обрезаться до целого числа, если типом информации вычисляемого элемента данных выбрано *Числовой (целое положительное)*.
> Кроме того, если в кэше нет последних данных, а в функции не определён период запроса, Zabbix по умолчанию будет обращаться к базе данных за историческими значениями вплоть до одной недели.

Вычисляемые элементы данных имеют общий синтаксис с выражениями триггеров. В вычисляемых элементах данных допускается сравнение со строками. На вычисляемые элементы данных можно ссылаться макросами или другими такими же объектами, как и в случае любого другого типа элементов данных.

Для использования вычисляемых элементов данных выберите **Вычисляемый** тип элементов данных.

Настраиваемые поля

**Ключ** — уникальный идентификатор элемента данных (в пределах узла сети). Вы можете создать любое имя ключа, используя поддерживаемые символы.

Формулу вычисления необходимо ввести в поле **Формула**. Нет никакой связи между формулой и ключом. Параметры ключа не используются в формуле никоим образом.

Синтаксис простой формулы:

```
    (/      /   ,<     1>,<     2>,...)
```

где:

| | |
|---|---|
| *функция* | Одна из поддерживаемых функций: last, min, max, avg, count и т.д. |
| *узел сети* | Узел сети элемента данных, который используется в вычислении.<br>Текущий узел сети может быть опущен (примерно как в `(//  ,  ,...)`). |
| *ключ* | Ключ элемента данных, который используется в вычислении.<br>Элементы данных, возвращающие значения с бинарным типом данных, не поддерживаются. |
| *параметр(ы)* | Параметры функции, если требуются.<br>Поддерживаются суффиксы времени и суффиксы памяти. |

**Attention:**
Пользовательские макросы в формуле будут раскрыты, если они используются для ссылки на параметр функции, параметр фильтра элементов данных или константу. Пользовательские макросы НЕ будут раскрыты при использовании ссылки на функцию, имя узла сети, ключ элемента данных, параметры ключа элементов данных или оператор.

Для получения более сложной формулы можно использовать комбинацию функций, операторов и скобок. Вы можете использовать все функции и операторы, которые поддерживаются в выражениях триггеров. Логика и приоритет операторов точно такие же.

В отличие от выражений триггеров, Zabbix обрабатывает вычисляемые элементы данных в соответствии с интервалом обновления элемента данных, а не при получении нового значения.

Все элементы данных, на которые имеются ссылки из функций истории в формуле вычисляемого элемента данных, должны существовать и собирать данные. Также, если вы измените ключ элемента данных, на который ссылается вычисляемый элемент данных, вам придётся вручную обновить все формулы, в которых этот ключ используется.

Вычисляемый элемент данных может перейти в неподдерживаемое состояние в нескольких случаях:

- элемент(ы) данных, на которые имеется ссылка
  - не найден
  - деактивирован
  - принадлежит деактивированному узлу сети
  - не поддерживается (за исключением функции nodata() и операторов с неизвестными значениями)
- отсутствуют данные для вычисления функции
- деление на ноль
- использован некорректный синтаксис

Примеры использования

Пример 1

Вычисление процентного отношения свободного места на диске для '/'.

Использование функции **last**:

```
100*last(//vfs.fs.size[/,free])/last(//vfs.fs.size[/,total])
```

Zabbix будет брать последние значения полного и свободного объёма диска и вычислять процентное отношение в соответствии с заданной формулой.

Пример 2

Вычисление 10-минутного усреднения количества значений, обрабатываемых Zabbix.

Использование функции **avg**:

```
avg(/Zabbix Server/zabbix[wcache,values],10m)
```

Обратите внимание, что интенсивное использование вычисляемых элементов данных с большими периодами времени может повлиять на производительность Zabbix сервера.

Пример 3

Вычисление общей пропускной способности на eth0.

Сумма двух функций:

```
last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes])
```

Пример 4

Расчет процента входящего трафика.

Более сложное выражение:

```
100*last(//net.if.in[eth0,bytes])/(last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes]))
```

См. также: Примеры агрегатных вычислений

1. Агрегированные вычисления

Обзор

Агрегированные вычисления — это тип вычисляемых элементов данных, который позволяет собирать информацию с нескольких элементов данных Zabbix сервером, и затем вычислять совокупность, в зависимости от использованной функции агрегирования.

Агрегированные вычисления не требуют наличия какого-либо агента, запущенного на наблюдаемом узле сети.

Синтаксис

Чтобы получить агрегаты, используйте одну из поддерживаемых aggregate functions: `avg`, `max`, `min`, `sum` и т. д. Затем добавьте функцию **foreach** в качестве единственного параметра и ее фильтр элементов данных, чтобы выбрать требуемые элементы данных:

```
aggregate_function(function_foreach(/host/key?[group="host group"],timeperiod))
```

Функция **foreach** (например, *avg_foreach*, *count_foreach* и т. д.) возвращает одно агрегированное значение для каждого выбранного элемента данных. Элементы данных выбираются с помощью фильтра элементов данных (`/host/key?[group="host group"]`) из истории элемента данных. Подробнее см. foreach functions.

Если у некоторых элементов данных нет данных за запрошенный период, они игнорируются при вычислении. Если ни у одного элемента данных нет данных, функция вернет ошибку.

В качестве альтернативы вы можете перечислить несколько элементов данных в качестве параметров для агрегирования:

```
aggregate_function(function(/host/key,parameter),function(/host2/key2,parameter),...)
```

Обратите внимание, что `function` здесь должна быть функцией истории/тренда.

**Note:**
Если результат агрегирования имеет значение с плавающей точкой, оно будет усечено до целого числа, если агрегированный тип информации элемента данных — *Numeric (unsigned)*.

Пользовательские макросы и макросы низкоуровневого обнаружения поддерживаются в:

- параметрах ключа элемента данных
- параметрах функции
- условиях фильтра (имя группы узлов сети и имя тега)
- константах выражения

Агрегированное вычисление может стать неподдерживаемым, если:

- не найден ни один из указанных элементов данных (это может произойти, если ключ элемента данных указан неверно, ни один из элементов данных не существует или все включенные группы указаны неверно)
- нет данных для вычисления функции

Примеры использования

Примеры ключей для агрегированных вычислений.

Пример 1

Общий размер дискового пространства в группе узлов сети «MySQL Servers».

```
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"]))
```

Пример 2

Сумма последних значений на узле сети всех элементов данных, совпадающих с net.if.in[*].

```
sum(last_foreach(/host/net.if.in[*]))
```

Пример 3

Средняя загрузка процессора в группе узлов сети «MySQL Servers».

```
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"]))
```

Пример 4

Усреднение за 5 минут количества запросов в секунду по группе узлов сети «MySQL Servers».

```
avg(avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m))
```

Пример 5

Средняя загрузка CPU по всем узлам сети в нескольких группах узлов сети, у которых имеются заданные теги.

```
avg(last_foreach(/*/system.cpu.load?[(group="Servers A" or group="Servers B" or group="Servers C") and (ta
```

Пример 6

Вычисление с использованием суммы последних значений элементов данных по всей группе узлов сети.

```
sum(last_foreach(/*/net.if.out[eth0,bytes]?[group="video"])) / sum(last_foreach(/*/nginx_stat.sh[active]?[
```

Пример 7

Общее количество неподдерживаемых элементов данных в группе узлов сети «Zabbix servers».

```
sum(last_foreach(/*/zabbix[host,,items_unsupported]?[group="Zabbix servers"]))
```

Пример 8

Сумма последних числовых результатов DNS проверок по всем узлам сети. Примером одного из возможных ключей элемента данных может являться `net.dns[192.0.2.0,example.com,A]` as an example of one possible key.

```
sum(last_foreach(/*/net.dns[*,*,*]))
```

Обратите внимание, что символы-заменители должны совпадать с количеством параметров в ключе (в данном случае `net.dns` имеет три параметра: ip, имя и тип).

Примеры корректного/некорректного синтаксиса

Выражения (включая вызовы функций) нельзя использовать в качестве параметров к функциям истории, динамики изменений или foreach. Однако, сами эти функции могут быть использованы в параметрах других (не относящихся к истории) функций.

| Выражение | Пример |
|---|---|
| Допустимое | `avg(last(/host/key1),last(/host/key2)*10,last(/host/key1)*100)`<br>`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)))` |
| Недопустимое | `sum(/host/key,10+2)`<br>`sum(/host/key, avg(10,2))`<br>`sum(/host/key,last(/host/key2))` |

Обратите внимание, что в таком выражении, как:

```
sum(sum_foreach(//resptime[*],5m))/sum(count_foreach(//resptime[*],5m))
```

нельзя гарантировать, что обе части всегда будут иметь одинаковый набор значений. Пока вычисляется одна часть выражения, может прийти новое значение за запрашиваемый период, и тогда вторая часть выражения будет иметь отличающийся набор значений.

#### 15 Зависимый элемент данных

Обзор

Существуют ситуации, когда один элемент данных собирает несколько метрик одновременно или когда имеет смысл собирать связанные метрики одновременно, например:

- загрузка CPU отдельных ядер
- входящий/исходящий/общий сетевой трафик

Чтобы обеспечить массовый сбор метрик и одновременное использование в нескольких связанных элементах данных, Zabbix поддерживает зависимые элементы данных. Зависимые элементы данных зависят от мастер-элемента, который собирает их данные одновременно, одним запросом. Новое значение для мастер-элемента автоматически заполняет значения зависимых элементов данных. Зависимые элементы данных не могут иметь другой интервал обновления, чем мастер-элемент.

Параметры предварительной обработки Zabbix можно использовать для извлечения той части, которая нужна зависимому элементу данных, из данных мастер-элемента.

Предварительной обработкой управляет процесс `preprocessing manager` вместе с рабочими потоками, которые выполняют шаги предварительной обработки. Все значения с предварительной обработкой (до Zabbix 7.4.1 — все значения), полученные от разных сборщиков данных, проходят через preprocessing manager перед добавлением в кэш истории. Для обмена IPC на основе сокетов используется связь между сборщиками данных (pollers, trappers и т. д.) и процессом предварительной обработки.

Сервер Zabbix или прокси Zabbix (если узел сети контролируется через прокси) выполняют шаги предварительной обработки и обработку зависимых элементов данных.

В качестве мастер-элементов можно задавать элементы данных любого типа, даже зависимые элементы данных. Дополнительные уровни зависимых элементов данных можно использовать для извлечения более мелких частей из значения существующего зависимого элемента данных.

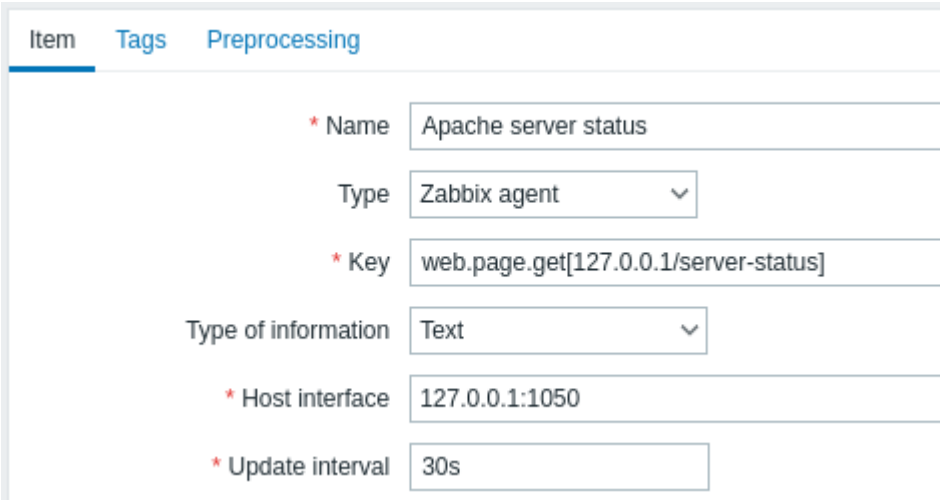
Ограничения

- Допускаются только зависимости на одном и том же узле сети (шаблоне)
- Прототип элемента данных может зависеть от другого прототипа элемента данных или обычного элемента данных с того же узла сети
- Зависимый элемент данных на узле сети с мастер-элементом из шаблона не будет экспортирован в XML

Конфигурация элемента данных

Зависимый элемент данных зависит от своего мастер-элемента по данным. Поэтому сначала должен быть настроен (или уже существовать) **мастер-элемент**:

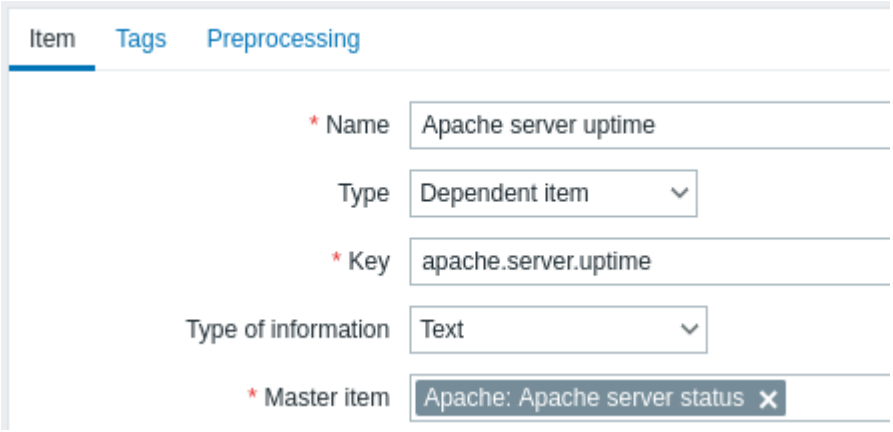
- Перейдите в: *Сбор данных → Узлы сети*
- Нажмите *Элементы данных* в строке узла сети
- Нажмите *Создать элемент данных*
- Введите параметры элемента данных в форме




Все обязательные поля ввода отмечены красной звездочкой.

Нажмите *Добавить*, чтобы сохранить мастер-элемент.

Затем можно настроить **зависимый элемент данных**.

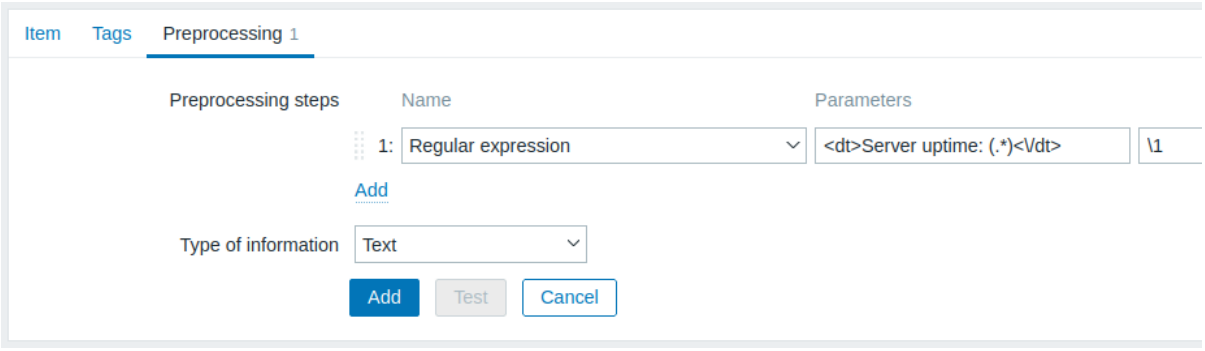



Все обязательные поля ввода отмечены красной звездочкой.

Поля, для которых требуется специальная информация для зависимых элементов данных:

| | |
|---|---|
| *Тип* | Выберите здесь **Зависимый элемент данных**. |
| *Ключ* | Введите ключ, который будет использоваться для идентификации элемента данных. |
| *Мастер-элемент* | Выберите мастер-элемент. Значение мастер-элемента будет использоваться для заполнения значения зависимого элемента данных. |
| *Тип информации* | Выберите тип информации, который будет соответствовать формату сохраняемых данных. |

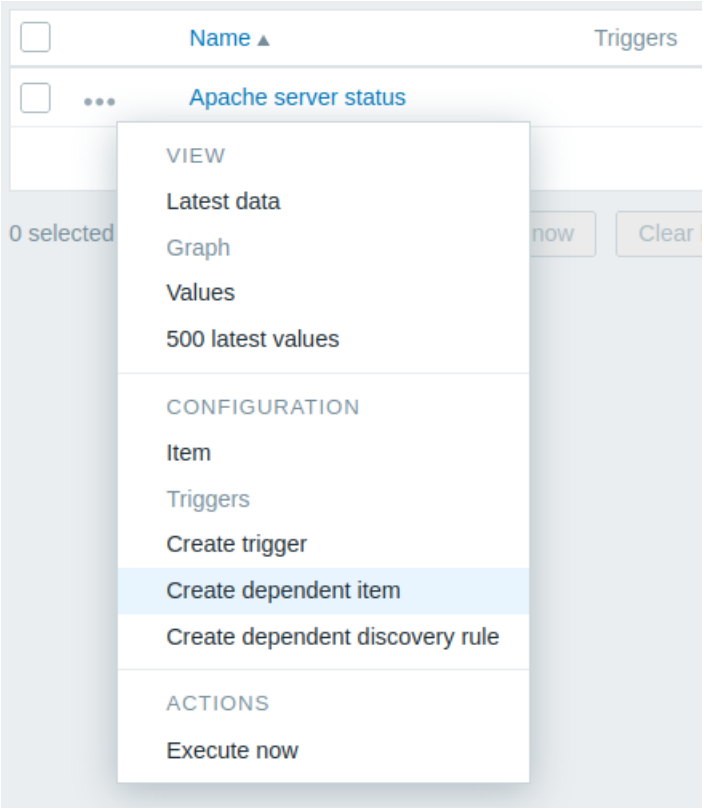
Вы можете использовать предобработку значения элемента данных, чтобы извлечь нужную часть значения мастер-элемента.




Без предобработки значение зависимого элемента данных будет точно таким же, как и значение мастер-элемента.

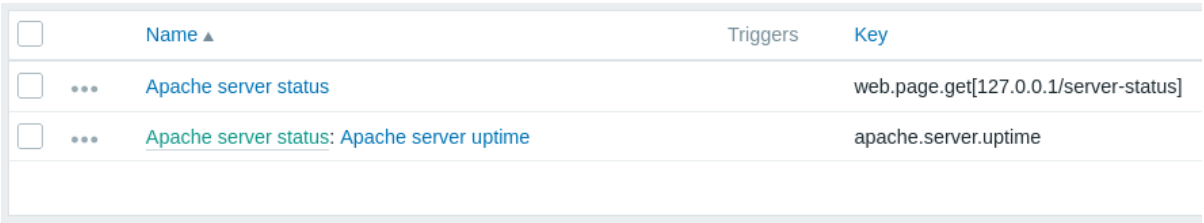
Нажмите *Добавить*, чтобы сохранить зависимый элемент данных.

Быстрый способ создания зависимого элемента данных доступен при нажатии кнопки ••• в списке элементов данных и выборе *Создать зависимый элемент данных*.




Отображение

В списке элементов данных зависимые элементы данных отображаются с именем основного элемента данных в качестве префикса.




Если основной элемент данных удаляется, то будут удалены и все его зависимые элементы данных.

**16 Элемент данных типа скрипт**

Обзор

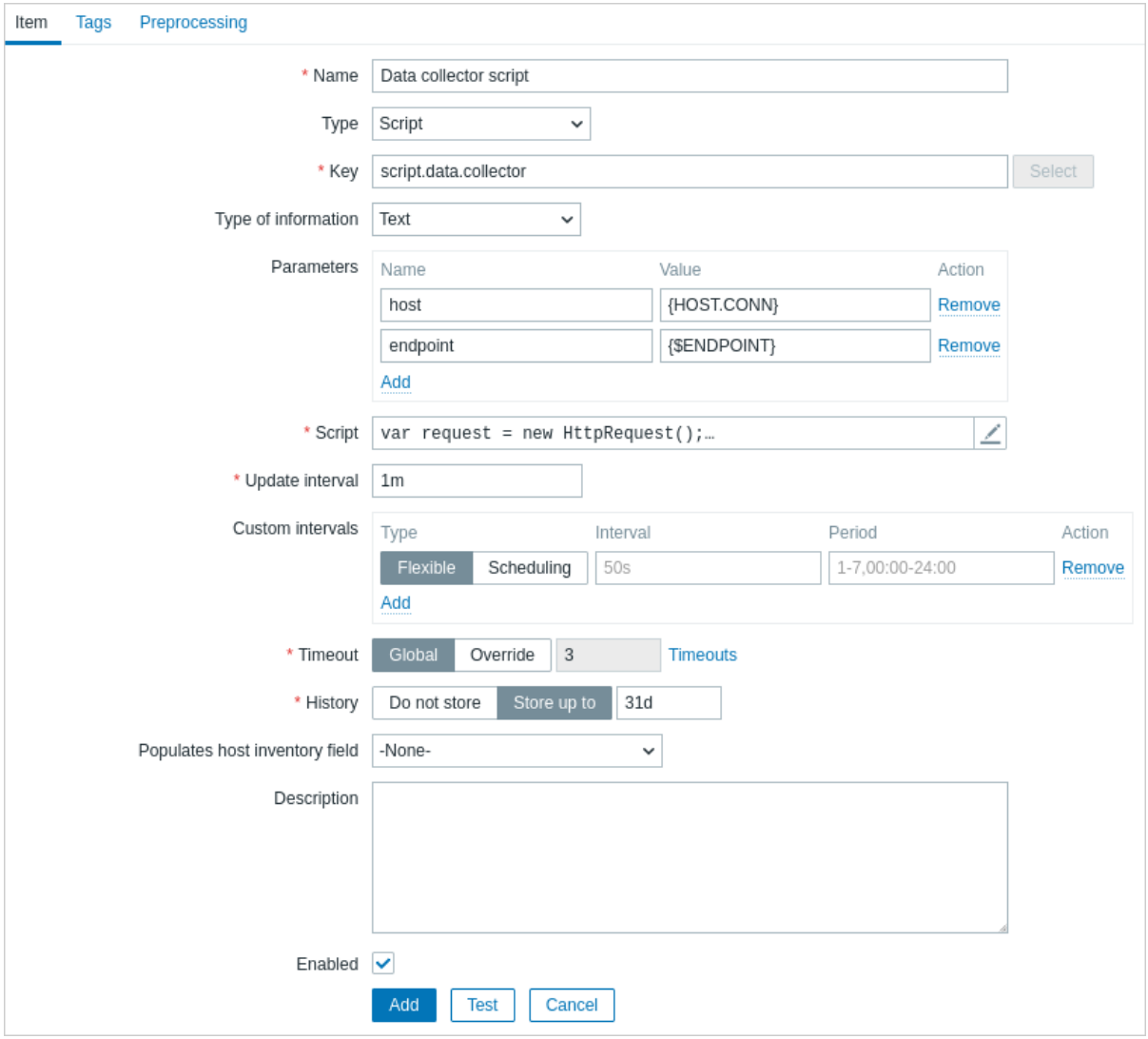
Скриптовые элементы данных можно использовать для сбора данных путём выполнения пользовательского JavaScript кода с возможностью получения данных через HTTP/HTTPS. В дополнение к скрипту можно указать опциональный список параметров (пары из имени и значения) и время ожидания.

Этот тип элементов данных может быть полезен в сценариях сбора данных, требующих нескольких шагов или сложной логики. Например, скриптовый элемент данных можно настроить на выполнение HTTP вызова, последующей обработки неким образом данных, полученных на первом шаге, и передачи преобразованного значения второму HTTP вызову.

Скриптовые элементы данные обрабатываются Zabbix сервером или прокси на уровне поллеров.

Настройка

В поле *Тип* диалога настройки элемента данных выберите Скрипт, затем заполните необходимые поля.




Все обязательные поля ввода отмечены красной звёздочкой.

Поля, которые требуют информацию, специфичную для элементов данных Скрипт:

| Поле | Описание |
|---|---|
| *Ключ (Key)* | Введите уникальный ключ, который будет использоваться для идентификации этого элемента данных. |
| *Параметры (Parameters)* | Укажите переменные, которые будут переданы скрипту как пары атрибутов и значений.<br>Поддерживаются пользовательские макросы. Чтобы узнать, какие встроенные макросы поддерживаются, выполните поиск по запросу «Скриптовый элемент данных» в таблице поддерживаемых макросов. |
| *Скрипт (Script)* | Введите код JavaScript в модальном редакторе, который открывается при нажатии на поле параметров или на кнопку со значком карандаша сразу за полем параметров. Этот код должен предоставлять логику для получения значения метрики.<br>Код имеет доступ ко всем параметрам и всем дополнительным объектам JavaScript, добавленным Zabbix.<br>Смотрите также: Руководство пользователя JavaScript. |
| *Время ожидания (Timeout)* | Время ожидания выполнения JavaScript (1-600s, превышение этого значения вернёт ошибку).<br>Обратите внимание, что в зависимости от скрипта для срабатывания тайм-аута может потребоваться больше времени.<br>Дополнительную информацию о параметре *Timeout* смотрите в разделе общие атрибуты элементов данных. |

Примеры

Простой сбор данных

Получение содержимого *https://www.example.com/release_notes*:

- Создайте элемент данных с типом «Скрипт».

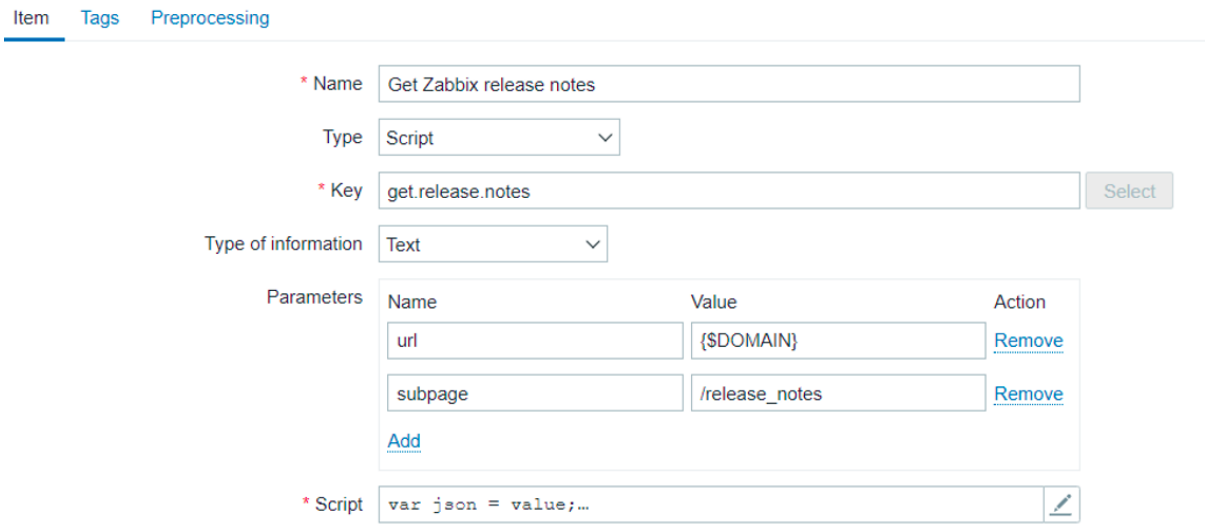
- В поле *Скрипт* введите следующий код:

```
var request = new HttpRequest();
return request.get("https://www.example.com/release_notes");
```

Сбор данных с параметрами

Получение содержимого указанной страницы с использованием параметров:

- Создайте элемент данных с типом «Скрипт» и двумя параметрами:
- **url : {$DOMAIN}** (пользовательский макрос {$DOMAIN} должен быть определён, желательно на уровне узла сети)
- **subpage : /release_notes**




- В поле *Скрипт* введите следующий код:

```
var obj = JSON.parse(value);
var url = obj.url;
var subpage = obj.subpage;
var request = new HttpRequest();
return request.get(url + subpage);
```

Несколько HTTP запросов

Сбор содержимого как *https://www.example.com*, так и *https://www.example.com/release_notes*:

- Создайте элемент данных с типом «Скрипт».

- В поле *Скрипт* введите следующий код:

```
var request = new HttpRequest();
return request.get("https://www.example.com") + request.get("https://www.example.com/release_notes");
```

Журналирование

Добавление записи «Log test» в журнал Zabbix сервера и получение при возврате «1» в качестве значения элемента данных:

- Создайте элемент данных с типом «Скрипт».

- В поле *Скрипт* введите следующий код:

```
Zabbix.log(3, 'Log test');
return 1;
```

### 17 Элемент данных браузера

Обзор

Элементы данных Browser позволяют отслеживать сложные веб-сайты и веб-приложения с помощью браузера.

**Attention:**
Поддержка элементов данных Browser в настоящее время является экспериментальной.

Элементы данных Browser собирают данные, выполняя заданный пользователем код JavaScript и получая данные по HTTP/HTTPS. Этот элемент данных может имитировать такие действия, связанные с браузером, как щелчки, ввод текста, переходы по веб-страницам и другие действия пользователя на веб-сайтах или в веб-приложениях.

Помимо скрипта, можно указать необязательный список параметров (пары имя и значение) и тайм-аут.

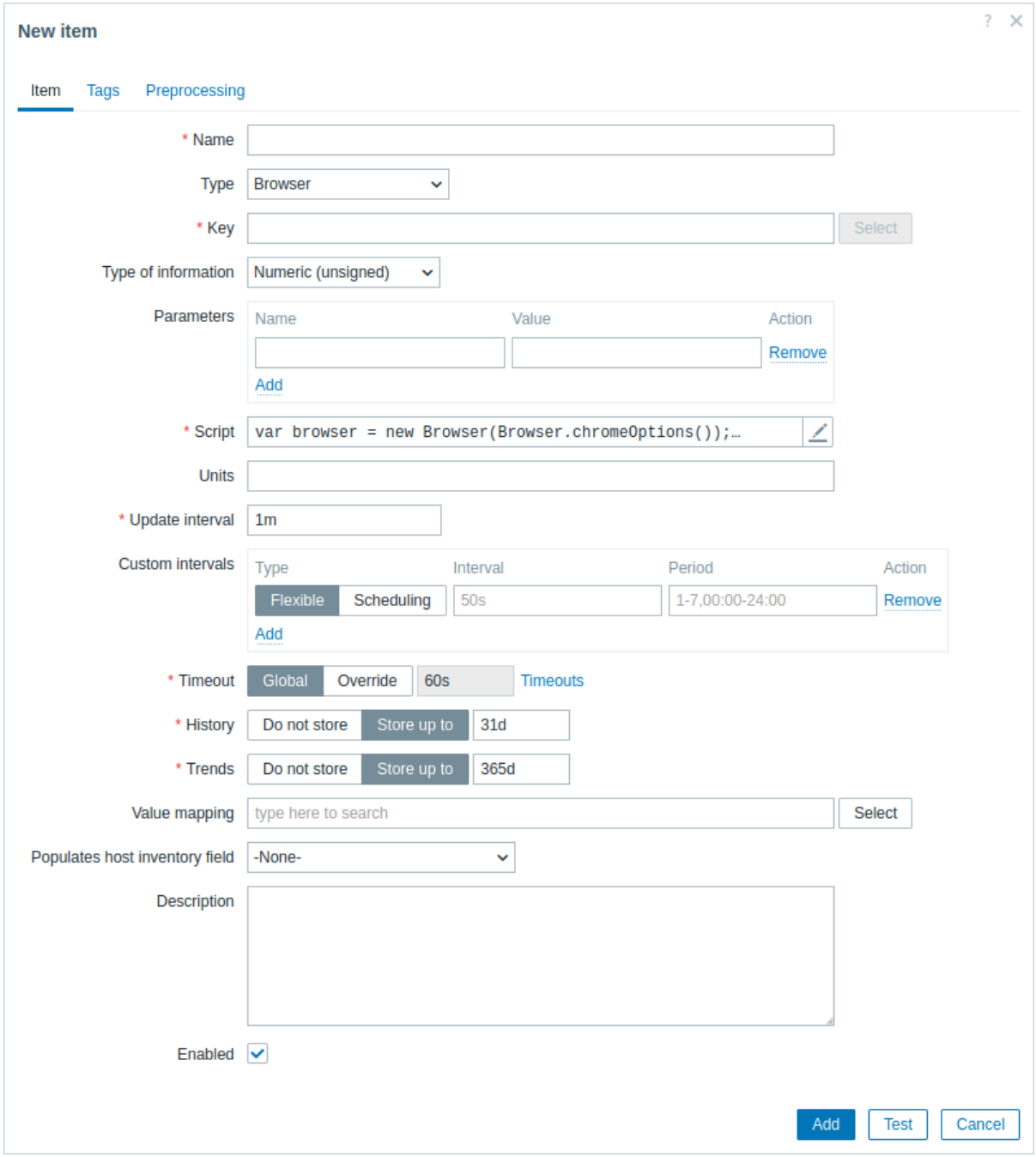
**Attention:**
Этот элемент данных частично реализует стандарт W3C WebDriver с использованием либо Selenium Server, либо обычного WebDriver (например, ChromeDriver) в качестве конечной точки веб-тестирования. Чтобы элемент данных работал, задайте параметр `WebDriverURL` в файле конфигурации Zabbix сервер/прокси для конечной точки (если используется ChromeDriver, см. Security Considerations). Для повышения производительности рекомендуется использовать отдельный сервер для среды веб-тестирования.

Проверки элементов данных Browser выполняются и обрабатываются browser pollers сервера Zabbix или прокси. При необходимости можно изменить количество предварительно созданных экземпляров browser pollers в параметре `StartBrowserPollers` файла конфигурации Zabbix сервер/прокси.

Для мониторинга сложных веб-сайтов и веб-приложений доступен шаблон Website by Browser в виде готового шаблона.

Настройка

В поле *Тип (Type)* диалога настроек элемента данных выберите Браузер и заполните обязательные поля.




Все обязательные поля ввода отмечены красной звёздочкой.

Поля, требующие информации, специфичной для браузерного элемента данных:

| Поле | Описание |
|---|---|
| *Ключ (Key)* | Введите уникальный ключ, который будет использоваться для идентификации элемента данных. |
| *Параметры (Parameters)* | Укажите переменные, которые будут переданы в скрипт как пары атрибут и значение. Поддерживаются пользовательские макросы. Чтобы узнать, какие встроенные макросы поддерживаются, выполните поиск по запросу «Браузерный элемент данных» в таблице поддерживаемых макросов. |
| *Скрипт (Script)* | Введите код JavaScript в модальном редакторе, который открывается при щелчке по полю параметра или по значку карандаша рядом с ним. Этот код должен предоставлять логику для возврата значения метрики.<br>Код имеет доступ ко всем параметрам, всем дополнительным объектам JavaScript и объектам JavaScript браузерного элемента данных, добавленным Zabbix.<br>Смотрите также: Руководство по JavaScript. |

| Поле | Описание |
|---|---|
| *Время ожидания (Timeout)* | Тайм-аут выполнения JavaScript (1-600s; при его превышении будет возвращена ошибка). Обратите внимание, что в зависимости от скрипта для срабатывания тайм-аута может потребоваться больше времени.<br>Дополнительную информацию о параметре *Timeout* смотрите в разделе общие атрибуты элементов данных. |

Примеры

Пример настройки Zabbix для мониторинга веб-сайтов с использованием шаблона *Website by Browser* см. в разделе Мониторинг веб-сайтов с помощью элементов данных Browser.

Скрипт по умолчанию

Следующий скрипт:

1. Инициализирует сессию браузера.
2. Переходит по указанному URL-адресу.
3. Собирает записи производительности и статистику сессии и возвращает их в виде строки JSON.

В поле *Скрипт (Script)* введите:

```
var browser = new Browser(Browser.chromeOptions());

try {
    browser.navigate("http://example.com");
    browser.collectPerfEntries();
}
finally {
    return JSON.stringify(browser.getResult());
}
```

Инициализация браузера с пользовательскими возможностями

Следующий скрипт:

1. Инициализирует сеанс браузера для доступного браузера на основе первого совпадающего браузера в порядке, указанном в скрипте.
2. Определяет возможности браузера, включая стратегию загрузки страницы и параметры, специфичные для каждого браузера, такие как режим headless для браузеров Chrome, Firefox и Microsoft Edge.

В поле *Script* введите:

```
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"
            }
        ]
    }
});
```

Инициализация браузера с GUI

По умолчанию сеансы браузера (за исключением Safari) инициализируются в режиме headless, то есть графический пользовательский интерфейс (GUI) браузера не отображается.

Следующий скрипт инициализирует сеанс браузера с включенным GUI.

Обратите внимание: если WebDriver не может найти исполняемый файл браузера, вы можете указать путь вручную.

```
var opts = Browser.chromeOptions();
opts.capabilities.alwaysMatch['goog:chromeOptions'].args = [];

//                  Firefox   GUI,                    :
// var opts = Browser.firefoxOptions();
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].binary = 'usr/bin/firefox';
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].args = [];

//                  Microsoft Edge   GUI,                     :
// 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:**
> Если ваши тесты выполняются на удаленном сервере или в контейнере, вы можете использовать клиент Virtual Network Computing (VNC) для подключения к VNC-серверу машины. Это позволяет удаленно просматривать графический интерфейс браузера и взаимодействовать с ним.

Создание снимков экрана

Следующий скрипт:

1. Инициализирует сеанс браузера.
2. Устанавливает размер области просмотра браузера, чтобы определить размер снимка экрана (указывается в параметрах, см. ниже).
3. Переходит по URL (указывается в параметре, см. ниже).
4. Собирает статистику сеанса, делает снимок экрана и добавляет его в собранную статистику.
5. Обрабатывает ошибки, захватывая сообщения об ошибках и снимок экрана.
6. Возвращает собранные результаты в виде строки JSON.

Скрипт также использует параметры из формы настройки элемента данных:

- webURL - http://example.com
- width - 1920
- height - 1080

В поле *Script* введите:

```
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);
}
```

Проверка входа в Zabbix

Следующий скрипт:

1. Инициализирует сессию браузера.
2. Переходит на страницу (указывается как параметр, смотрите ниже).
3. Вводит имя пользователя и пароль (указываются как параметры, смотрите ниже).
4. Находит и нажимает кнопку входа.
5. Находит и нажимает кнопку выхода.
6. Собирает данные о производительности до и после входа, а также после выхода.
7. Обрабатывает ошибки, фиксируя сообщения об ошибках и снимок экрана.
8. Возвращает собранные результаты в виде строки JSON.

Скрипт также использует параметры из диалога настроек элемента данных:

- webURL — http://{HOST.CONN}/index.php
- username — {$USERNAME}
- password — {$PASSWORD}

В поле *Скрипт (Script)* введите:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); //            JSON,             ,          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);
}
```

Поиск ссылок

Следующий скрипт:

1. Инициализирует сеанс браузера.
2. Определяет функцию для удаления дубликатов элементов из массива (см. шаг 5).
3. Переходит на страницу (указанную в параметрах, см. ниже).
4. Находит ссылки на странице.
5. Удаляет повторяющиеся ссылки, чтобы они были уникальными.
6. Извлекает только ссылки, начинающиеся с "http".
7. Форматирует извлеченные ссылки в определенную структуру.
8. Обрабатывает ошибки, захватывая сообщения об ошибках и снимок экрана.
9. Возвращает собранные результаты в виде строки JSON.

Скрипт также использует параметры из формы конфигурации элемента данных item configuration form:

- scheme - {$WEBSITE.SCHEME}
- domain - {$WEBSITE.DOMAIN}
- path - {$WEBSITE.PATH}

В поле *Script* введите:

```
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);
}
```

#### 4. История и динамика изменений

Обзор

История и динамика изменений — два способа хранения в Zabbix собранных данных.

В то время как история хранит каждое собранное значение, динамика изменений хранит усреднённую информацию за каждый час, что, следовательно, менее требовательно к ресурсам.

Хранение истории

Вы можете указать, как много дней будет храниться история:

- в диалоге свойств элемента данных
- при массовом обновлении элементов данных
- при настройке задач очистки истории

Любые более старые данные будут удалены с помощью автоматической очистки базы данных (Housekeeper).

Общий настоятельный совет — хранить историю как можно меньшее количество дней и, таким образом, не перегружать базу данных большим количеством значений истории.

Вместо долговременного хранения истории вы можете хранить больше данных динамики изменений. Например, вы можете хранить историю 14 дней, а динамику изменений — 5 лет.

Вы можете прикинуть, сколько потребуется места для истории в сравнении с динамикой изменений, используя страницу оценки размера базы данных.

При меньшем хранении истории, вы всё равно сможете просматривать старые данные на графиках, поскольку для отображения старых данных графики будут использовать значения динамики изменений.

**Attention:**
Если история установлена в значение «0», то элемент данных будет обновлять только зависимые элементы данных и поля инвентарных данных. Функции триггера не будут вычисляться, так как триггерные вычисления основаны только на данных истории.

**Note:**
В качестве альтернативного подхода к хранению истории ознакомьтесь с функционалом экспорта истории в подгружаемых модулях.

Хранение динамики изменений

Динамика изменений является встроенным механизмом обработки данных истории, где по числовым типам данных для каждого часа хранятся значения минимума, максимума, среднее и общее количество значений.

Вы можете указать, как много дней будет храниться динамика изменений:

- в диалоге свойств элемента данных
- при массовом обновлении элементов данных
- при настройке задач очистки истории

Обычно динамика изменений может храниться намного дольше чем история. Любые более старые данные будут удалены при помощи автоматической очистки базы данных (Housekeeper).

Во время работы, по мере поступления новых значений, Zabbix сервер собирает и хранит динамику изменений в специальном кэше динамики изменений. Сервер сбрасывает динамику изменения **предыдущего часа** каждого элемента данных в базу данных (при этом она становится доступной из веб-интерфейса) в следующих случаях:

- сервер получает первое за текущий час значение элемента данных
- до конца текущего часа остаётся 5 или менее минут и всё ещё не пришло ни одно значение элемента данных за текущий час
- сервер останавливается

Таким образом, самое раннее, когда можно увидеть новое значение динамики изменения текущего часа в веб-интерфейсе — это начало следующего часа (если элемент данных обновляется регулярно), а самое позднее — конец следующего часа (если элемент данных обновляется редко), что даёт 2 часа максимум.

Если в момент сбрасывания значения динамики изменения сервер замечает, что значение за соответствующий час в базе данных уже имеется (например, сервер был перезапущен в середине часа), ему, возможно, придётся выполнять SQL инструкции обновления вместо вставки. Поэтому на больших инсталляциях, если необходим перезапуск сервера, постарайтесь спланировать его так, чтобы остановка сервера произошла в конце часа, а запуск — в начале следующего, чтобы избежать перекрытие данных динамики изменений.

При работе со значениями динамики изменений Zabbix сервер к таблицам истории базы данных не обращается.

**Attention:**
Если динамика изменений установлена в значение «0», Zabbix сервер вообще не будет считать и хранить динамику изменений.

**Note:**
Динамика изменений вычисляется и сохраняется с использованием того же типа данных, что и исходные значения. В результате среднее значение вычисления значений целочисленных типов данных округляется, и чем меньше интервал значений, тем меньше будет точность результата. Например, если элемент данных имеет значения 0 и 1, то средним значением будет 0, а не 0.5.
Перезапуск сервера также может привести к потере точности вычислений средних значений у целочисленных типов данных за текущий час.

### 5. Пользовательские параметры

Обзор

Иногда вы можете захотеть выполнять через агент проверку, которая не предопределена в Zabbix. В этом случае вам на помощь придут пользовательские параметры.

Вы можете написать команду, которая вернёт необходимые вам данные, и добавить эту команду как пользовательский параметр в файл конфигурации агента (параметр конфигурации «UserParameter»).

Пользовательский параметр имеет следующий синтаксис:

```
UserParameter=<  >,<    >
```

Как вы можете видеть, пользовательский параметр также содержит ключ. Ключ потребуется при настройке элемента данных. Введите ключ по вашему выбору так, чтобы на него было легко ссылаться (он должен быть уникальным в пределах узла сети).

Перезапустите агента или используйте опцию управления работой агента, чтобы агент подхватил новый параметр, например:

```
zabbix_agentd -R userparameter_reload
```

Затем, при настройке элемента данных введите этот ключ, чтобы сослаться на команду из пользовательского параметра, которую вы хотите выполнять.

Команды из пользовательских параметров выполняются Zabbix агентом. Может быть возвращено до 16 МБ данных до шагов предварительной обработки элементов данных.

В операционных системах UNIX используется интерпретатор командной строки **/bin/sh**. Пользовательские параметры подчиняются времени ожидания проверок агента; если время ожидания будет превышено, созданный по пользовательским параметрам процесс будет завершён.

Смотрите также:

- Пошаговую инструкцию по использованию пользовательских параметров.
- Выполнение команд

Примеры простых пользовательских параметров

Простая команда:

```
UserParameter=ping,echo 1
```

Для элемента данных с ключом «ping» агент будет всегда возвращать «1».

Более сложный пример:

```
UserParameter=mysql.ping,mysqladmin -uroot ping|grep -c alive
```

Агент будет возвращать «1», если MySQL сервер доступен, «0» — в противном случае.

Гибкие пользовательские параметры

Гибкие пользовательские параметры допускают в ключе параметры. В этом случае гибкие пользовательские параметры могут быть основой для создания нескольких элементов данных.

Гибкие пользовательские параметры имеют следующий синтаксис:

```
UserParameter=  [*],
```

| Параметр | Описание |
|---|---|
| **Ключ** | Уникальный ключ элемента данных. [*] задаёт, что ключ может принимать параметры из скобок. Параметры указываются при настройке элемента данных. |
| **Команда** | Команда, которая выполняется для получения значения ключа.<br>*Только для гибких пользовательских параметров*:<br>Вы можете использовать в команде позиционные ссылки $1...$9 для того, чтобы сослаться на соответствующий параметр в ключе элемента данных.<br>Zabbix разбирает параметры, заключенные в [ ] ключа элемента данных, и заменяет $1,...,$9 в команде соответственно.<br>$0 будет заменена оригинальной командой (до раскрытия $0,...,$9) для выполнения.<br>Позиционные ссылки интерпретируются Zabbix агентом независимо от того, заключены ли они в двойные (") или в одинарные (') кавычки.<br>Для использования позиционных ссылок без изменения, укажите двойной символ доллара — например, awk '{print $$2}'. В этом случае $$2 фактически превратится в $2 при выполнении команды. |

**Attention:**
Позиционные ссылки, начинающиеся со знака $, ищутся и заменяются Zabbix агентом только в случае гибких пользовательских параметров. В случае простых пользовательских параметров такая обработка ссылок пропускается и, следовательно, экранирование любых знаков $ не требуется.

**Attention:**
По умолчанию некоторые символы в пользовательских параметрах не разрешены. Смотрите раздел UnsafeUserParameters для получения полного списка таких символов.

Пример 1

Очень простой пример:

```
UserParameter=ping[*],echo $1
```

Мы можем создать неограниченное количество элементов данных для наблюдения за чем угодно, указав в таком формате: ping[что угодно].

- ping[0] — будет всегда возвращать «0»
- ping[aaa] — будет всегда возвращать «aaa»

Пример 2

Давайте добавим больше смысла!

```
UserParameter=mysql.ping[*],mysqladmin -u$1 -p$2 ping | grep -c alive
```

Этот параметр можно использовать для мониторинга доступности баз данных MySQL. Параметрами мы можем передать имя пользователя и пароль:

```
mysql.ping[zabbix,our_password]
```

Пример 3

Сколько строк в файле, которые соответствуют регулярному выражению?

```
UserParameter=wc[*],grep -c "$2" $1
```

Этот параметр можно использовать для подсчёта количества строк в файле.

```
wc[/etc/passwd,root]
wc[/etc/services,zabbix]
```

Результат команды

Результирующим значением команды является стандартный вывод вместе со стандартным выводом ошибок.

> **Attention:**
> Текстовый элемент данных (с типами информации символ, журнал или текст) не станет неподдерживаемым в случае вывода в стандартный вывод ошибок.

Возвращаемое значение ограничено 16 МБ (включая конечные пробелы, которые усекаются); также применяются ограничения баз данных.

Пользовательские параметры, которые возвращают текст (для типов информации символ, журнал, текстовый ) могут возвращать пробел. В случае ошибочного результата элемент данных станет неподдерживаемым.

**1. Расширение Zabbix агентов**

В этом руководстве приведены пошаговые инструкции, как расширить функциональность Zabbix агента с помощью пользовательского параметра.

Шаг 1

Напишите скрипт или составьте командную строку для получения необходимого параметра.

Например, мы можем написать следующую команду, чтобы получить общее количество запросов, выполненных сервером MySQL:

```
mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

При выполнении команда возвращает общее количество SQL запросов.

Шаг 2

Добавьте команду в zabbix_agentd.conf:

```
UserParameter=mysql.questions,mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

**mysql.questions** — это уникальный идентификатор. Это может быть любой допустимый идентификатор ключа, например, *queries*.

Протестируйте этот параметр, используя Zabbix агент с флагом ”-t” (если запускаете под root, стоит отметить, что агент может иметь отличающиеся права при работе демоном):

```
zabbix_agentd -t mysql.questions
```

Шаг 3

Перезагрузите пользовательские параметры из файла конфигурации, выполнив:

```
zabbix_agentd -R userparameter_reload
```

Вы также можете перезапустить агента вместо выполнения команды управления работой.

Протестируйте этот параметр, используя утилиту zabbix_get.

Шаг 4

Добавьте новый элемент данных с Ключом=mysql.questions к наблюдаемому узлу сети. Тип этого элемента данных должен быть или Zabbix агент, или Zabbix агент (активный).

Помните, очень важно указать корректный тип возвращаемого значения на Zabbix сервере. Иначе Zabbix не будет принимать значения.

#### 6. Счётчики производительности Windows

Обзор

Вы можете эффективно отслеживать счётчики производительности Windows, используя ключ *perf_counter[]*.

Например:

```
perf_counter["\Processor(0)\Interrupts/sec"]
```

или

```
perf_counter["\Processor(0)\Interrupts/sec", 10]
```

Для получения более подробной информации об этом ключе и его эквиваленте только для английского языка `perf_counter_en`, смотрите специфичные ключи элементов данных для Windows.

Чтобы получить полный список счётчиков производительности для мониторинга, вы можете выполнить:

```
typeperf -qx
```

Вы также можете использовать низкоуровневое обнаружение для обнаружения нескольких экземпляров объектов счётчиков производительности Windows и автоматизировать создание элементов данных perf_counter по множеству экземпляров объектов.

Числовое представление

Windows поддерживает числовые представления (индексы) объектов и имён счётчиков производительности. Zabbix поддерживает такие числовые представления в виде параметров к ключам элементов данных `perf_counter`, `perf_counter_en` и в параметрах конфигурации `PerfCounter`, `PerfCounterEn`.

Однако, если вы не можете гарантировать, что числовые индексы сопоставляются с корректными строками на отдельных хостах, не рекомендуется их использовать. Если вам нужно создать переносимые элементы данных, которые будут работать на разных узлах сети и с разными локализованными версиями Windows, вы можете использовать ключ `perf_counter_en` или параметр конфигурации `PerfCounterEn`, которые позволяют использовать имена на английском языке независимо от языковых настроек системы.

Для того чтобы найти цифровые эквиваленты, запустите **regedit**, а затем найдите *Counter* в *HKEY_LOCAL_MACHINE\SOFTWARE\Microso NT\CurrentVersion\Perflib\009*.

Запись в реестре содержит информацию наподобие этой:

```
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
....
```

Здесь вы можете найти соответствующие числа для каждой части строки счётчика производительности, такой как «\System\% Processor Time»:

```
System → 2
% Processor Time → 6
```

Затем вы можете использовать эти числа для преобразования пути в числа:

```
\2\6
```

Параметры счетчиков производительности

Вы можете использовать некоторые параметры PerfCounter для отслеживания счётчиков производительности Windows.

Например, вы можете добавить следующее содержимое в файл конфигурации Zabbix агента:

```
PerfCounter=UserPerfCounter1,"\Memory\Page Reads/sec",30

PerfCounter=UserPerfCounter2,"\4\24",30
```

С такими параметрами вы можете просто использовать ключи *UserPerfCounter1* или *UserPerfCounter2* при создании соответствующих элементов данных.

Не забудьте перезапустить Zabbix агента после внесения изменений в файл конфигурации.

**7. Массовое обновление**

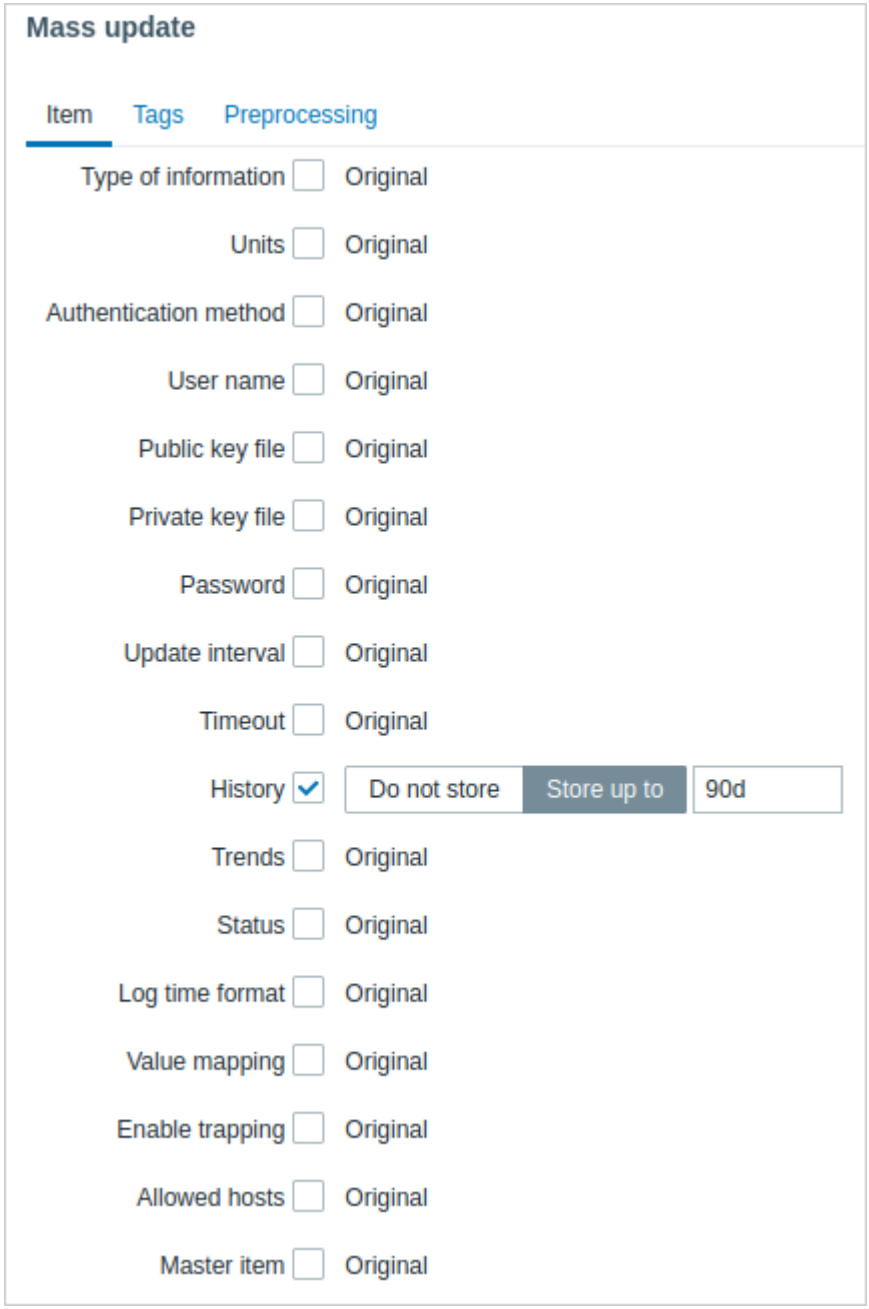
Обзор

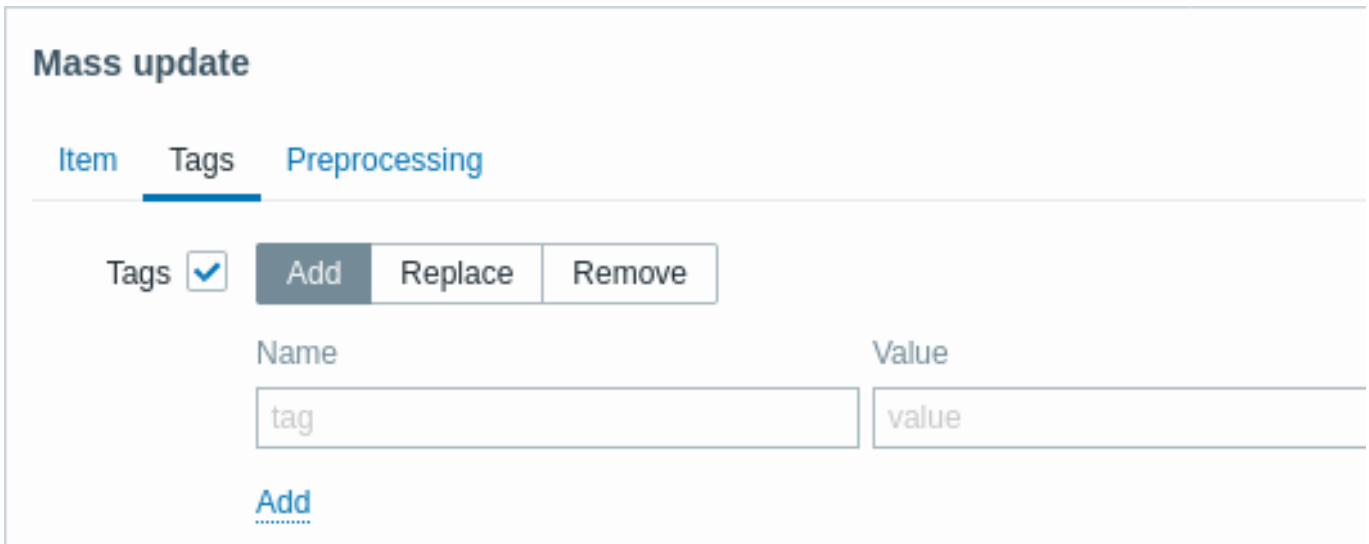
Иногда вы можете захотеть изменить какие-нибудь атрибуты у нескольких элементов данных одновременно. Вместо того, чтобы открывать на редактирование каждый элемент данных по отдельности, вы можете воспользоваться для этого функцией массового обновления.

Использование массового обновления

Чтобы массово обновить некоторые элементы данных, выполните следующие действия:

- Установите флажки у элементов данных, которые нужно обновить, в списке
- Нажмите *Массовое обновление* под списком
- Перейдите на вкладку с нужными атрибутами (*Элемент данных*, *Теги* или *Предобработка*)
- Установите флажки у атрибутов, которые нужно обновить
- Введите новые значения для атрибутов



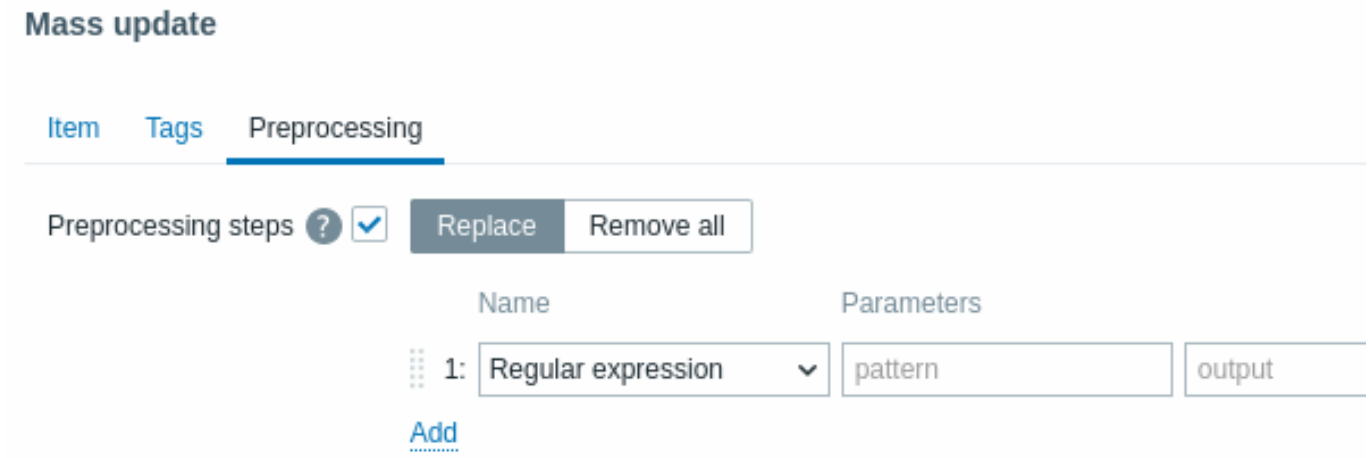




Параметр *Теги* позволяет:

- *Добавить* - добавить указанные теги к элементам данных (теги с одинаковым именем, но разными значениями не считаются ”дубликатами” и могут быть добавлены к одному и тому же элементу данных).
- *Заменить* - удалить указанные теги и добавить теги с новыми значениями

- *Удалить* - удалить указанные теги из элементов данных

В тегах поддерживаются пользовательские макросы, макросы {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} и {HOST.ID}.




Установите флажок рядом с *Шаги предобработки*, чтобы активировать массовое обновление шагов предобработки.

Массовое обновление *Предобработка* позволяет:

- *Заменить* - заменить существующие шаги предобработки у элементов данных на указанные ниже шаги предобработки
- *Удалить все* - удалить все шаги предобработки из элементов данных

После завершения нажмите *Обновить*.

**8. Преобразование значений**

Обзор

Для более «человеческого» представления полученных значений, вы можете использовать преобразования значений, которые содержат соответствия между числовыми либо строковыми значениями и строковыми представлениями.

Например, элемент данных, который имеет значения «0» или «1», может использовать преобразование значений для представления этих значений в более читабельном виде:

- 0 → Недоступен
- 1 → Доступен

Преобразование значений для видов резервного копирования данных может быть настроено следущим образом:

- F → Полный
- D → Дифференциальный
- I → Инкрементный

Диапазоны значений напряжения можно отображать так:

- <=209 → Низкое
- 210-230 → OK
- >=231 → Высокое

Преобразования значений используются в веб-интерфейсе Zabbix и в оповещениях, отсылаемых по каналам оповещений.

**Attention:**
Подстановка полученного значения его настроенным представлением выполняется как веб-интерфейсом Zabbix, так и сервером; однако, сервер обрабатывает подстановки только в следующих случаях:<br><br>
- при заполнении инвентарных данных узла сети;
- при раскрытии поддерживаемых макросов {ITEM.VALUE}, {ITEM.LASTVALUE}, {EVENT.OPDATA} и {EVENT.CAUSE.OPDATA}.

Преобразования значений настраиваются на уровне шаблонов или узлов сети. Сразу после добавления, эти преобразования значений становятся доступными для всех элементов данных соответствующего шаблона или узла сети. При

настройке элементов данных выберите в параметре *Преобразование значений (Value mapping)* имя ранее добавленного преобразования значений.

**Note:**
Наследование преобразований значений отсутствует — узлы сети и шаблоны не наследуют преобразования значений от присоединённых шаблонов. Элементы данных шаблона на узле сети продолжат использовать преобразования значений, настроенные в своём шаблоне.

**Note:**
Преобразования значений можно использовать с элементами данных, которые имеют типы информации *Числовой (целое положительное)*, *Числовой (с плавающей точкой)* и *Символ*.

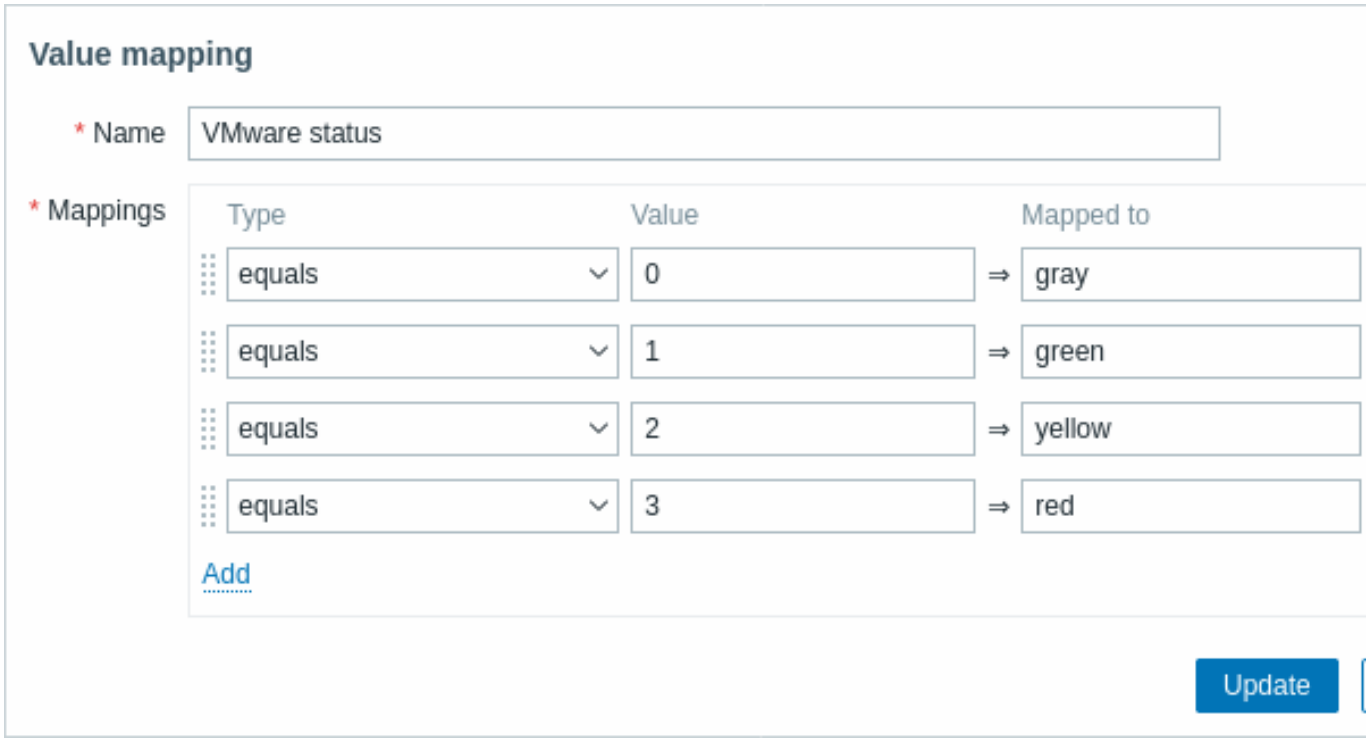
Преобразования значений экспортируются/импортируются вместе с соответствующим шаблоном или узлом сети. Их можно обновлять массово, используя диалоги массового обновления узла сети и шаблона.

Конфигурация

Чтобы настроить сопоставление значений, выполните следующие действия:

1. Откройте форму конфигурации узла сети или шаблона.

2. На вкладке *Сопоставление значений* нажмите *Добавить*, чтобы добавить новое сопоставление значений, или нажмите на имя существующего сопоставления, чтобы отредактировать его.



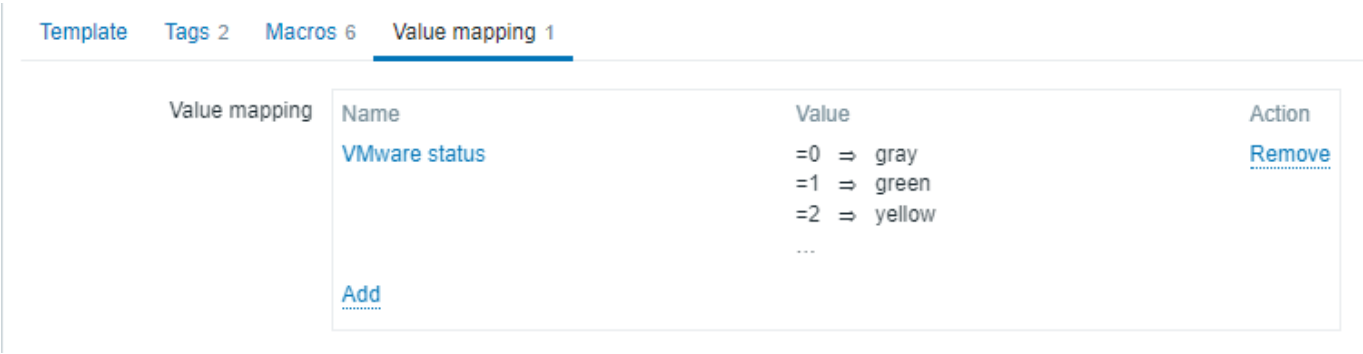

Параметры сопоставления значений:

| Параметр | Описание |
|---|---|
| *Имя Сопоставления* | Уникальное имя набора сопоставлений значений.<br>Отдельные правила для сопоставления числовых/строковых значений со строковыми представлениями.<br><br>Сопоставление применяется в порядке следования правил, которые можно менять местами перетаскиванием. |

| Параметр | Описание |
|---|---|
| *Тип* | Тип сопоставления:<br>**equals** - будут сопоставляться равные значения;<br>**is greater than or equals** - будут сопоставляться равные или большие значения;<br>**is less than or equals** - будут сопоставляться равные или меньшие значения;<br>**in range** - будут сопоставляться значения в диапазоне; диапазон задается как <number1>-<number2> или <number>; поддерживаются несколько диапазонов (например, 1-10,101-110,201);<br>**regexp** - будут сопоставляться значения, соответствующие регулярному выражению (глобальные регулярные выражения не поддерживаются);<br>**default** - будут сопоставляться все оставшиеся значения, не имеющие специальных сопоставлений.<br><br>Для сопоставления диапазонов поддерживаются только числовые типы значений (*is greater than or equals*, *is less than or equals*, *in range*). |
| *Значение* | Входящее значение (может содержать диапазон или регулярное выражение, в зависимости от типа сопоставления). |
| *Сопоставлено с* | Строковое представление (до 64 символов) для входящего значения. |

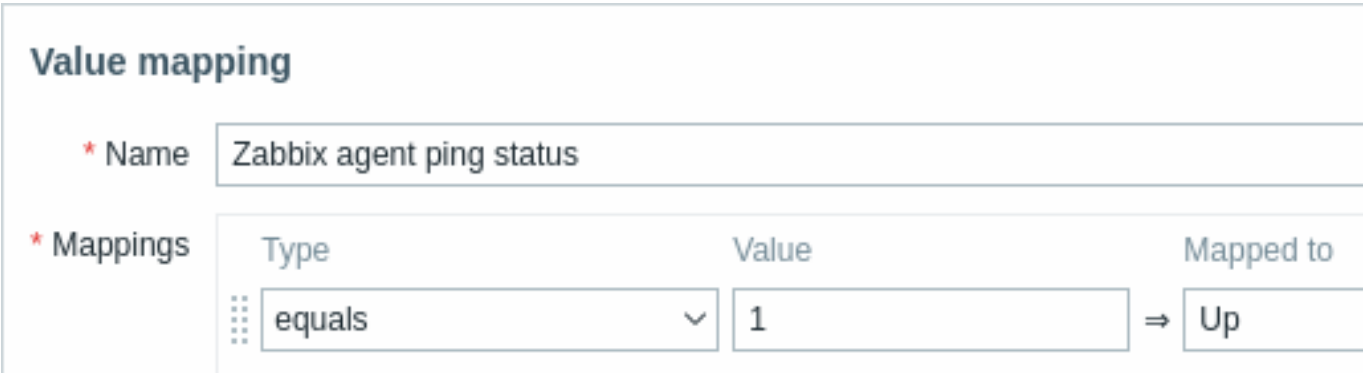
Все обязательные поля ввода отмечены красной звездочкой.

При просмотре сопоставления значений в списке видны только первые три сопоставления, а три точки указывают на наличие дополнительных сопоставлений.



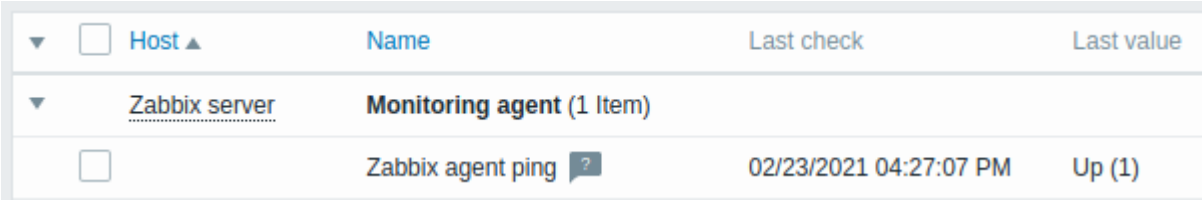

Пример преобразования значений

Один из предустановленных агентских элементов данных *Zabbix agent ping* для отображения своих значений использует существующее преобразование значений уровня шаблона, именуемое «Zabbix agent ping status».




В диалоге настройки элемента данных вы можете увидеть ссылку на это преобразование значений в поле *Преобразование значений (Value mapping)*:

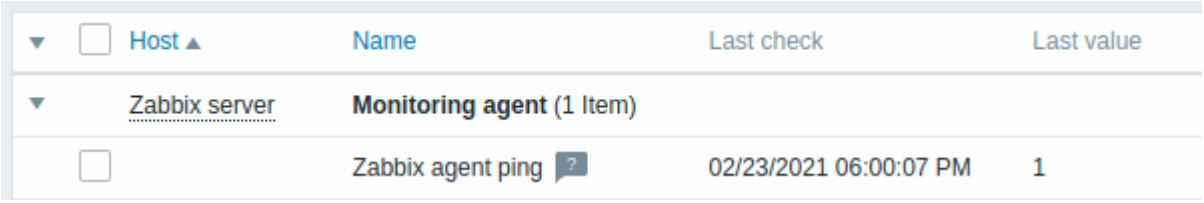
В разделе *Мониторинг* → *Последние данные* (*Monitoring* → *Latest data*) это преобразование используется, чтобы отобразить «Up» (вместе с исходным значением в скобках).



| Host | Name | Last check | Last value |
|---|---|---|---|
| Zabbix server | **Monitoring agent** (1 Item) | | |
| | Zabbix agent ping | 02/23/2021 04:27:07 PM | Up (1) |

**Note:**
В разделе *Последние данные* отображаемые значения сокращаются до 20 символов. Если используется преобразование значений, то усечение не применяется к преобразованному значению, оно применяется только к исходному значению (отображается в круглых скобках).

Без предустановленного преобразования значений вы могли бы увидеть только «1», что может быть неочевидным для понимания.



| Host | Name | Last check | Last value |
|---|---|---|---|
| Zabbix server | **Monitoring agent** (1 Item) | | |
| | Zabbix agent ping | 02/23/2021 06:00:07 PM | 1 |

**9. Очередь**

Обзор

Очередь отображает элементы данных, ожидающих обновления. Очередь — просто **логическое** представление данных. Это не очередь IPC и не какой-то другой механизм очереди в Zabbix.

Элементы данных, наблюдаемые через прокси, также включены в очередь — они будут считаться задерживающимися за период обновления данных истории с прокси.

В очереди отображаются только те элементы данных, которые имеют запланированное время обновления. Это означает, что следующие типы элементов данных исключены из очереди:

- активные элементы данных Zabbix агента log, logrt и журнал событий
- элементы данных SNMP трапов
- траппер элементы данных
- элементы данных веб-мониторинга
- зависимые элементы данных

Статистика, представленная в очереди, является хорошим показателем производительности сервера Zabbix.

Очередь запрашивается напрямую с Zabbix сервера с использованием протокола JSON. Информация доступна, только если Zabbix сервер запущен.

**Attention:**
Элементы данных не учитываются в очереди, если интерфейс элемента данных становится недоступным из-за проблем с подключением или агент не работает должным образом.

Чтение очереди

Для чтения очереди, перейдите в *Администрирование → Очередь → Обзор очереди (Administration → Queue → Queue overview)*.

Queue overview

| Items | 5 seconds | 10 seconds | 30 seconds | 1 minute | 5 minutes | More than 10 minutes |
|---|---|---|---|---|---|---|
| Zabbix agent | 1 | 11 | 1 | 0 | 0 | 0 |
| Zabbix agent (active) | 0 | 0 | 0 | 0 | 0 | 0 |
| Simple check | 0 | 0 | 0 | 0 | 0 | 0 |
| SNMPv1 agent | 0 | 0 | 0 | 0 | 0 | 0 |
| SNMPv2 agent | 0 | 0 | 0 | 0 | 0 | 0 |
| SNMPv3 agent | 0 | 0 | 0 | 0 | 0 | 0 |
| Zabbix internal | 0 | 0 | 0 | 0 | 0 | 0 |
| Zabbix aggregate | 0 | 0 | 0 | 0 | 0 | 0 |
| External check | 0 | 0 | 0 | 0 | 0 | 0 |
| Database monitor | 0 | 0 | 0 | 0 | 0 | 0 |
| HTTP agent | 0 | 0 | 0 | 0 | 0 | 0 |

Эта картинка в целом нормальная, поэтому мы можем считать, что сервер работает хорошо.

Очередь показывает несколько элементов данных, ожидающих до 30 секунд. Было бы здорово знать, что это за элементы данных.

Чтобы это сделать, выберите *Детали очереди* (*Queue details*) в верхнем выпадающем меню. Теперь вы можете видеть список этих задерживающихся элементов данных.

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 |

При наличии таких деталей имеется возможность выяснить, почему эти элементы данных могут задерживаться.

При задержке одного или двух элементов данных, возможно, нет причин для беспокойства. Они могут обновиться в любую секунду. Однако, если вы видите большое количество элементов данных, ожидающих обновления длительное время, может иметься более серьёзная проблема.

**Смотрите также:** Выравнивание часовых поясов при использовании интервалов по расписанию

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 |

Элемент данных очереди

Можно использовать специальный внутренний элемент данных **zabbix[queue,<от>,<до>]** для наблюдения за состоянием очереди Zabbix. Он будет возвращать количество элементов данных, задерживающихся за указанный период времени. Для получения более подробной информации смотрите Внутренние элементы данных.

### 10. Кэш значений

Обзор

Чтобы делать вычисления выражений триггеров, вычисляемых элементов данных и некоторых макросов намного быстрее, в Zabbix сервере поддерживается опция кэша значений.

Этот находящийся в оперативной памяти кэш используется для доступа к данным истории, вместо выполнения прямых SQL-запросов к базе данных. Если данные истории не присутствуют в кэше, недостающие значения запрашиваются из базы данных, а кэш обновляется соответственно.

Значения элементов данных остаются в кэше значений до тех пор, пока не произойдёт одно из следующих событий:

- элемент данных будет удалён (кэшированные значения удаляются после следующей синхронизации конфигурации);
- значение элемента данных выйдет за пределы диапазона времени или количества, указанного в выражении триггера/вычисляемого элемента данных (кэшированное значение удаляется при получении нового значения);
- диапазон времени или количества, указанный в выражении триггера/вычисляемого элемента данных, изменён таким образом, что для расчёта требуется меньше данных (ненужные кэшированные значения удаляются через 24 часа).

**Note:**
Состояние кэша значений можно наблюдать с помощью опции управления работой сервера `diaginfo` (или `diaginfo=valuecache`) и проверкой раздела кэша значений на предмет диагностической информации. Это может быть полезно для определения неправильно настроенных триггеров или вычисляемых элементов данных.

Чтобы включить функцию кэширования значений, файлом конфигурации сервера Zabbix поддерживается необязательный параметр параметр **ValueCacheSize**.

Для наблюдения за кэшем значений поддерживаются два внутренних элемента данных: **zabbix[vcache,buffer,<режим>]** и **zabbix[vcache,cache,<параметр>]**. Для получения более подробных сведений смотрите внутренние элементы данных.

### 11. Выполнить сейчас

Обзор

Опция *Выполнить сейчас (Execute now)* позволяет выполнить пассивную проверку немедленно.

Сбор значений элементов данных в Zabbix'е основывается на заданных интервалах обновления. Некоторые элементы данных (например, правила низкоуровневого обнаружения) имеют довольно продолжительные интервалы, а вам может понадобиться извлечь новое значение незамедлительно — например, чтобы заметить изменения в обнаруживаемых ресурсах.

Опция *Выполнить сейчас (Execute now)* поддерживается для следующих типов элементов данных:

- Zabbix агент (пассивный)
- Простая проверка
- SNMP агент (v1/v2/v3)
- Zabbix внутренний
- Внешняя проверка
- Монитор баз данных
- HTTP агент
- IPMI агент
- SSH агент
- TELNET агент
- JMX агент
- Вычисляемый
- Зависимый элемент данных
- Скрипт
- Браузер

Если элемент данных относится к типу «Зависимый элемент данных (Dependent item)», то его основной элемент данных также должен быть одного из вышеуказанных типов.

Настройка

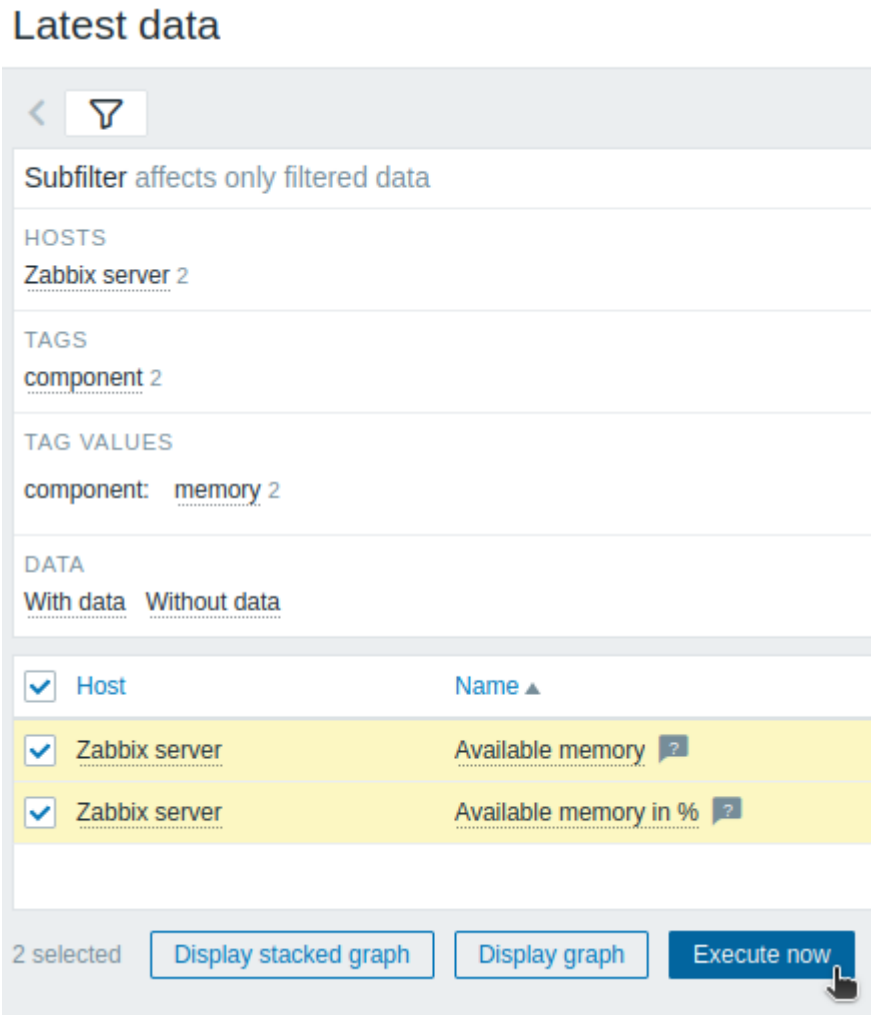
Пассивную проверку можно выполнить из:

- *Мониторинг → Последние данные (Monitoring → Latest data)*
- *Сбор данных (Data collection) → Узлы сети (Hosts) → Элементы данных (Items)* или *Правила обнаружения (Discovery)*
- Меню элементов данных
- При редактировании элементов данных или правил низкоуровневого обнаружения (в их диалогах настроек)

**Attention:**
Чтобы быть выполненной, проверка должна присутствовать в кэше конфигурации; для получения более подробных сведений смотрите CacheUpdateFrequency. Перед выполнением проверки кэш конфигурации **не обновляется**, поэтому совсем недавние изменения конфигурации элементов данных/правил обнаружения не подхватятся. Чтобы проверить элемент данных или правило обнаружения, создаваемые либо созданные только что, можно использовать вместо этого опцию Тест (Test).

Например, для выполнения пассивной проверки в *Мониторинг → Последние данные (Monitoring → Latest data)*:

1. Выберите элементы данных; можно выполнить проверку по нескольким элементам данных сразу.
2. Нажмите *Выполнить сейчас (Execute now)*.




**Note:**
В разделе *Последние данные (Latest data)* опция *Выполнить сейчас (Execute now)* доступна пользователям для узлов сети, на которые они имеют права на *чтение/запись*, либо права только на *чтение* и опцию действия *Вызов "Выполнить сейчас" над узлами сети только для чтения* (*Invoke "Execute now" on read-only hosts*), активированную для их роли.

### 12. Ограничение проверок агента

Обзор

Можно управлять тем, какие ключи элементов данных Zabbix агенту или Zabbix агенту 2 резрешено или запрещено использовать при выполнении проверок элементов данных, удалённых команд или скриптов.

Чтобы это сделать, используйте эти параметры конфигурации агента для для задания разрешающих/запрещающих правил:

- AllowKey=< >
- DenyKey=< >

< > должен содержать один ключ элемента данных и поддерживает подстановочный знак (*). Подстановочный знак соответствует любому количеству любых символов в данной позиции, и может использоваться для сопоставления как ключа элемента данных, так и параметров (например, vfs.file.*[*]).

**Attention:**
Для улучшения безопасности рекомендуется использовать точные ключи элементов данных, а не подстановочные символы. Более подробную информацию смотрите в разделе Обеспечение безопасности разрешающих/запрещающих правил.

Важные правила

- Все элементы данных system.run по умолчанию запрещены (даже когда DenyKey является пустым), как если бы последним правилом было добавлено DenyKey=system.run[*]. Благодаря этому, вы можете разрешить конкретные элементы данных system.run без явного запрета остальных элементов данных system.run.
- Элемент данных, указанный в AllowKey, также должен быть указан в DenyKey (за исключением элементов данных system.run); в противном случае Zabbix агент **не запустится**.
- По возможности, используйте AllowKey для разрешения только нужных элементов данных и запретите всё остальное. Некоторые ключи могут быть злонамеренно использованы для чтения непредназначенных для этого файлов через так называемые обходные пути (path traversal) (например, vfs.file.contents["../../../../etc/passwd"]) а новые версии Zabbix агента могут вводить ключи, не покрываемые вашими правилами DenyKey.
- Настройки AllowKey и DenyKey не влияют на параметры агента HostnameItem, HostMetadataItem и HostInterfaceItem.
- Запрещённые элементы данных становятся неподдерживаемыми без указания причины или какого-либо сообщения об ошибке; например:
  - Zabbix агент с параметром командной строки --print (-p) не будет показывать запрещённые ключи элементов данных.
  - Zabbix агент с параметром командной строки --test (-t) вернёт состояние «Unsupported item key.» (Неподдерживаемый ключ элемента данных) для запрещённых ключей элементов данных.
  - Когда журналирование активировано (LogRemoteCommands=1), Zabbix агент не будет записывать в журнал агента отклонённые удалённые команды.

Порядок правил Allow/Deny

Можно указать неограниченное количество правил AllowKey или DenyKey, но их порядок имеет значение.

- Правила проверяются по очереди, сверху вниз.
- Как только ключ элемента данных соответствует правилу, элемент данных разрешается или запрещается; а проверка правил прекращается.

Например, при проверке ключа vfs.file.contents[/etc/passwd] правила обрабатываются следующим образом:

```
AllowKey=vfs.file.contents[/tmp/app.log]    #                          ,                    .
AllowKey=vfs.file.contents[/etc/passwd]     #                       ;
DenyKey=vfs.file.*[*]                       #              ,                    .
```

Следующий порядок правил запретит проверку элемента данных:

```
DenyKey=vfs.file.*[*]                       #                       ;
AllowKey=vfs.file.contents[/etc/passwd]     #              ,                    .
AllowKey=vfs.file.contents[/tmp/app.log]    #              ,                    .
```

Примеры

Следующие примеры демонстрируют распространенные шаблоны конфигурации для AllowKey и DenyKey.

Разрешение конкретных проверок и команд

Разрешить проверку только двух элементов данных vfs.file и двух команд system.run:

```
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.*[*]
```

**Note:**
Указание DenyKey=system.run[*] не является необходимым, поскольку все остальные команды system.run по умолчанию запрещены.

Разрешение скриптов

Разрешение Zabbix агенту выполнять скрипты на узле сети всеми доступными способами:

- Глобальные скрипты, которые могут быть выполнены через веб-интерфейс или посредством API (этот метод всегда использует ключ system.run[myscript.sh])
- Удалённые команды из операций действия (этот метод всегда использует ключ system.run[myscript.sh,nowait])
- Элементы данных Zabbix агента system.run со скриптом, например:
  - system.run[myscript.sh]
  - system.run[myscript.sh,wait]
  - system.run[myscript.sh,nowait]

```
AllowKey=system.run[myscript.sh,*]
```

Чтобы управлять параметром wait/nowait, вам нужно установить другое правило. Например, вы можете разрешить только элементы данных system.run[myscript.sh,wait], исключая таким образом другие способы:

```
AllowKey=system.run[myscript.sh,wait]
```

Обеспечение безопасности разрешающих/запрещающих правил

В этом примере показано, как обезопасить чрезмерно разрешительные правила AllowKey или DenyKey.

Consider the following rules:

```
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat*"]
DenyKey=vfs.file.*
DenyKey=system.cpu.load[*]
```

**Note:**
В Windows, вам нужно экранировать пробелы в путях с помощью символа «карет» (^).

Эти правила содержат подстановочный символ (*), который может быть использован неверно:

- Скрипт test.bat может выполняться с любыми аргументами, включая те, которые для этого не предназначены.
- Шаблон vfs.file.* соответствует только ключам элементов данных без параметров; однако, для всех элементы данных vfs.file требуются параметры.
- Шаблон system.cpu.load[*] соответствует только ключам элементов данных с параметрами; однако, для элементов данных system.cpu.load параметры не требуются.

Чтобы сделать эти правила безопасными, явно разрешите выполнение test.bat только с конкретными аргументами, а также запретите правильные шаблоны ключей элементов данных; например:

```
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[*]
```

Можно протестировать эти правила, выполняя следующие команды, которые будут возвращать 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]
```

Примеры шаблонов

Следующая таблица показывает, как работает соответствие шаблонов ключей элеменов данных:

- Ключ соответствует шаблону, только если он удовлетворяет **всем** условиям в столбце *Соответствует*.
- Параметры ключа элемента данных должны быть полность заключены в квадратные скобки (например, `vfs.file.contents[*` и `vfs.file.contents*utf8]` являются некорректными шаблонами).

| Шаблон | Соответствует | Примеры |
|---|---|---|
| `*` | Любой ключ с параметрами или без них. | |
| `vfs.file.*` | Ключ начинается с `vfs.file.`<br>Без параметров | Соответствует:<br>`vfs.file.size`<br>`vfs.file.contents`<br><br>Не соответствует:<br>`vfs.file.contents[]`<br>`vfs.file.size[/var/log/app.log]` |
| `vfs.*.contents` | Ключ начинается с `vfs.`<br>Ключ заканчивается на `.contents`<br>Без параметров | Соответствует:<br>`vfs..contents`<br>`vfs.mount.point.file.contents`<br><br>Не соответствует:<br>`vfs.contents`<br>`vfs.file.contents[]` |
| `vfs.file.*[*]` | Ключ начинается с `vfs.file.`<br>С любыми (в том числе пустыми) параметрами | Соответствует:<br>`vfs.file.get.custom[]`<br>`vfs.file.size[/var/log/app.log, utf8]`<br><br>Не соответствует:<br>`vfs.file.get.custom` |
| `vfs.file.contents` | Ключ — строка `vfs.file.contents`<br>Без параметров | Соответствует:<br>`vfs.file.contents`<br><br>Не соответствует:<br>`vfs.file.contents[/etc/passwd]` |
| `vfs.file.contents[]` | Ключ — строка `vfs.file.contents[]`<br>Пустые параметры | Соответствует:<br>`vfs.file.contents[]`<br><br>Не соответствует:<br>`vfs.file.contents` |
| `vfs.file.contents[*]` | Ключ — строка `vfs.file.contents`<br>С любыми (в том числе пустыми) параметрами | Соответствует:<br>`vfs.file.contents[/path/to/file]`<br><br>Не соответствует:<br>`vfs.file.contents` |
| `vfs.file.contents[/etc/passwd,*]` | Ключ — строка `vfs.file.contents`<br>Первый параметр — строка `/etc/passwd`<br>Любой второй параметр (в том числе пустой) | Соответствует:<br>`vfs.file.contents[/etc/passwd,]`<br>`vfs.file.contents[/etc/passwd,utf8]`<br><br>Не соответствует:<br>`vfs.file.contents[]`<br>`vfs.file.contents[/etc/passwd]` |
| `vfs.file.contents[*passwd*]` | Ключ — строка `vfs.file.contents`<br>Первый параметр содержит строку `passwd`<br>Второго параметра нет | Соответствует:<br>`vfs.file.contents[/etc/passwd]`<br><br>Не соответствует:<br>`vfs.file.contents[/etc/passwd,]`<br>`vfs.file.contents[/etc/passwd,utf8]` |
| `vfs.file.contents[*passwd*,*]` | Ключ — строка `vfs.file.contents`<br>Первый параметр содержит строку `passwd`<br>Любой второй параметр (в том числе пустой) | Соответствует:<br>`vfs.file.contents[/etc/passwd,]`<br>`vfs.file.contents[/etc/passwd,utf8]`<br><br>Не соответствует:<br>`vfs.file.contents[/etc/passwd]`<br>`vfs.file.contents[/tmp/test]` |

| Шаблон | Соответствует | Примеры |
|---|---|---|
| vfs.file.contents[/etc/passwd,utf8] | Ключ vfs.file.contents<br>Первый параметр — строка /etc/passwd<br>Второй параметр — строка utf8 | Соответствует:<br>vfs.file.contents[/etc/passwd,utf8]<br><br>Не соответствует:<br>vfs.file.contents[/etc/passwd,]<br>vfs.file.contents[/etc/passwd,utf16] |

### 3. Триггеры

Обзор

Триггеры — это логические выражения, которые «оценивают» данные, собранные элементами данных, и отражают текущее состояние системы.

Тогда как элементы данных используются для сбора системных данных, крайне непрактично наблюдать за этими данными всё время в ожидании выполнения условия, которое приведёт к оповещению или которое заслуживает внимания. Работа по «оценке» данных может быть оставлена на выражения триггеров.

Выражения триггеров позволяют задать порог, при котором состояние данных «приемлемое». Таким образом, если входящие данные превышают приемлемое состояние, триггер «срабатывает» — или, другими словами, меняет состояние на ПРОБЛЕМА.

Триггер может принимать следующие состояния:

| Состояние | Описание |
|---|---|
| OK | Нормальное состояние триггера. |
| Проблема | Что-то случилось. Например, загрузка процессора стала слишком высокой. |
| Неизвестно | Значение триггера не может быть вычислено. Смотрите Неизвестное состояние. |

В простом триггере мы можем захотеть установить порог по среднему за пять минут значению каких-либо данных, например, загрузки CPU. Такое можно реализовать, определяя выражение триггера, где:

- к значению, полученному в ключе элемента данных, применяется функция «avg»
- используется период оценки в пять минут
- порог задаётся равным «2»

```
avg(/host/key,5m)>2
```

Этот триггер «срабатывает» (переходит в состояние ПРОБЛЕМА), если среднее за пять минут *превышает* 2.

В более сложном триггере выражение может включать в себя **комбинацию** из нескольких функций и нескольких порогов. Смотрите также: Выражение триггера.

Нельзя создавать триггеры для элементов данных, возвращающих значения бинарного типа данных.

**Note:**
После включения триггера (изменение его состояния конфигурации с *Деактивирован* на *Активирован*) выражение триггера будет вычислено, как только элемент данных из выражения получит своё значение или когда подойдёт время обработки функции даты и времени и/или функции nodata().

Большинство функций триггеров вычисляются на основании данных истории значения элемента данных, в то же время некоторые функции триггеров для долгосрочной аналитики (например, **trendavg()**, **trendcount()** и другие) используют данные динамики изменений.

Время вычисления

Триггер пересчитывается каждый раз, когда Zabbix сервер получает новое значение, которое является частью выражения. Когда получено новое значение, каждая функция, которая включена в выражение, пересчитывается (не только та, что получила новое значение).

Кроме того, триггер пересчитывается каждый раз, когда получено новое значение, **и** каждые 30 секунд, если в выражении триггера используются функции даты и времени и/или функция nodata().

Функции даты и времени и/или функция nodata() пересчитываются каждые 30 секунд процессами синхронизации истории Zabbix (history syncer).

Триггеры, которые ссылаются **только** на функции динамики изменений, вычисляются один раз за наименьший период в выражении. Смотрите также функции динамики изменений.

Период вычисления

Период вычисления используется в функциях, которые ссылаются на историю элемента данных. Это позволяет указать интересующий нас интервал. Интервал можно указать в виде периода времени (30s, 10m, 1h) или в виде диапазонов значений (#5 — для пяти последних значений).

Период вычисления измеряется до «сейчас», где «сейчас» — это время последнего пересчёта триггера (смотрите Время вычисления выше); «сейчас» — это не «текущее» время сервера.

Период вычисления указывает либо:

- Учитывать все значения между «сейчас - период времени» и «сейчас» (или, при наличии сдвига времени, между «сейчас - сдвиг_времени - период времени» и «сейчас - сдвиг_времени»)
- Учитывать не более чем количество значений из прошлого, до «сейчас»
  - Если имеется 0 доступных значений за этот период времени или указанное количество — тогда триггер или вычисляемый элемент данных, который использует такую функцию, становится неподдерживаемым

Обратите внимание, что:

- Если в триггере используется одна функция (которая ссылается на данные истории), «сейчас» всегда последнее полученное значение. Например, если последнее значение получено час назад, период вычисления будет рассматриваться до последнего значения часом ранее.
- Новый триггер вычисляется сразу, как только получено первое значение (функции истории); он будет вычисляться в течение 30 секунд при наличии функций даты и времени и функции nodata(). Таким образом, триггер будет вычислен, даже если заданный период оценки (например, один час) еще не прошёл с момента создания триггера. Триггер также будет вычислен после первого значения — даже если диапазон оценки был задан, например, последними десятью значениями.

Неизвестное состояние

Имеется вероятность, что в выражении появится неизвестный операнд, если:

- использован неподдерживаемый элемент данных
- вычисление функции по поддерживаемому элементу данных приводит к ошибке

В этом случае триггер обычно вычисляется как «неизвестно» (хотя имеются некоторые исключения). Для получения более подробных сведений смотрите Выражения с неизвестными операндами.

О неизвестных триггерах можно получить оповещение.

### 1. Настройка триггера

Обзор

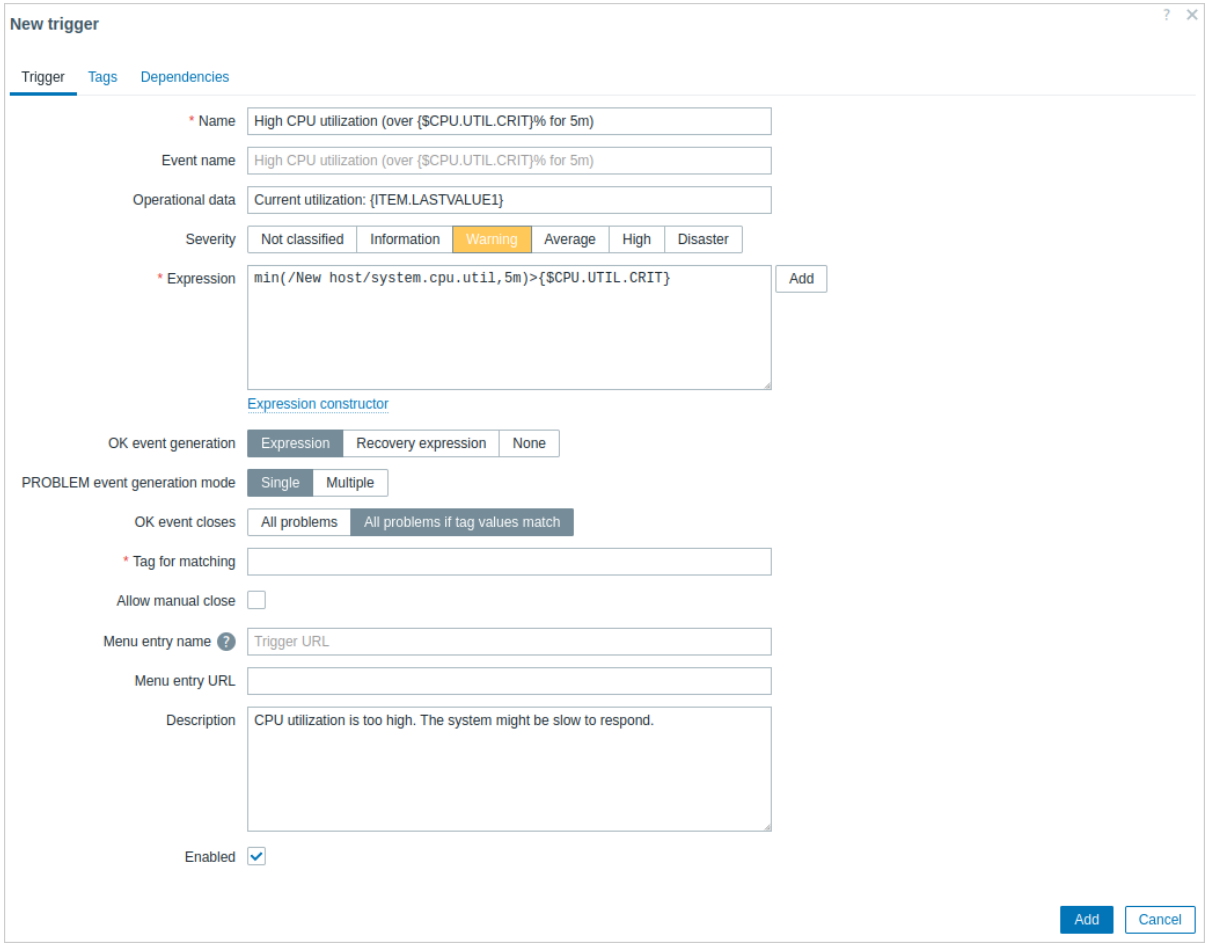
Чтобы настроить триггер, выполните следующие действия:

- Перейдите в: *Сбор данных* > *Узлы сети*
- Нажмите *Триггеры* в строке узла сети
- Нажмите *Создать триггер* справа (или на имени триггера, чтобы изменить существующий триггер)
- Введите параметры триггера в форме

См. также общую информацию о триггерах и времени их вычисления.

Конфигурация

Вкладка **Trigger** содержит все основные атрибуты триггера.

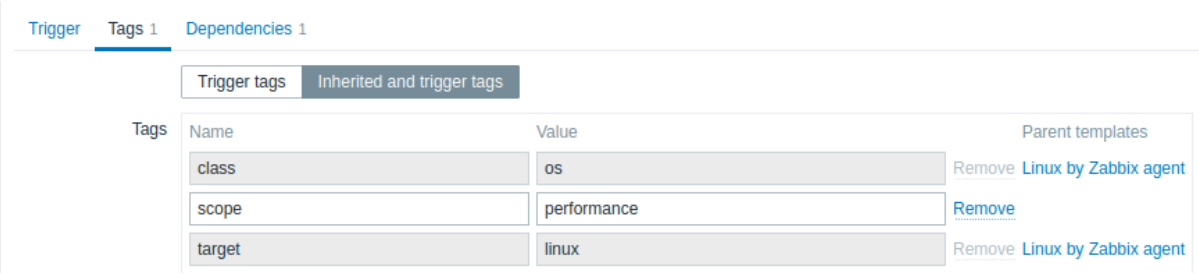



Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Name* | Имя триггера.<br>Поддерживаются макросы: {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.*} и пользовательские макросы {$MACRO}.<br>Макросы **$1, $2...$9** можно использовать для ссылки на первую, вторую... девятую константу выражения.<br>*Примечание*: макросы $1-$9 будут корректно разрешаться, если они ссылаются на константы в относительно простых, прямолинейных выражениях. Например, имя "Processor load above $1 on {HOST.NAME}" автоматически изменится на "Processor load above 5 on New host", если выражение last(/New host/system.cpu.load[percpu,avg1])>5. |
| *Event name* | Если задано, это имя будет использоваться для создания имени события проблемы. По умолчанию имя события совпадает с именем триггера.<br>Имя события может использоваться для формирования информативных оповещений, содержащих данные о проблеме (см. пример).<br>Поддерживается тот же набор макросов, что и в имени триггера, а также макросы выражений {TIME}, {TIMESTAMP} и {?EXPRESSION}. |
| *Operational data* | Операционные данные позволяют задавать произвольные строки вместе с макросами. Макросы будут динамически разрешаться в данные реального времени в *Monitoring* > *Problems*. Если макросы в имени триггера (см. выше) разрешаются в свои значения в момент возникновения проблемы и становятся основой статического имени проблемы, то макросы в операционных данных сохраняют возможность динамически отображать самые актуальные сведения. Если на уровне триггера операционные данные не настроены, будут отображаться последние значения всех элементов данных из выражения.<br>Поддерживается тот же набор макросов, что и в имени триггера. |
| *Severity* | Установите требуемую серьезность триггера, нажимая кнопки. |

| Parameter | Description |
|---|---|
| *Expression* | Логическое выражение, используемое для определения условий проблемы.<br>Поддерживаются суффиксы времени и суффиксы размера памяти.<br>Проблема создается после выполнения всех условий, включенных в выражение, то есть когда выражение принимает значение TRUE. Проблема будет устранена, как только выражение примет значение FALSE, если только дополнительные условия восстановления не указаны в *Recovery expression*. |
| *OK event generation* | Параметры генерации событий OK:<br>**Expression** - события OK генерируются на основе того же выражения, что и события проблемы;<br>**Recovery expression** - события OK генерируются, если выражение проблемы принимает значение FALSE, а выражение восстановления - значение TRUE;<br>**None** - в этом случае триггер никогда не вернется в состояние OK самостоятельно. |
| *Recovery expression* | Логическое выражение (необязательное), определяющее дополнительные условия, которые должны быть выполнены перед устранением проблемы, после того как исходное выражение проблемы уже было оценено как FALSE.<br>Выражение восстановления полезно для гистерезиса триггера. Невозможно устранить проблему только по выражению восстановления, если выражение проблемы по-прежнему TRUE.<br>Это поле доступно только если для *OK event generation* выбрано значение 'Recovery expression'. |
| *PROBLEM event generation mode* | Режим генерации событий проблемы:<br>**Single** - одно событие генерируется, когда триггер впервые переходит в состояние 'Problem';<br>**Multiple** - событие генерируется при *каждой* оценке триггера как 'Problem'. |
| *OK event closes* | Выберите, закрывает ли событие OK:<br>**All problems** - все проблемы этого триггера;<br>**All problems if tag values match** - только те проблемы триггера, у которых совпадают значения тегов события. |
| *Tag for matching* | Введите имя тега события, которое будет использоваться для корреляции событий.<br>Это поле отображается, если для свойства *OK event closes* выбрано значение 'All problems if tag values match', и в этом случае является обязательным. |
| *Allow manual close* | Установите этот флажок, чтобы разрешить ручное закрытие событий проблемы, создаваемых этим триггером. Ручное закрытие возможно при подтверждении событий проблемы. |
| *Menu entry name* | Если поле не пустое, введенное здесь имя (до 64 символов) используется в нескольких разделах веб-интерфейса как подпись для URL триггера, указанного в параметре *Menu entry URL*. Если поле пустое, используется имя по умолчанию *Trigger URL*.<br>Поддерживается тот же набор макросов, что и в имени триггера, а также {EVENT.ID}, {HOST.ID} и {TRIGGER.ID}. |
| *Menu entry URL* | Если поле не пустое, введенный здесь URL (до 2048 символов) доступен как ссылка в меню событий в нескольких разделах веб-интерфейса, например при нажатии на имя проблемы в *Monitoring* > *Problems* или в виджете панели *Problems*.<br>Поддерживается тот же набор макросов, что и в имени триггера, а также {EVENT.ID}, {HOST.ID} и {TRIGGER.ID}. Примечание: пользовательские макросы со скрытыми значениями не будут разрешаться в URL. |
| *Description* | Текстовое поле, используемое для предоставления дополнительной информации об этом триггере. Может содержать инструкции по устранению конкретной проблемы, контактные данные ответственных сотрудников и т. д.<br>Поддерживается тот же набор макросов, что и в имени триггера. |
| *Enabled* | Снимите этот флажок, если требуется отключить триггер.<br>Проблемы отключенного триггера больше не отображаются в веб-интерфейсе, но не удаляются. |

Вкладка **Tags** позволяет определить теги на уровне триггера. Все проблемы этого триггера будут помечены значениями, указанными здесь.




Кроме того, параметр *Inherited and trigger tags* позволяет просматривать теги, определенные на уровне шаблона, если триггер получен из этого шаблона. Если существует несколько шаблонов с одинаковым тегом, эти теги отображаются один раз, а имена шаблонов разделяются запятыми. Триггер не "наследует" и не отображает теги уровня узла сети.

| Parameter | Description |
|---|---|
| *Name/Value* | Задайте пользовательские теги для пометки событий триггера.<br>Теги представляют собой пару из имени тега и значения. Можно использовать только имя или имя вместе со значением. У триггера может быть несколько тегов с одинаковым именем, но разными значениями.<br>В тегах событий поддерживаются пользовательские макросы, пользовательские макросы с контекстом, макросы низкоуровневого обнаружения и макрофункции functions с `{{ITEM.VALUE}}`, `{{ITEM.LASTVALUE}}`.<br>Макросы низкоуровневого обнаружения можно использовать внутри контекста макроса. Макрос {TRIGGER.ID} поддерживается в значениях тегов триггера. Это может быть полезно для идентификации триггеров, созданных из прототипов триггеров, и, например, для подавления проблем от этих триггеров во время обслуживания.<br>Если общая длина развернутого значения превышает 255, оно будет обрезано до 255 символов.<br>См. все макросы, поддерживаемые для тегов событий.<br>Теги событий можно использовать для корреляции событий, в условиях действий, а также они будут видны в *Monitoring* > *Problems* или в виджете *Problems*. |

Вкладка **Dependencies** содержит все зависимости триггера.

Нажмите *Add*, чтобы добавить новую зависимость.

**Note:**
Вы также можете настроить триггер, открыв существующий, нажав кнопку *Clone* и затем сохранив его под другим именем.

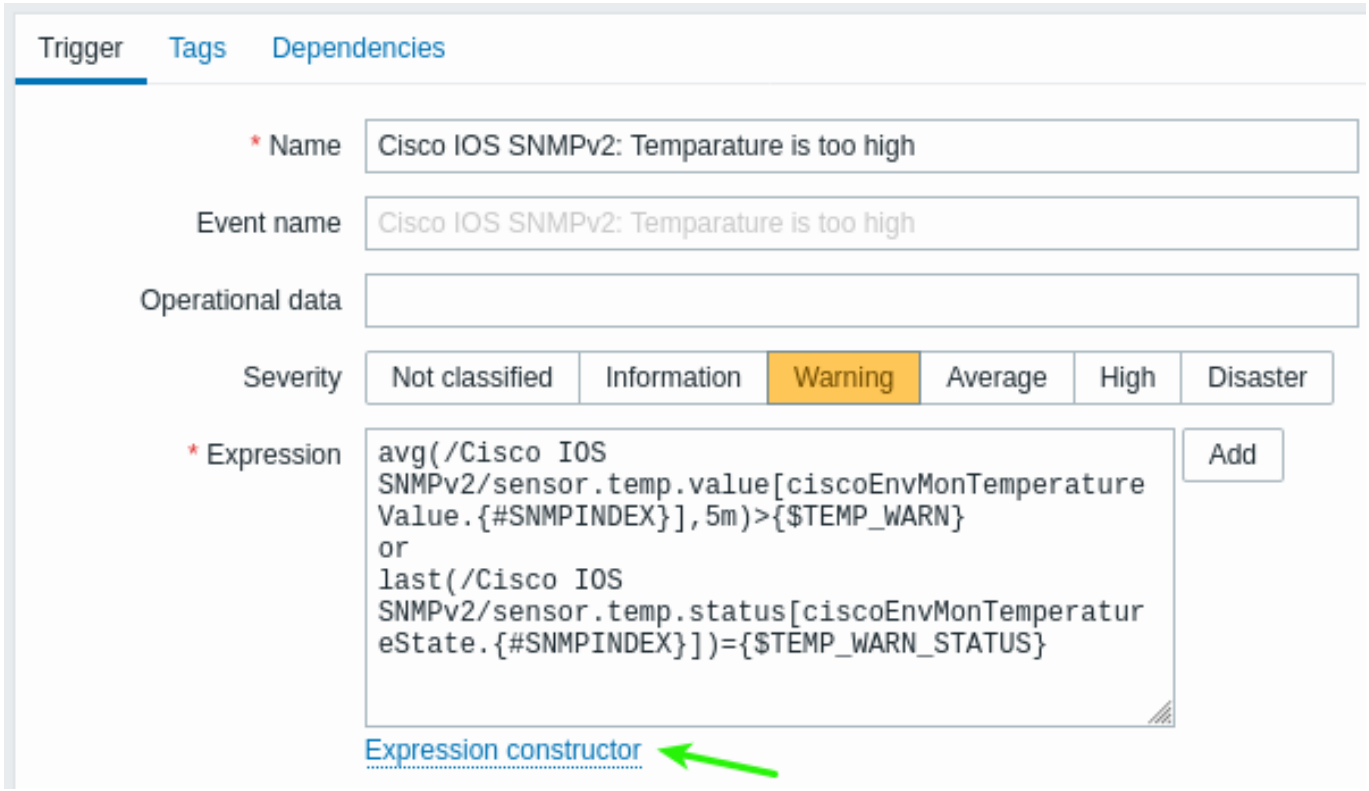
Проверка выражений

Можно проверить настроенное выражение триггера, чтобы увидеть, каким будет результат выражения в зависимости от полученного значения.

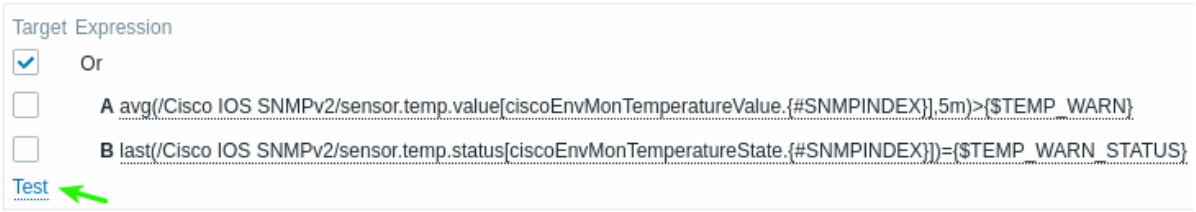
В качестве примера взято следующее выражение из официального шаблона:

```
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}
```

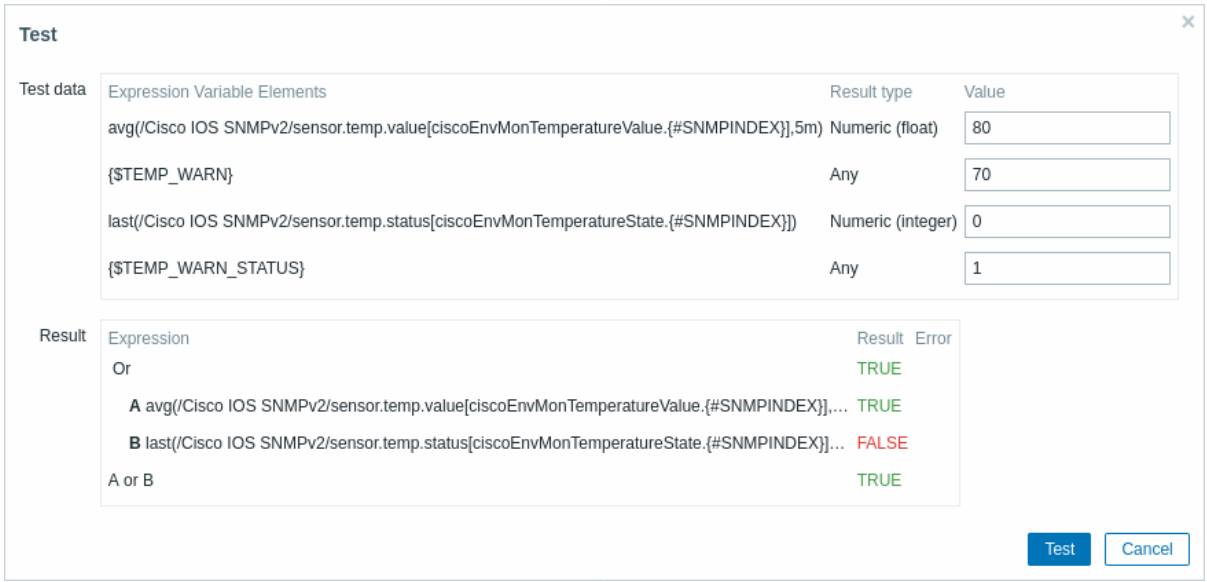
Чтобы проверить выражение, нажмите *Конструктор выражений* под полем выражения.




В конструкторе выражений перечислены все отдельные выражения. Чтобы открыть окно проверки, нажмите *Тест* под списком выражений.




В окне проверки можно ввести примерные значения (’80’, ’70’, ’0’, ’1’ в этом примере), а затем посмотреть результат выражения, нажав кнопку *Тест*.




Можно увидеть результат как отдельных выражений, так и всего выражения.

”TRUE” означает, что указанное выражение верно. В данном случае A значение ”80” больше указанного значения {$TEMP_WARN}, ”70” в этом примере. Как и ожидалось, отображается результат ”TRUE”.

”FALSE” означает, что указанное выражение неверно. В данном случае B значение {$TEMP_WARN_STATUS} ”1” должно быть равно указанному значению, ”0” в этом примере. Как и ожидалось, отображается результат ”FALSE”.

Выбранный тип выражения — "OR". Если хотя бы одно из указанных условий (A или B в данном случае) имеет значение TRUE, итоговый результат также будет TRUE. Это означает, что текущее значение превышает предупреждающее значение и возникла проблема.

**2. Выражение триггера**

Обзор

Используемые в триггерах выражения являются очень гибкими. Вы можете использовать их для создания сложных логических проверок, касающихся отслеживаемой статистики.

Простое выражение использует **функцию**, которая применяется к элементу данных с некоторыми параметрами. Функция возвращает результат, который сравнивается с порогом, используя оператор и константу.

Синтаксис простого полезного выражения: (/ / , )< >< >.

Например:

```
min(/Zabbix server/net.if.in[eth0,bytes],5m)>100K
```

сработает, если количество полученных байт за последние пять минут всегда превышает 100 килобайт.

С функциональной точки зрения существует два типа выражений триггеров, хоть они и имеют абсолютно одинаковый синтаксис:

- выражение проблемы — задаёт условия проблемы
- выражение восстановления (опционально) — задаёт дополнительные условия решения проблемы

Когда задается только выражение проблемы, это выражение будет использоваться в качестве порога как проблемы, так и восстановления проблемы. Как только выражение проблемы вычисляется в значение ИСТИНА, возникает проблема. Как только выражение проблемы вычисляется в значение ЛОЖЬ, проблема считается решённой.

Если задаются оба выражения, как проблемы, так и дополнительного выражения восстановления, решение проблемы становится более сложным: не только выражение проблемы должно стать ЛОЖЬ, но также и выражение восстановления должно стать ИСТИНА. Такой подход полезен для создания гистерезиса и помогает избежать частых переключений триггера (из/в состояния ПРОБЛЕМЫ и обратно).

> **Note:**
> Использовать макрос {TRIGGER.VALUE} в выражении восстановления непродуктивно, поскольку это выражение вычисляется только тогда, когда триггер находится в состоянии «Проблема». Следовательно, при вычислении выражения {TRIGGER.VALUE} всегда будет разрешаться в «1» (что указывает на состояние «Проблема»).

Функции

Функции позволяют обсчитывать собранные значения (вычислять среднее, минимальное, максимальное, сумму), искать строки, ссылаться на текущее время и принимать во внимание другие факторы.

Имеется полный список поддерживаемых функций.

Обычно функции возвращают числовые значения для последующего сравнения. При возврате строк можно выполнять сравнения при помощи операторов **=** и **<>** (смотрите пример).

Параметры функции

Параметры функции позволяют указать:

- узел сети и ключ элемента данных (функции, ссылающиеся только на историю элемента данных узла сети)
- параметры, специфичные для функции
- другие выражения (недоступны для функций, ссылающихся только на историю элемента данных узла сети, см. другие выражения для примеров)

Узел сети и ключ элемента данных можно указать как `/host/key`.<br> Опускание имени узла сети в первом параметре (то есть как в `function(//key,parameter,...)`) поддерживается только в определенных контекстах:

- В формуле вычисляемых элементов данных
- В макросах выражений, которые можно использовать в:
  - поле *Имя события*
  - имени графика
  - подписи элементов карты "Host" и "Trigger" map elements

В этих контекстах также можно использовать макрос {HOST.HOST}. {HOST.HOST<1-9>} можно использовать в случае поля *Имя события* и элемента карты "Trigger", чтобы ссылаться на конкретный элемент в выражении триггера. Когда имя узла сети опущено или заменено на {HOST.HOST} в этих контекстах, ссылка указывает на первый элемент в выражении триггера или на первый элемент в графике. Вне этих поддерживаемых контекстов опускание имени узла сети в выражениях триггера приведет к ошибке. См. Пример 18 для иллюстрации использования двойной косой черты в макросах имени события.

Ссылающийся элемент данных должен находиться в поддерживаемом состоянии (за исключением функции **nodata()**, которая вычисляется и для неподдерживаемых элементов данных).

Хотя другие выражения триггера в качестве параметров функции ограничены функциями без истории в триггерах, это ограничение не применяется в вычисляемых элементах данных.

Параметры, зависящие от функции

Параметры, которые зависят от используемой функции, помещаются после ключа элемента данных и отделяются от него запятой. Смотрите полный список этих параметров по поддерживаемым функциям.

Большинство числовых функций принимают время в качестве параметра. Чтобы указать время, вы можете использовать секунды или суффиксы времени. Параметр, которому предшествует символ решётки, имеет другое значение:

| Выражение | Описание |
|---|---|
| **sum**(/узел сети/ключ,**10m)** | Сумма значений за последние 10 минут. |
| **sum**(/узел сети/ключ,**#10)** | Сумма последних десяти значений. |

В случае с функцией **last** параметры с символом решётки имеют другой смысл — они обозначают N-ое предыдущее значение, таким образом, с учётом заданных значений 30, 70, 20, 60, 50 (от самого последнего к наименее последнему значению):

- last(/ / ,#2) вернёт «70»
- last(/ / ,#5) вернёт «50»

Сдвиг времени

В качестве параметра функции поддерживается опциональный сдвиг времени со временем или количеством значений. Этот параметр позволяет ссылаться на данные из периода времени в прошлом.

Сдвиг времени начинается с now — указывает текущее время, за которым следует +N< > или -N< > — для добавления или вычитания N-го количества единиц времени.

Например, avg(/ / ,1h:now-1d) вернет среднее значение за час днём ранее.

**Attention:**
Сдвиг времени, указанный в месяцах (M) и годах (y), поддерживается только для функций динамики изменений. Остальные функции поддерживают секунды (s), минуты (m), часы (h), дни (d) и недели (w).

**Сдвиг времени с абсолютными периодами времени**

В параметре сдвига времени поддерживаются абсолютные периоды времени, например, с полуночи до полуночи для дня, понедельник-воскресенье для недели, первый — последний день месяца для месяца.

Сдвиг времени по абсолютным периодам времени начинается с now — указывает текущее время, за которым следует любое количество временных операций: /< > — определяет начало и конец единицы времени, например, с полуночи до полуночи для дня, +N< > или -N< > — для добавления или вычитания N-го количества единиц времени.

Обратите внимание, что значение сдвига времени может быть больше или равно 0, тогда как минимальный период времени равен значению 1.

| Параметр | Описание |
|---|---|
| 1d:now/d | Вчера |
| 1d:now/d+1d | Сегодня |
| 2d:now/d+1d | Последние 2 дня |
| 1w:now/w | Последняя неделя |
| 1w:now/w+1w | Текущая неделя |

Другие выражения

Параметры функций могут содержать другие выражения в соответствии со следующим синтаксисом:

```
min(min(/     /  ,1h),min(/     2/  2,1h)*10)
```

Обратите внимание, что другие выражения нельзя использовать, если функция ссылается на историю элемента данных. Например, следующий синтаксис **не** разрешён:

~~`min(/     /  ,#5*10)`~~

Операторы

Следующие операторы поддерживаются триггерами **(представлены по убыванию приоритета выполнения)**:

| Приоритет | Оператор | Определение | **Заметки по неизвестным значениям** | Принудительное преобразование операнда в дробное число [1] |
|---|---|---|---|---|
| **1** | **-** | Унарный минус | **-**Неизвестно → Неизвестно | Да |
| **2** | **not** | Логическое НЕ | **not** Неизвестно → Неизвестно | Да |
| **3** | * | Умножение | 0 * Неизвестно → Неизвестно<br>(да, Неизвестно, не 0 — чтобы не потерять Неизвестно в арифметических операциях)<br>1.2 * Неизвестно → Неизвестно | Да |
| | / | Деление | Неизвестно / 0 → ошибка<br>Неизвестно / 1.2 → Неизвестно<br>0.0 / Неизвестно → Неизвестно | Да |
| **4** | + | Арифметический плюс | 1.2 + Неизвестно → Неизвестно | Да |
| | - | Арифметический минус | 1.2 - Неизвестно → Неизвестно | Да |
| **5** | < | Меньше чем. Этот оператор можно представить в виде:<br><br>A<B ⇔ (A<B-0.000001) | 1.2 < Неизвестно → Неизвестно | Да |
| | <= | Меньше чем или равно. Этот оператор можно представить в виде:<br><br>A<=B ⇔ (A≤B+0.000001) | Неизвестно <= Неизвестно → Неизвестно | Да |
| | > | Больше чем. Этот оператор можно представить в виде:<br><br>A>B ⇔ (A>B+0.000001) | | Да |
| | >= | Больше чем или равно. Этот оператор можно представить в виде:<br><br>A>=B ⇔ (A≥B-0.000001) | | Да |

| Приоритет | Оператор | Определение | **Заметки по неизвестным значениям** | Принудительное преобразование операнда в дробное число [1] |
|---|---|---|---|---|
| **6** | **=** | Равно. Этот оператор можно представить в виде:<br><br>A=B ⇔ (A≥B-0.000001) и (A≤B+0.000001) | | Нет [1] |
| | **<>** | Не равно. Этот оператор можно представить в виде:<br><br>A<>B ⇔ (A<B-0.000001) или (A>B+0.000001) | | Нет [1] |
| **7** | **and** | Логическое И | 0 **and** Неизвестно → 0<br>1 **and** Неизвестно → Неизвестно<br>Неизвестно **and** Неизвестно → Неизвестно | Да |
| **8** | **or** | Логическое ИЛИ | 1 **or** Неизвестно → 1<br>0 **or** Неизвестно → Неизвестно<br>Неизвестно **or** Неизвестно → Неизвестно | Да |

[1] Строковый операнд по-прежнему приводится к числовому значению, если:

- другой операнд числовой
- над операндом используется оператор, отличный от **=** или **<>**

При операторах сравнения (<, <=, >, >=): если приведение завершается с ошибкой (например, для ""), результатом выражения будет Неизвестно (без сравнения строковых значений).

При операторах равенства (=, <>): если приведение завершается с ошибкой, то оба операнда сравниваются как сырые строковые значения.

Операторы **not**, **and** и **or** зависят от регистра и должны быть в нижнем регистре. Они также должны быть окружены символами пробелов или круглыми скобками.

Все операторы, кроме унарных **-** и **not**, имеют ассоциативность слева направо. Унарные **-** и **not** не ассоциативны (имеется в виду, что нужно использовать **-(-1)** и **not (not 1)** вместо **--1** и **not not 1**).

Результат вычисления:

- Операторы **<**, **<=**, **>**, **>=**, **=**, **<>** должны возвращать '1' в выражении триггера, если указанное соотношение правдиво, и '0', если оно ложно. Если по крайней мере один операнд Неизвестен, то и результат будет Неизвестно;
- **and** по известным операндам должен возвращать «1», если оба из этих операндов сравнения не равны «0»; в противном случае, будет давать «0»; для неизвестных операндов **and** вернёт «0», только если один из операндов сравнения равен «0»; в противном случае, он вернёт «Неизвестно»;
- **or** по известным операндам должен возвращать «1», если какой-либо из этих операндов сравнения не равен «0»; в противном случае, будет возвращать «0»; для неизвестных операндов **or** вернёт «1», только если один из операндов сравнения не равен «0»; в противном случае, он вернёт «Неизвестно»;
- Результат логического операнда отрицания **not** для известного операнда равен «0», если значение этого операнда сравнения не равно «0»; «1», если значение его операнда сравнения равно «0». Для неизвестных операндов **not** вернёт «Неизвестно».

Кэширование значений

Значения, которые необходимы для вычисления триггеров, кэшируются Zabbix сервером. По этой причине такое вычисление триггеров приводит к более высокой нагрузке на базу данных на некоторое время после перезапуска сервера. Кэш значений не очищается, когда значения истории элементов данных удаляются (либо вручную, либо при

помощи автоматической очистки истории), поэтому сервер будет использовать кэшированные значения, пока они не станут старше, чем периоды времени, заданные в функциях триггеров, либо пока сервер не будет перезапущен.

> **Note:**
> Если в кэше нет последних данных и в функции не определён период запроса, Zabbix по умолчанию будет запрашивать у базы данных значения истории вплоть до одной недели назад.

Примеры триггеров

Пример 1

Высокая загрузка процессора на сервере Zabbix.

```
last(/Zabbix server/system.cpu.load[all,avg1])>5
```

Используя функцию «last()», мы ссылаемся на самое последнее значение. `/Zabbix server/system.cpu.load[all,avg1]` представляет короткое имя наблюдаемого параметра. Эта строка указывает, что сервер — «Zabbix server», а наблюдаемый ключ — «system.cpu.load[all,avg1]». Наконец, >5 означает, что триггер будет переходить в состояние ПРОБЛЕМА всякий раз, когда самое последнее измерение загрузки процессора на сервере Zabbix server будет превышать 5.

Пример 2

www.example.com перегружен.

```
last(/www.example.com/system.cpu.load[all,avg1])>5 or min(/www.example.com/system.cpu.load[all,avg1],10m)>
```

Это выражение будет истинным, когда либо текущая загрузка процессора станет более 5, либо загрузка процессора больше значения 2 в течение последних 10 минут.

Пример 3

/etc/passwd был изменён.

```
last(/www.example.com/vfs.file.cksum[/etc/passwd],#1)<>last(/www.example.com/vfs.file.cksum[/etc/passwd],#
```

Это выражение будет истинным, когда предыдущее значение контрольной суммы файла /etc/passwd отличается от самого нового значения.

Аналогичные выражения могут быть полезны для мониторинга изменений в важных файлах, таких как /etc/passwd, /etc/inetd.conf, /kernel и других.

Пример 4

Кто-то в данный момент скачивает из Интернета большой файл.

Используется функция min:

```
min(/www.example.com/net.if.in[eth0,bytes],5m)>100K
```

Это выражение будет истинным, когда количество полученных байт на eth0 превышает 100 КБ за последние 5 минут.

Пример 5

Оба узла кластера SMTP серверов недоступны.

Обратите внимание на то, что в выражении используются два разных узла сети:

```
last(/smtp1.example.com/net.tcp.service[smtp])=0 and last(/smtp2.example.com/net.tcp.service[smtp])=0
```

Это выражение будет истинным, когда оба SMTP сервера недоступны — и на smtp1.example.com, и на smtp2.example.com.

Пример 6

Zabbix агент нуждается в обновлении.

Используется функция find():

```
find(/example.example.com/agent.version,,"like","beta8")=1
```

Это выражение будет истинным, когда версия Zabbix агента содержит «beta8».

Пример 7

Сервер недоступен.

```
count(/example.example.com/icmpping,30m,,"0")>5
```

Это выражение будет истинным, если узел сети «example.example.com» недоступен более пяти раз за последние 30 минут.

Пример 8

Нет данных за последние 3 минуты.

Используется функция nodata():

```
nodata(/example.example.com/tick,3m)=1
```

Для того, чтобы этот триггер заработал, элемент данных «tick» должен быть определён как элемент данных с типом Zabbix траппер. Узел сети должен периодически отправлять данные этому элементу данных, используя zabbix_sender. Если никаких данных не было получено за последние 180 секунд, значение триггера станет ПРОБЛЕМА.

*Обратите внимание*, что «nodata» можно использовать с любым типом элементов данных.

Пример 9

Активность CPU в ночное время.

Используется функция time():

```
min(/Zabbix server/system.cpu.load[all,avg1],5m)>2 and time()<060000
```

Триггер может изменить свое состояние в ПРОБЛЕМА только в ночное время (00:00 — 06:00).

Пример 10

Активность CPU в любое время с исключением.

Используется функция time() и оператор **not**:

```
min(/zabbix/system.cpu.load[all,avg1],5m)>2
and not (dayofweek()=7 and time()>230000)
and not (dayofweek()=1 and time()<010000)
```

Триггер может изменить свое состояние на Проблема в любое время, за исключением интервала в 2 часа при изменении недели (воскресенье, 23:00 — понедельник, 01:00).

Пример 11

Проверка синхронизации времени на клиенте со временем на Zabbix сервере.

Используется функция fuzzytime():

```
fuzzytime(/MySQL_DB/system.localtime,10s)=0
```

Триггер изменит состояние на проблему тогда, когда локальное время на сервере MySQL_DB и Zabbix сервере будут различаться более чем на 10 секунд. Обратите внимание, что с Zabbix агентом «system.localtime» должен быть настроен как пассивная проверка; с Zabbix агентом 2 может быть настроено и как активная проверка.

Пример 12

Сравнение средней загрузки сегодня со средним значением загрузки за это же время вчера (использование сдвига времени в виде now-1d).

```
avg(/server/system.cpu.load,1h)/avg(/server/system.cpu.load,1h:now-1d)>2
```

Этот триггер сработает, если средняя загрузка за последний час более чем вдвое превысит среднюю загрузку за тот же час вчера.

Пример 13

Использование значения другого элемента данных в качестве порогового значения триггера:

```
last(/Template PfSense/hrStorageFree[{#SNMPVALUE}])<last(/Template PfSense/hrStorageSize[{#SNMPVALUE}])*0.
```

Триггер изменит свое состояние на проблему, если количество свободного пространства на диске упадёт ниже 10 процентов.

Пример 14

Использование результата вычисления для получения количества триггеров больше порога:

```
(last(/server1/system.cpu.load[all,avg1])>5) + (last(/server2/system.cpu.load[all,avg1])>5) + (last(/serve
```

Этот триггер сработает, если по крайней мере два триггера в выражении будут в состоянии ПРОБЛЕМА.

Пример 15

Сравнение строковых значений двух элементов данных: операнды здесь — это функции, которые возвращают строки.

Проблема: создание оповещения, если версия Ubuntu отличается от версии на другом узле сети

```
last(/NY Zabbix server/vfs.file.contents[/etc/os-release])<>last(/LA Zabbix server/vfs.file.contents[/etc/
```

Пример 16

Сравнение двух строковых значений — операнды здесь:

- функция, которая возвращает строку
- комбинация макросов и строк

Проблема: обнаружение изменений в DNS запросе

Ключ элемента данных:

```
net.dns.record[192.0.2.0,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1]
```

с макросами, заданными как

```
{$WEBSITE_NAME} = example.com
{$DNS_RESOURCE_RECORD_TYPE} = MX
```

который обычно возвращает:

```
example.com           MX       0 mail.example.com
```

Итак, наше выражение триггера определяет, отклоняется ли результат DNS запроса от ожидаемого значения:

```
last(/Zabbix server/net.dns.record[192.0.2.0,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1])<>"{$WEBSITE
```

Обратите внимание на кавычки вокруг второго операнда.

Пример 17

Сравнение двух строковых значений — операнды здесь:

- функция, которая возвращает строку
- строковая константа со специальными символами \ и ”

Проблема: определить, идентично ли содержимое файла /tmp/hello следующим данным:

```
\" //hello ?\"
```

Опция 1. Указать строку напрямую:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])="\\\" //hello ?\\\""
```

Обратите внимание на то, как экранируются символы \ и ” при сравнении строки напрямую.

Опция 2. Использование макроса

```
{$HELLO_MACRO} = \" //hello ?\"
```

в выражении:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])={$HELLO_MACRO}
```

Пример 18

Сравнение длительных периодов.

Проблема: Загрузка Exchange сервера увеличилась более чем на 10% за последний месяц

```
trendavg(/Exchange/system.cpu.load,1M:now/M)>1.1*trendavg(/Exchange/system.cpu.load,1M:now/M-1M)
```

Вы можете также использовать поле Имени события в настройках триггера, чтобы создать осмысленное сообщение оповещения, например, для получения примерно следующего

```
"Load of Exchange server increased by 24% in July (0.69) comparing to June (0.56)"
```

имя события должно быть определено следующим образом:

```
Load of {HOST.HOST} server increased by {{?100*trendavg(//system.cpu.load,1M:now/M)/trendavg(//system.cpu.
```

Также для подобных проблем полезно было бы разрешить закрывать триггер вручную в настройках самого триггера.

Гистерезис

Порой нам необходим некоторый интервал между состояниями OK и Проблема, а не просто порог. Например, если мы хотим задать триггер, который переходит в состояние Проблема, когда температура в серверной комнате становится больше 20°C, и мы хотим, чтобы он оставался в этом состоянии, пока температура не опустится ниже 15°C, то просто порога в 20°C для триггера будет недостаточно.

Вместо этого нам нужно сначала задать выражение триггера для события проблемы (температура выше 20°C). Затем нам нужно задать дополнительное условие восстановления (температура ниже 15°C). Это можно сделать, задавая дополнительный параметр *Выражение восстановления* в настройках триггера.

В этом случае восстановление проблемы будет проходить в два этапа:

- Во-первых, выражение проблемы (температура выше 20°C) должно быть вычислено в значение ЛОЖЬ
- Во-вторых, выражение восстановления (температура ниже 15°C) должно быть вычислено в значение ИСТИНА

Выражение восстановления будет вычисляться, только когда сначала будет решено событие о проблеме.

**Warning:**
Выражение восстановления, которое в состоянии ИСТИНА, не решает проблему, если выражение проблемы по-прежнему равно ИСТИНА!

Пример 1

Температура в серверной комнате слишком высокая.

Выражение проблемы:

```
last(/server/temp)>20
```

Выражение восстановления:

```
last(/server/temp)<=15
```

Пример 2

Очень мало свободного места на диске.

Выражение проблемы: если меньше 10ГБ за последние 5 минут

```
max(/server/vfs.fs.size[/,free],5m)<10G
```

Выражение восстановления: если больше 40ГБ за последние 10 минут

```
min(/server/vfs.fs.size[/,free],10m)>40G
```

Выражения с неизвестными операндами

В большинстве случаев, неизвестный операнд (такой как неподдерживаемый элемент данных) в выражении незамедлительно изменит значение триггера на .

Однако, в некоторых случаях неизвестные операнды (неподдерживаемые элементы данных, ошибки функций) при вычисления выражений допустимы:

- Функция `nodata()` вычисляется независимо от того, поддерживается элемент данных, на который она ссылается, или же не поддерживается.
- Логические выражения с ИЛИ и И могут быть вычислены в известные значения независимо от неизвестных операндов в двух случаях:
  - **Случай 1**: "1 `or` некая_функция(неподдерживаемый_элемент_данных1) `or` некая_функция(неподдерживаемый_элемент `or` ..." может быть вычислено с известным результатом («1» или "Проблема"),
  - **Случай 2**: "0 `and` некая_функция(неподдерживаемый_элемент_данных1) `and` некая_функция(неподдерживаемый_элеме `and` ..." может быть вычислено с известным результатом («0» или "OK").<br> Zabbix пытается вычислить такие логические выражения, принимая неподдерживаемые элементы данных как неизвестные операнды. В двух случаях, упомянутых выше, будет производиться известное значение; в остальных случаях значением триггера будет .
- Если вычисление функции по поддерживаемому элементу данных приведёт к ошибке, значением функции будет и оно будет частью дальнейшего вычисления выражения как неизвестный операнд.

Обратите внимание на то, что неизвестные операнды могут «исчезать» только в логических выражениях, описанных выше. В арифметических выражениях неизвестные операнды всегда приводят к результату (за исключением деления на 0).

**Attention:**
Выражение, приводящее к , не меняет состояние триггера («Проблема/OK»). Таком образом, если он был в состоянии «Проблема» (смотрите Случай 1), то остаётся в том же проблемном состоянии, даже если известная часть разрешается («1» становится «0»), потому что выражение теперь вычисляется в , а это не меняет состояние триггера.

Если выражение триггера с несколькими неподдерживаемыми элементами данных вычисляется как , сообщение об ошибке в веб-интерфейсе ссылается на последний вычисленный неподдерживаемый элемент данных.

### 3. Зависимости триггеров

Обзор

Иногда доступность одного узла сети зависит от другого. Сервер, который находится за маршрутизатором, станет недоступен, если маршрутизатор выйдет из строя. Если для обоих настроены триггеры, вы можете получить уведомления о недоступности двух узлов сети, хотя виноват был только маршрутизатор.

В таких случаях может быть полезна некоторая зависимость между узлами сети. При заданной зависимости уведомления о зависимых узлах могут быть задержаны, и будет отправлено только уведомление о корневой проблеме.

Хотя Zabbix не поддерживает зависимости между узлами сети напрямую, их можно определить с помощью другого, более гибкого метода - зависимостей триггеров. Триггер может зависеть от одного или нескольких других триггеров.

В нашем простом примере мы открываем форму настройки триггера сервера и задаем, что он зависит от соответствующего триггера маршрутизатора. При такой зависимости триггер сервера не будет менять свое состояние, пока триггер, от которого он зависит, находится в состоянии 'PROBLEM' - и, следовательно, никаких действий для зависимого триггера выполняться не будет и уведомления отправляться не будут.

Если недоступны и сервер, и маршрутизатор, и задана зависимость между триггером "server down" и триггером "router down", Zabbix не будет выполнять действия для зависимого триггера.

Пока родительский триггер находится в состоянии PROBLEM, его зависимые триггеры могут сообщать значения, которым нельзя доверять. Поэтому зависимые триггеры не будут переоцениваться, пока родительский триггер (маршрутизатор в примере выше):

- не вернется из состояния 'PROBLEM' в состояние 'OK';
- не изменит свое состояние с 'PROBLEM' на 'UNKNOWN';
- не будет закрыт вручную, по корреляции или с помощью функций date and time и/или nodata();
- не будет устранен значением элемента данных, не участвующего в зависимом триггере;
- не будет отключен, либо будет отключен элемент данных или узел сети, которому принадлежит отключенный элемент данных

Во всех случаях, перечисленных выше, зависимый триггер (сервер) будет переоценен только после получения нового метрика для него. Это означает, что зависимый триггер может обновиться не сразу.

Также:

- Зависимость триггера может быть добавлена от любого триггера узла сети к любому другому триггеру узла сети, если это не приводит к циклической зависимости.
- Зависимость триггера может быть добавлена от одного шаблона к другому. Если какой-либо триггер из шаблона A зависит от какого-либо триггера из шаблона B, шаблон A можно связать с узлом сети (или другим шаблоном) только вместе с шаблоном B, но шаблон B можно связать с узлом сети (или другим шаблоном) отдельно.
- Зависимость триггера может быть добавлена от триггера шаблона к триггеру узла сети. В этом случае связывание такого шаблона с узлом сети создаст триггер узла сети, который будет зависеть от того же триггера шаблона, от которого зависел триггер. Это позволяет, например, иметь шаблон, в котором некоторые триггеры зависят от триггеров маршрутизатора (узла сети). Все узлы сети, связанные с этим шаблоном, будут зависеть от этого конкретного маршрутизатора.
- Зависимость триггера не может быть добавлена от триггера узла сети к триггеру шаблона.
- Зависимость триггера может быть добавлена от прототипа триггера к другому прототипу триггера (в рамках одного правила низкоуровневого обнаружения) или к реальному триггеру. Прототип триггера не может зависеть от прототипа триггера из другого правила LLD или от триггера, созданного из прототипа триггера. Прототип триггера узла сети не может зависеть от триггера из шаблона.

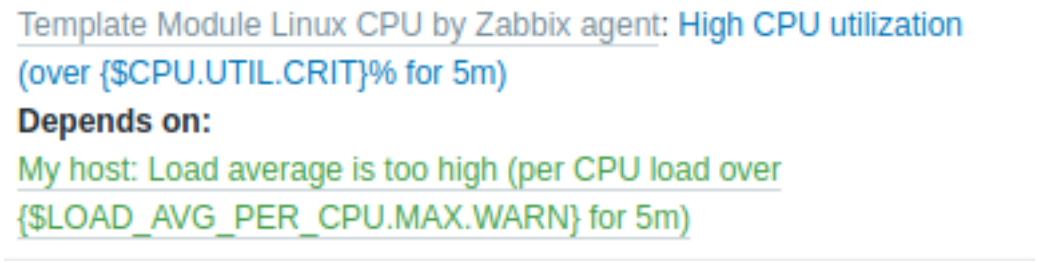
Конфигурация

Чтобы задать зависимость:

1. Откройте вкладку *Dependencies* в форме конфигурации триггера.
2. Нажмите *Add* в разделе *Dependencies* и выберите один или несколько триггеров, от которых будет зависеть этот триггер.

3. Нажмите *Update*.

Теперь в списке у триггера отображается признак его зависимости.




Пример нескольких зависимостей

Например, узел сети Host находится за Router2, а Router2 находится за Router1.

```
Zabbix – Router1 – Router2 – Host
```

Если Router1 недоступен, то, очевидно, узел сети Host и Router2 тоже недоступны, однако получение трех уведомлений о том, что недоступны узел сети Host, Router1 и Router2, — это избыточно.

Поэтому в этом случае мы определяем две зависимости:

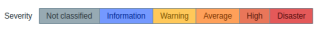
```
    'Host is down'              'Router2 is down'
    'Router2 is down'              'Router1 is down'
```

Перед изменением состояния триггера 'Host is down' Zabbix проверит соответствующие зависимости триггеров. Если такие зависимости найдены и один из этих триггеров находится в состоянии 'Problem', то статус триггера не будет изменен, действия не будут выполнены и уведомления не будут отправлены.

Zabbix выполняет эту проверку рекурсивно. Если Router1 или Router2 недоступен, триггер Host не будет обновлен.

#### 4. Важность триггеров

Важность триггера определяет, насколько триггер важен.




Zabbix по умолчанию поддерживает следующие важности триггеров:

| Важность | Цвет | Описание |
|---|---|---|
| Не классифицировано (Not classified) | Серый | Может использоваться, когда уровень важности события неизвестен, не был определён, не входит в рамки обычного мониторинга и т.п.; например, во время первоначальной настройки, как заполнитель для последующей оценки, как часть процесса интеграции. |
| Информация (Information) | Светло-синий | Может использоваться для информационных событий, которые не требуют немедленного внимания, но могут дать ценную информацию. |
| Предупреждение (Warning) | Жёлтый | Может использоваться для обозначения потенциальной проблемы, которая может потребовать расследования или принятия мер, но не является критической. |

| Важность | Цвет | Описание |
|---|---|---|
| Средняя (Average) | Оранжевый | Может использоваться для обозначения серьёзной проблемы, которую следует решить относительно быстро, чтобы предотвратить дальнейшие проблемы. |
| Высокая (High) | Светло-красный | Может использоваться для обозначения критических проблем, требующих немедленного внимания во избежание значительных сбоев. |
| Чрезвычайная (Disaster) | Красный | Может использоваться для обозначения серьёзного инцидента, который требует немедленных действий для предотвращения, например, сбоев в работе системы или потери данных. |

**Note:**
Имеется возможность настроить имена важностей триггеров и их цвета.

Важности триггеров используются для:

- визуального представления триггеров — различные цвета для различных уровней важности;
- аудио в глобальных оповещениях — различные звуковые сигналы для различных важностей;
- оповещений пользователей — различные способы оповещений (каналы оповещения) для различных важностей. Например, SMS — для триггеров высокой и чрезвычайной важности, Email — для остального;
- ограничения действий по условиям важностей триггеров.

#### 5. Пользовательские важности триггеров

Названия важностей триггеров и цвета элементов веб-интерфейса, связанных с важностью, можно настроить в *Администрирование → Общие → Опции отображения триггеров*. Цвета являются общими по всем темам веб-интерфейса.

Перевод пользовательских названий уровней важности

**Attention:**
Если используются переводы веб-интерфейса Zabbix, по умолчанию пользовательские названия уровней важности переопределят переведённые названия.

Названия уровней важности триггеров по умолчанию доступны для перевода во всех локалях. Если название уровня важности изменено, пользовательское название будет использовано во всех локалях и потребуется дополнительный ручной перевод.

Процедура перевода пользовательского названия уровня важности:

- задайте требуемое пользовательское название важности, например: «Important»
- откройте на редактирование <папка_с_веб-интерфейсом>/locale/<требуемая_локаль>/LC_MESSAGES/frontend.po
- добавьте 2 строки:

```
msgid "Important"
msgstr "<           >"
```

и сохраните файл.

- создайте .mo файлы, как описано в <папка_с_веб-интерфейсом>/locale/README

Здесь **msgid** должно совпадать с новым пользовательским названием уровня важности, а **msgstr** должно являться его переводом для указанного языка.

Эту процедуру необходимо выполнить после каждого изменения названия важности.

#### 6. Массовое обновление

Обзор

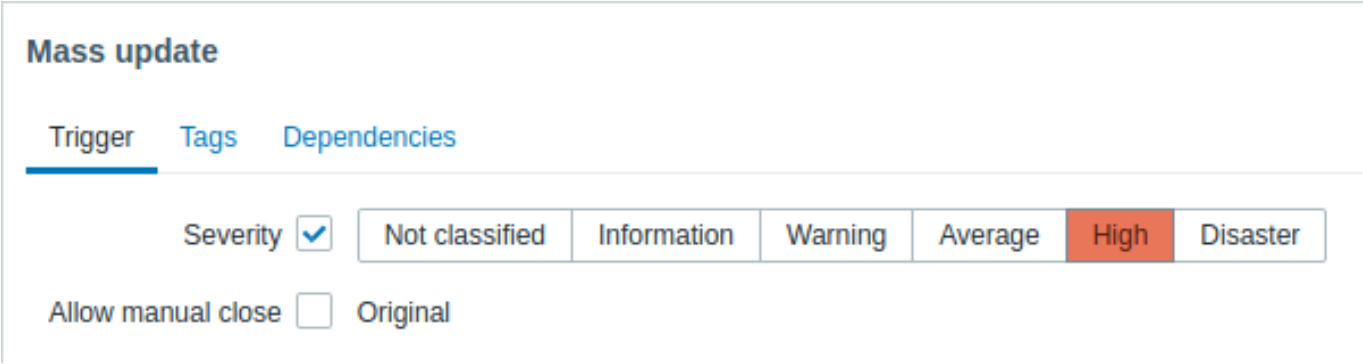
Используя массовое обновление, вы можете изменить некоторые атрибуты у нескольких триггеров за один раз, что избавит вас от необходимости открывать на изменение каждый отдельный триггер.

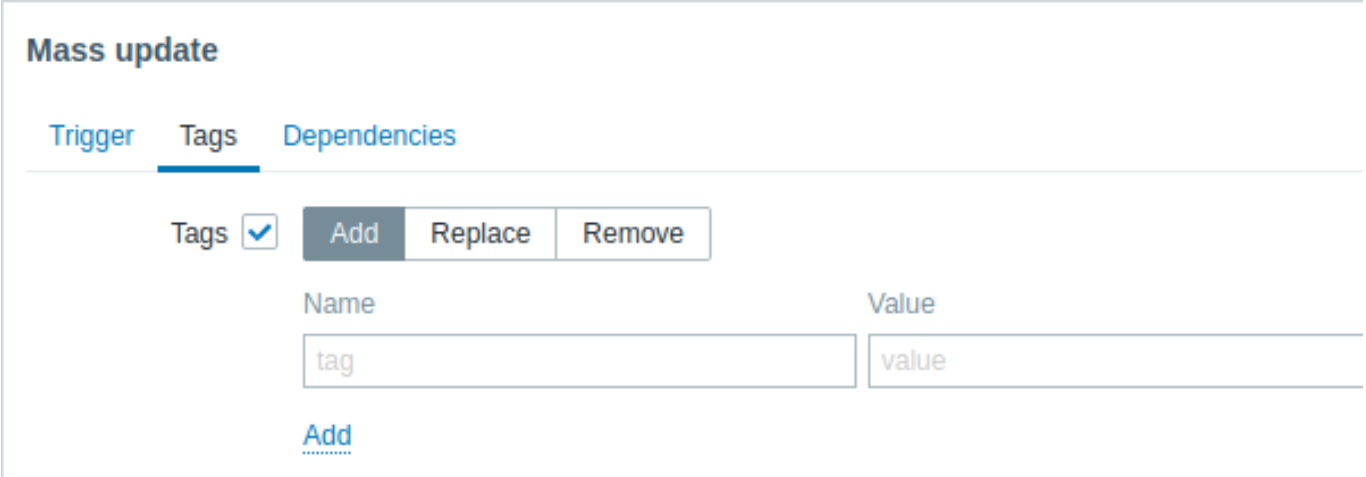
Использование массового обновления

Чтобы массово обновить некоторые триггеры, выполните следующие действия:

- Отметьте флажками триггеры, которые нужно обновить, в списке
- Нажмите *Mass update* под списком

- Перейдите на вкладку с нужными атрибутами (*Trigger*, *Tags* или *Dependencies*)
- Отметьте флажками любые атрибуты, которые нужно обновить

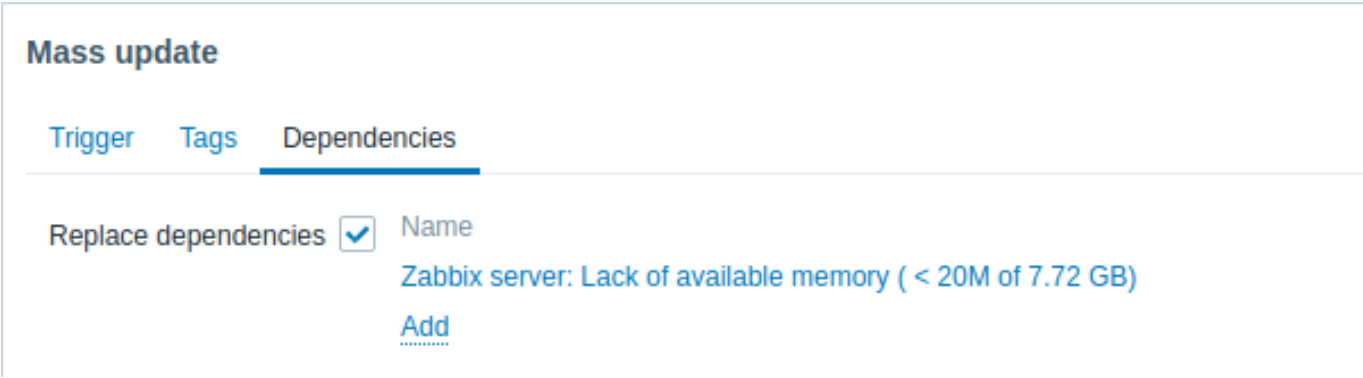



При выборе соответствующей кнопки для обновления тегов доступны следующие варианты:

- *Add* - позволяет добавить новые теги к триггерам;
- *Replace* - удалит все существующие теги из триггера и заменит их указанными ниже;
- *Remove* - удалит указанные теги из триггеров.

Обратите внимание, что теги с одинаковым именем, но разными значениями не считаются "дубликатами" и могут быть добавлены к одному и тому же триггеру.




*Replace dependencies* - удалит все существующие зависимости из триггера и заменит их указанными.

Нажмите *Update*, чтобы применить изменения.

**7. Прогнозирующие функции триггеров**

Обзор

Иногда имеются признаки надвигающейся проблемы. Эти признаки можно заметить, таким образом можно заблаговременно предпринять меры, чтобы предотвратить или хотя бы уменьшить влияние проблемы.

В Zabbix имеются средства прогнозирования будущего поведения наблюдаемой системы на основе исторических данных. Эти средства реализованы в виде прогнозирующих функций триггеров.

Функции

Перед настройкой триггера необходимо определить, что такое состояние проблемы и сколько времени нужно для принятия мер. Далее есть два способа настроить триггер, сигнализирующий о возможной нежелательной ситуации. Первый: триггер должен сработать, когда ожидается, что после «времени реагирования» система будет в проблемном состоянии. Второй: триггер должен сработать, когда прогнозируется, что система достигнет проблемного состояния за время, меньшее, чем «время реагирования». Для этого используются функции триггеров **forecast** и **timeleft** соответственно. Надо отметить, что статистический анализ, лежащий в основе этих двух функций, в основном идентичен. Вы можете настроить триггер любым удобным для вас способом с аналогичными результатами.

Параметры

Обе функции используют практически идентичный набор параметров. Воспользуйтесь списком поддерживаемых функций для справки.

Интервал времени

Прежде всего вам необходимо указать исторический период, который Zabbix должен проанализировать для составления прогноза. Делайте это привычным способом при помощи параметра и необязательного сдвига времени по аналогии с функциями **avg**, **count**, **delta**, **max**, **min** и **sum**.

Горизонт предсказания

(Только **forecast**)<br> Параметр определяет, насколько далеко в будущее Zabbix будет экстраполировать зависимости, которые ему удастся найти в исторических данных. Независимо от того, используете ли вы _ или нет, всегда отсчитывается от текущего момента.

Достигаемый порог

(Только **timeleft**)<br> Параметр задаёт значение, которого должен достичь анализируемый элемент данных, вне зависимости от того, сверху или снизу. После того, как мы определили f(t) (смотрите ниже), мы должны решить уравнение f(t) = и вернуть ближайший к текущему моменту корень, который находится справа от текущего момента, или вернуть 1.7976931348623158E+308, если таковых корней нет.

**Note:**
Когда значения элемента данных приближаются к порогу, а затем пересекают его, **timeleft** делает вывод, что пересечение уже находится в прошлом, и поэтому переключается на следующее пересечение с уровнем , если таковое имеется. Наилучшим решением было бы использовать прогнозирования в качестве дополнения к обычным диагностикам проблем, а не замены одного на другое.[1]

Функции аппроксимации

По умолчанию является линейной (*linear*) функцией. Но если наблюдаемая система более сложная, вы можете выбрать один из следующих вариантов.

| | x = f(t) |
|---|---|
| линейная (*linear*) | $x = a + b*t$ |
| полиномиальная (*polynomialN*)[2] | $x = a_0 + a_1*t + a_2*t^2 + ... + a_n*t^n$ |
| экспоненциальная (*exponential*) | $x = a*\exp(b*t)$ |
| логарифмическая (*logarithmic*) | $x = a + b*\log(t)$ |
| степенная (*power*) | $x = a*t^b$ |

Режимы

(Только **forecast**)<br> Каждый раз, когда вычисляется функция триггера, данные запрашиваются из указанного периода истории и по полученным данным строится указанная аппроксимация. Поэтому, если данные немного изменятся, то и построенная аппроксимация немного изменится. Если мы будем просто рассчитывать значение функции аппроксимации в заданный момент времени в будущем, то вы ничего не будете знать о том, как согласно прогнозу будет меняться анализируемый элемент данных между текущим моментом и этим моментом в будущем. При некоторых параметрах (вроде *polynomial*) просто лишь одно значение из будущего может ввести в заблуждение.

| | Результат **forecast** |
|---|---|
| значение (*value*) | f(сейчас + ) |
| максимум (*max*) | $\max_{\text{сейчас} <= t <= \text{сейчас} +}$ f(t) |
| минимум (*min*) | $\min_{\text{сейчас} <= t <= \text{сейчас} +}$ f(t) |
| дельта (*delta*) | *max - min* |

| | Результат **forecast** | |
|---|---|---|
| среднее (*avg*) | среднее значение f(t) (сейчас <= t <= сейчас + ) | в соответствии с определением |

Подробности

Для того, чтобы избежать вычислений с большими числами, мы рассматриваем штамп времени первого значения в указанном периоде плюс 1 наносекунда как новую точку отсчёта времени (текущие штампы времени порядка $10^9$, в квадрате уже $10^{18}$, а точность дробных значений около $10^{-16}$). 1 наносекунда прибавляется для того, чтобы все значения времени были положительными, поскольку построение логарифмической (*logarithmic*) и степенной (*power*) аппроксимаций подразумевает вычисление log(t). Этот сдвиг времени не влияет на линейную (*linear*), полиномиальную (*polynomial*) и экспоненциальную (*exponential*) функции (за исключением более легких и более точных вычислений), но изменяет форму логарифмической (*logarithmic*) и степенной (*power*) функций.

Возможные ошибки

Функции возвращают -1 в следующих ситуациях:

- указанных период не содержит данных;
- результат математической операция не определён[3];
- сложности вычислений (к сожалению, для некоторых наборов входных данных диапазона и точности формата чисел с плавающей точкой двойной точности становится недостаточно)[4].

**Note:**
Никаких предупреждений или ошибок не выдаётся, если выбранная функция аппроксимации плохо описывает предоставленные данные или данных просто недостаточно для точного прогноза.

Примеры и обработка ошибок

Чтобы получать предупреждение, когда на вашем узле сети заканчивается свободное место на диске, можно использовать такое выражение триггера:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h
```

Однако может возникнуть код ошибки -1, и тогда ваш триггер перейдет в состояние проблемы. В целом это хорошо, потому что вы получите предупреждение о том, что ваши прогнозы работают некорректно, и вам следует изучить их более тщательно, чтобы выяснить причину. Но иногда это плохо, потому что -1 может просто означать, что за последний час не было получено данных о свободном месте на диске узла сети. Если у вас слишком много ложноположительных оповещений, рассмотрите использование более сложного выражения триггера [5]:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1
```

С **forecast** ситуация немного сложнее. Прежде всего, -1 может перевести триггер в состояние проблемы, а может и не перевести, в зависимости от того, используете ли вы выражение вида `forecast(/host/item,(...))<...` или `forecast(/host/item,(...))>...`

Кроме того, -1 может быть допустимым прогнозом, если для значения элемента данных нормально быть отрицательным. Но вероятность такой ситуации в реальном мире ничтожно мала (см. как работает оператор =). Поэтому добавьте `... or forecast(/host/item,(...))=-1` или `... and forecast(/host/item,(...))<>-1`, если вы хотите или не хотите соответственно считать -1 проблемой.

Примечания

[1] Например, простой триггер вроде `timeleft(/host/item,1h,X) < 1h` может перейти в состояние проблемы, когда значение элемента данных приближается к X, а затем внезапно восстановиться, как только значение X будет достигнуто. Если проблема заключается в том, что значение элемента данных ниже X, используйте: `last(/host/item) < X or timeleft(/host/item,1h,X) < 1h` Если проблема заключается в том, что значение элемента данных выше X, используйте: `last(/host/item) > X or timeleft(/host/item,1h,X) < 1h`

[2] Степень полинома может быть от 1 до 6, *polynomial1* эквивалентен *linear*. Однако используйте полиномы более высокой степени с осторожностью. Если период оценки содержит меньше точек, чем требуется для определения коэффициентов полинома, степень полинома будет понижена (например, запрошен *polynomial5*, но есть только 4 точки, поэтому будет подогнан *polynomial3*).

[3] Например, подгонка функций *exponential* или *power* включает вычисление log() от значений элемента данных. Если данные содержат нули или отрицательные числа, будет выдана ошибка, поскольку log() определена только для положительных значений.

[4] Для подгонок *linear*, *exponential*, *logarithmic* и *power* все необходимые вычисления можно записать явно. Для *polynomial* без дополнительных шагов можно вычислить только *value*. Вычисление *avg* включает вычисление полиномиальной

первообразной (аналитически). Вычисление *max*, *min* и *delta* включает вычисление производной полинома (аналитически) и нахождение ее корней (численно). Решение f(t) = 0 включает нахождение корней полинома (численно).

[5] Но в этом случае -1 может привести к тому, что ваш триггер восстановится из состояния проблемы. Чтобы обеспечить полную защиту, используйте: `timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and ({TRIGGER.VALUE}=0 and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1 or {TRIGGER.VALUE}=1)`

### 4. События

Обзор

В Zabbix генерируются несколько типов событий:

- события триггеров - всякий раз, когда триггер изменяет свой статус (*OK→PROBLEM→OK*)
- события сервисов - всякий раз, когда сервис изменяет свой статус (*OK→PROBLEM→OK*)
- события обнаружения - когда обнаруживаются узлы сети или сервисы
- события автоматической регистрации - когда активные агенты автоматически регистрируются сервером
- внутренние события - когда элемент данных/правило низкоуровневого обнаружения становится неподдерживаемым или триггер переходит в неизвестное состояние

События имеют временную метку и могут служить основой для таких действий, как отправка уведомлений по электронной почте и т. д.

Чтобы просмотреть сведения о событиях в веб-интерфейсе, перейдите в *Monitoring → Problems*. Там можно щелкнуть по дате и времени события, чтобы просмотреть сведения о нем.

Дополнительная информация доступна здесь:

- события триггеров
- другие источники событий

### 1. Генерация событий на триггеры

Обзор

Изменение состояния триггера является наиболее частым и наиболее важным источником событий. Каждый раз, когда триггер меняет своё состояние, генерируется событие. Событие содержит подробную информацию об изменении состояния триггера — когда это случилось, и какое сейчас новое состояние.

Триггерами создаются два типа событий — Проблема и OK.

События проблемы

Событие проблемы создаётся:

- когда выражение триггера вычисляется как ИСТИНА, если триггер был в состоянии OK;
- каждый раз выражение триггера вычисляется как ИСТИНА, если у триггера активирована опция множественной генерации событий проблема.

События OK

Событие OK закрывает связанные событие(-я) о проблеме и может быть создано тремя компонентами:

- триггеры — на основании настроек «Генерация OK событий» и «OK событие закрывает»;
- корреляция событий
- менеджер задач — при закрытии вручную события

Триггеры

У триггеров имеется настройка «Генерация OK событий», которая определяет, как генерируются OK события:

- *Выражение* — событие OK генерируется для триггера в состоянии проблема, когда выражение этого триггера вычисляется как ЛОЖЬ. Это самая простая настройка, активированная по умолчанию.
- *Выражение восстановления* — событие OK генерируется для триггера в состоянии проблема, когда выражение этого триггера вычисляется как ЛОЖЬ и выражение восстановления вычисляется как ИСТИНА. Эту опцию можно использовать, если критерий восстановления триггера отличается от критерия проблемы.
- *Нет* — событие OK никогда не генерируется. Эту опцию можно использовать в сочетании с множественной генерацией проблем для простой отправки оповещений, когда что-то происходит.

Кроме того, у триггеров имеется настройка «OK событие закрывает», которая определяет, какие события о проблемах должны быть закрыты:

- *Все проблемы* — событие OK закроет все открытые проблемы, созданные этим триггером
- *Все проблемы, если значение тега совпадает* — событие OK закрывает все открытые проблемы, которые созданы этим триггером, и имеющие по крайней мере одно совпадающее значение тега. Тег задается настройкой триггера «Теги для совпадения». Если нет никаких событий о проблемах для закрытия, тогда событие OK не генерируется. Эту опцию часто называют корреляцией событий на уровне триггеров.

Корреляция событий

Корреляция событий (также называемая глобальной корреляцией событий) является способом настройки пользовательских правил закрытия событий (в результате чего генерируются OK события).

Правила определяют, каким образом новые события проблемы соотносятся с существующими событиями проблемы и позволяют закрыть новое событие или совпадающие события при помощи генерации OK событий.

Однако, корреляцию событий необходимо настраивать очень осторожно, так как она может негативно повлиять на производительность обработки событий или, при неправильной настройке, закрыть больше событий, чем требуется (в худшем случае могут быть закрыты вообще все события о проблемах). Несколько советов по настройке:

1. всегда старайтесь уменьшить масштаб корреляции, указав уникальный тег для контрольного события (парное событие со старыми событиями) и использовав условие корреляции «тег нового события»
2. не забывайте добавлять условие на основе старого события при использовании операции «закрыть старые события», в противном случае могут быть закрыты все существующие проблемы
3. избегайте использования распространенных имён тегов, которые используются различными настройками корреляции

Менеджер задач

Если у триггера активирована настройка «Разрешить закрывать вручную», тогда появится возможность закрывать вручную те события о проблемах, которые созданы этим триггером. Это можно сделать в веб-интерфейсе при обновлении проблемы. Событие не закрывается напрямую — вместо этого создается задача «закрытие события», которая в ближайшее время обрабатывается менеджером задач. Диспетчер задач создаст соответствующее OK событие, и событие проблемы будет закрыто.

**2. Другие источники событий**

События на услуги

События на услуги генерируются, только если действия на услуги активированы. В этом случае каждое изменение состояния услуги создает новое событие:

- Событие о проблеме — когда состояние услуги меняется с OK на ПРОБЛЕМА
- Событие OK — когда состояние услуги меняется с ПРОБЛЕМА на OK

Событие содержит подробную информацию об изменении состояния услуги — когда произошло изменение и какое теперь новое состояние.

События на обнаружения

Zabbix периодически сканирует диапазоны IP адресов, заданные в правилах сетевого обнаружения. Частота этой проверки настраивается отдельно по каждому правилу. После того, как узел сети или сервис обнаружен, генерируется событие (или несколько событий) на обнаружение.

Zabbix генерирует следующие события:

| Событие | Когда генерируется |
|---|---|
| Сервис доступен | Каждый раз, когда Zabbix обнаруживает активный сервис. |
| Сервис недоступен | Каждый раз, когда Zabbix не может обнаружить сервис. |
| Узел сети доступен | Если хотя бы один сервис доступен для IP. |
| Узел сети недоступен | Если все сервисы не доступны. |
| Сервис обнаружен | Если сервис стал доступен после его недоступности или обнаружен впервые. |
| Сервис потерян | Если сервис потерян после того как был доступен. |
| Узел сети обнаружен | Если узел сети стал доступен после его недоступности или обнаружен впервые. |
| Узел сети потерян | Если узел сети потерян после того как был доступен. |

События на авторегистрацию активных агентов

Авторегистрация активных агентов создает события в Zabbix.

Если настроено, событие на авторегистрацию активного агента создаётся, если ранее неизвестный активный агент запрашивает свои проверки или если изменились метаданные узла сети. Сервер добавляет новый автоматически зарегистрированный узел сети, используя полученные IP адрес и порт агента.

Для получения более подробной информации, смотрите страницу об автоматической регистрации активных агентов.

Внутренние события

Внутренние события возникают, когда:

- элемент данных меняет своё состояние с «нормального» на «не поддерживается»
- элемент данных меняет своё состояние с «не поддерживается» на «нормальное»
- правило низкоуровневого обнаружения меняет своё состояние с «нормального» на «не поддерживается»
- правило низкоуровневого обнаружения меняет своё состояние с «не поддерживается» на «нормальное»
- триггер меняет своё состояние с «нормального» на «неизвестное»
- триггер меняет своё состояние с «неизвестного» на «нормальное»

Цель введения внутренних событий — дать возможность пользователям узнавать, когда происходят некоторые внутренние события; например, элемент данных становится неподдерживаемым и перестаёт собирать данные.

Внутренние события создаются, только когда активированы действия на эти события. Чтобы остановить генерирование внутренних событий (например, о том, что элементы данных становятся неподдерживаемыми), отключите все действия на внутренние события в Настройка → Действия → Внутренние действия.

**Note:**
Если внутренние действия отключены, когда какой-либо объект находился в «неподдерживаемом» состоянии, событие восстановления по этому объекту все же будет создано.
Если внутренние действия активированы, когда какой-либо объект находился в «неподдерживаемом» состоянии, событие восстановления по этому объекту будет создано, даже если «событие о проблеме» не было ранее создано по этому объекту.

Смотрите также: Получение оповещений на неподдерживаемые элементы данных

### 3. Закрытие проблем вручную

Обзор

Хотя, как правило, события о проблемах решаются автоматически, когда состояние триггера меняется с «Проблема» на «OK», возможны случаи, когда при помощи выражения триггера трудно определить факт того, что проблема уже решена. В таких случаях проблему нужно решить вручную.

Например, *syslog* может сообщать о том, что для оптимальной производительности необходимо отрегулировать некоторые параметры ядра. В это случае информация о проблеме сообщается Linux администраторам, они исправляют её и затем вручную закрывают эту проблему.

Проблемы можно закрывать вручную только у триггеров, у которых включена опция *Разрешить закрывать вручную*.

Когда проблема «закрывается вручную», Zabbix генерирует новую внутреннюю задачу для Zabbix сервера. Затем процесс *диспетчер задач* выполняет эту задачу и генерирует событие OK, тем самым закрывая событие о проблеме.

Закрытая вручную проблема не означает, что её триггер никогда больше не перейдет в состояние «Проблема» снова. Выражение триггера пересчитывается, что может сгенерировать проблему:

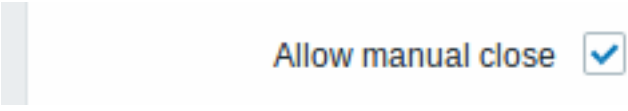
- При поступлении новых данных от любого элемента данных, которые входят в выражение триггера (обратите внимание, что значения, отбрасываемые шагом троттлинга предварительной обработки, не учитываются как поступившие и не могут быть причиной пересчёта выражения триггера);
- Когда в выражении используются функции даты и времени и/или функция nodata().

Настройка

Для закрытия проблем вручную необходимо выполнить два шага.

Настройка триггера

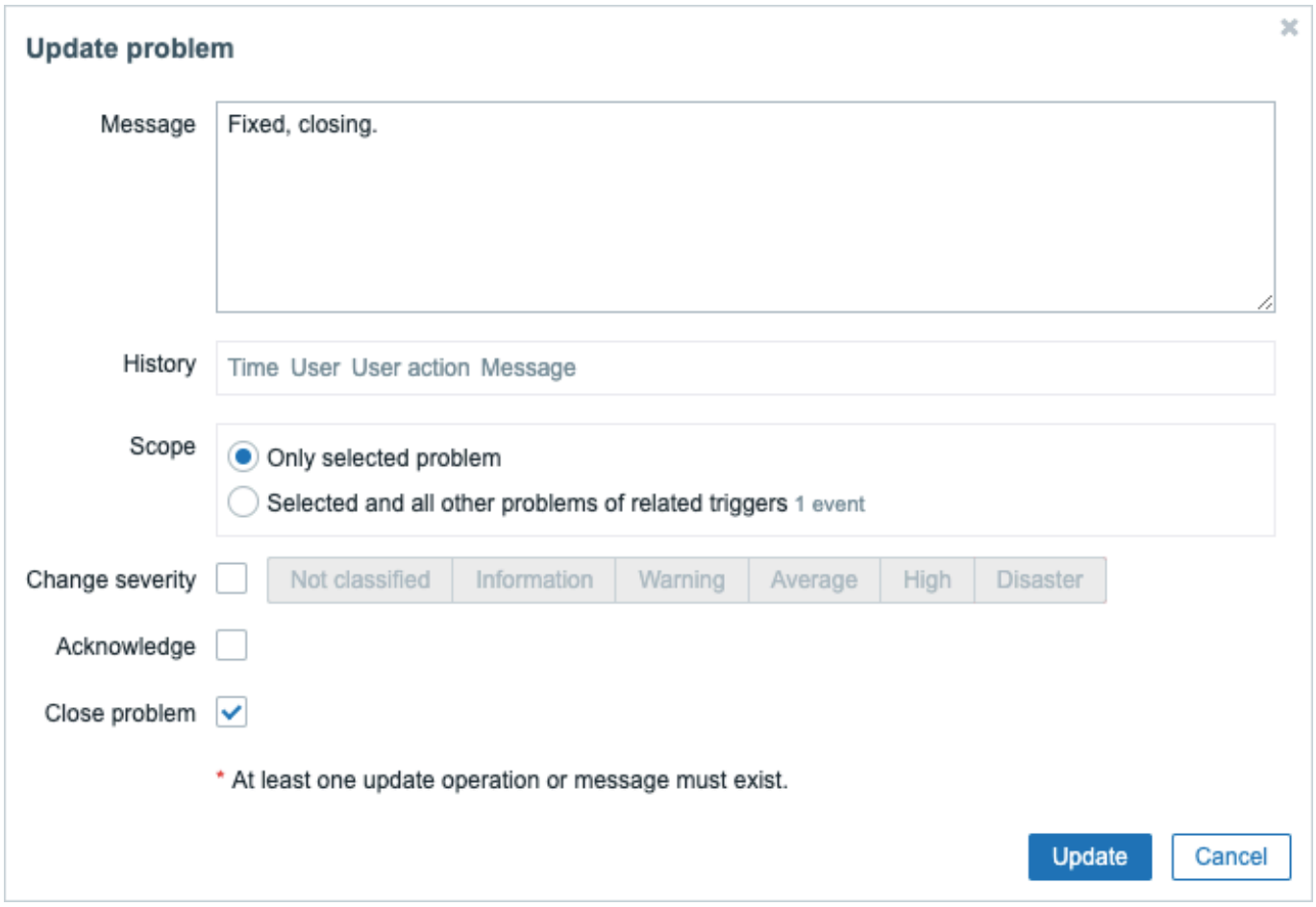
В настройке триггера включите опцию *Разрешить закрывать вручную (Allow manual close)*.



Окно обновления проблемы

Если возникает проблема для триггера с флагом *Manual close*, вы можете открыть всплывающее окно обновления проблемы для этой проблемы и закрыть проблему вручную.

Чтобы закрыть проблему, установите флажок *Close problem* в форме и нажмите *Update*.




Все обязательные поля ввода отмечены красной звездочкой.

Запрос обрабатывается сервером Zabbix. Обычно для закрытия проблемы требуется несколько секунд. В течение этого процесса в *Monitoring* → *Problems* в качестве статуса проблемы отображается *CLOSING*.

Проверка

Можно проверить, что проблема была вручную закрыта:

- в деталях события, доступных через *Мониторинг* → *Проблемы*;
- используя макрос {EVENT.UPDATE.HISTORY} в сообщениях оповещения, который предоставит такую информацию.

**5. Корреляция событий**

Обзор

Корреляция событий позволяет сопоставлять события о проблемах с их решениями очень точным и гибким способом.

Корреляцию событий можно задавать:

- на уровне триггера — один триггер можно использовать, чтобы связать отдельные проблемы с их решением
- глобально — проблемы можно соотнести с их решением с другого триггера/метода опроса с использованием глобальных правил корреляции

**1. Корреляция событий на основе триггеров**

Обзор

Корреляция событий на основе триггеров позволяет сопоставлять отдельные проблемы, о которых сообщает один триггер.

В то время как в большинстве случаев события OK могут закрывать все события о проблеме, созданные одним триггером, бывают случаи, когда необходим более обстоятельный подход. Например, при мониторинге файлов журналов вы можете захотеть обнаруживать некоторые проблемы в файле журнала и закрывать их по отдельности, а не все разом.

Это тот случай, когда у триггеров параметр *Режим генерации событий ПРОБЛЕМА* выставлен в значение *Множественный*. Такие триггеры обычно используются для мониторинга журналов, обработки трапов и т.п.

В Zabbix имеется возможность сопоставить события о проблемах, основываясь на тегах событий. Теги используются для извлечения значений и создания метки по событиям о проблемах. Используя преимущества такого подхода, проблемы могут быть закрыты отдельно на основании совпадения тега.

Другими словами, один триггер может создавать отдельные события, которые идентифицируются при помощи тега событий. Следовательно, события о проблемах могут быть идентифицированы каждый по отдельности и закрыты индивидуально на основании метки тега события.

Как это работает

В мониторинге журналов вы можете встретиться со строками, похожими на эти:

```
1:      1
2:      2
3:      1
4:      2
```

Идея корреляции событий состоит в том, чтобы была возможность сопоставить событие о проблеме из Строки1 с решением из Строки3 и событие о проблеме из Строки2 с решением из Строки4 и закрыть эти проблемы по отдельности:

```
1:      1
3:      1       #       1

2:      2
4:      2       #       2
```

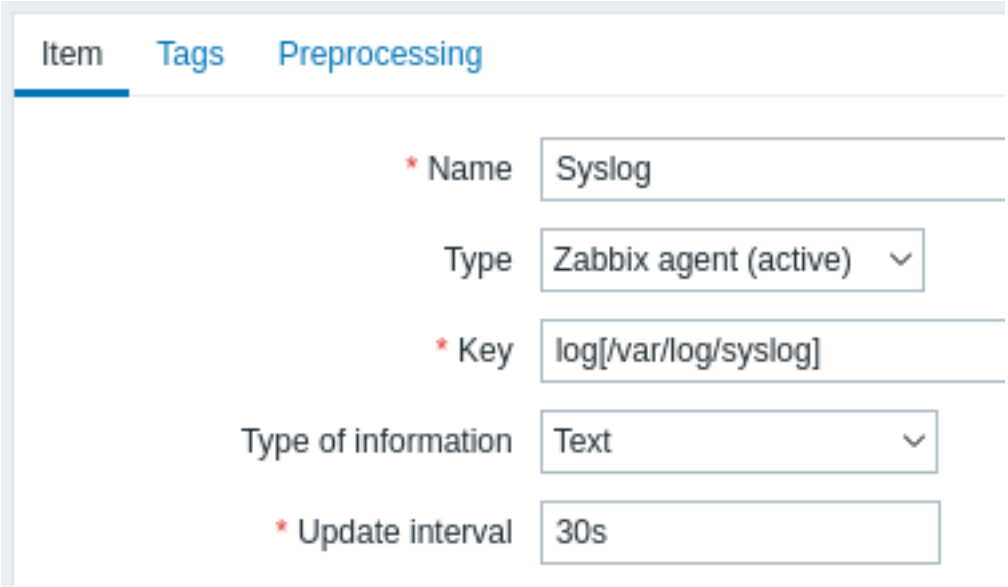
Чтобы такое сделать, вам необходимо пометить эти связанные события как, например, «Служба 1» и «Служба 2». Это можно сделать, применив регулярное выражение к строке из файла журнала, чтобы извлечь значение тега. Затем при создании события они будут помечены как «Служба 1» и «Служба 2» соответственно, и проблема может быть сопоставлена с решением.

Настройка

Элемент данных

Для начала вы можете настроить элемент данных, который мониторит файл журнала, например:

`log[/var/log/syslog]`




Когда элемент данных будет настроен, подождите минуту, пока изменения конфигурации вступят в силу, и затем перейдите в Последние данные, чтобы убедиться, что элемент данных начал собирать данные.

Триггер

Когда элемент данных заработал, вам необходимо настроить триггер. Важно решить, на какие записи в файле журнала необходимо обратить внимание. Например, следующее выражение триггера будет искать строки, которые содержат «Stopping», для предупреждения о возможных проблемах:

```
find(/My host/log[/var/log/syslog],,"regexp","Stopping")=1
```

**Attention:**
Чтобы убедиться, что каждая строка, которая содержит подстроку «Stopping», считается проблемой, также в настройках триггера выставьте *Режим генерации событий ПРОБЛЕМА* в значение «Множественный».

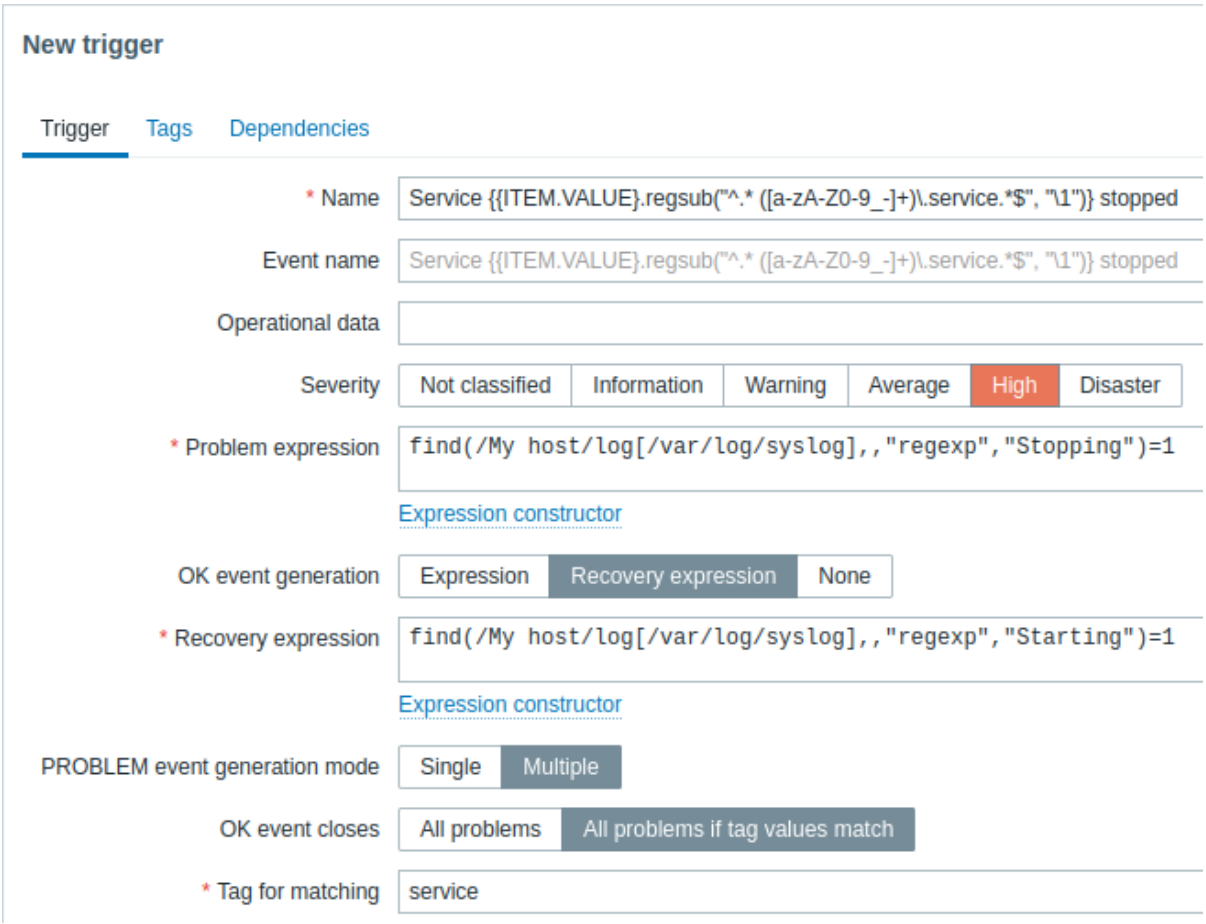
Затем определим выражение восстановления. Следующие выражение решит все проблемы, если в журнале будет найдена строка, содержащая подстроку «Starting»:

```
find(/My host/log[/var/log/syslog],,"regexp","Starting")=1
```

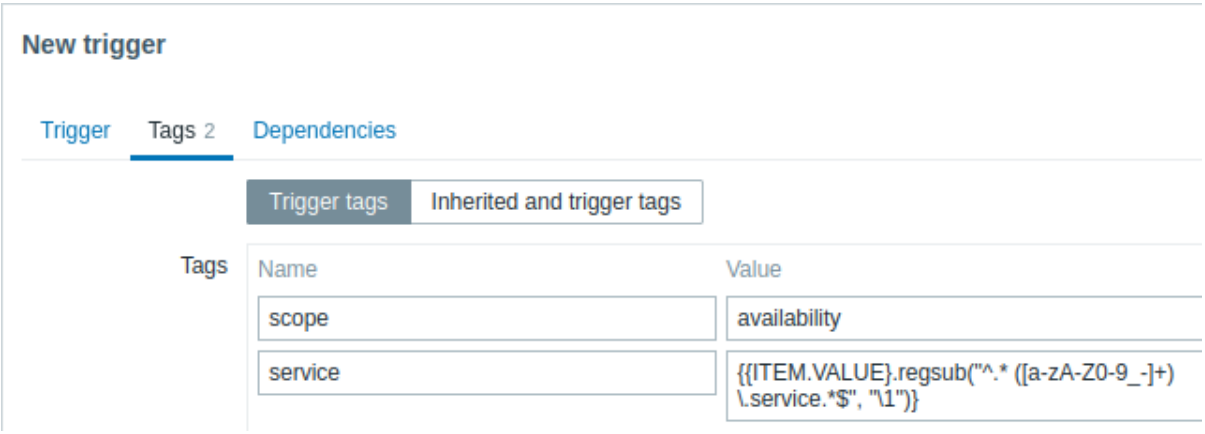
Поскольку мы этого не хотим, важно каким-то образом убедиться, что закрываются только нужные проблемы, а не просто все проблемы. Вот где могут помочь теги.

Проблемы и решения можно сопоставить, указав в настройках триггера тег. Необходимо выполнить следующие настройки:

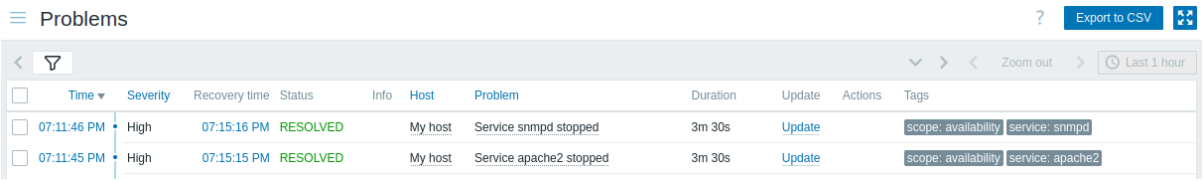
- *Режим генерации событий ПРОБЛЕМА*: Множественный
- *OK событие закрывает*: Все проблемы если значения тегов совпадают
- Введите имя тега для сопоставления событий




- настройка тегов для извлечения значений тегов из строк журнала




Если настройка выполнена успешно, вы сможете увидеть события о проблемах с тегами по приложению и сопоставление с их решением в *Мониторинг → Проблемы*.




**Warning:**
Посколько возможна некорректная настройка, когда аналогичные теги событий могут быть созданы по **не связанным** проблемам, пожалуйста, ознакомьтесь со случаями, которые описаны ниже!

- Индексированные макросы всегда ссылаются на поле *Выражение* конфигурации триггера, а не на *Выражение восстановления*. Например, в событии восстановления **{ITEM.VALUE1}** будет раскрываться в последнее значение первого элемента данных **в выражении проблемы** на момент восстановления. Если выражение восстановления основано на другом элементе данных, а значение элемента данных выражения проблемы изменится к моменту восстановления, события получат разные теги и не будут коррелированы.
- При наличии двух приложений, которые пишут сообщения об ошибках и восстановлениях в один и тот же файл журнала, пользователь может решить использовать два тега *service* в одном и том же триггере с разными значениями тегов с использованием отдельных регулярных выражений в значениях тегов, чтобы извлечь имена (скажем, служба A и служба B) из макроса {ITEM.VALUE} (например, когда форматы сообщений отличаются). Однако, такой подход может не сработать как планируется, если не будет соответствия регулярным выражениям. Не соответствующие регулярные выражения дадут пустые значения тегов, а одного пустого значения тега в событиях как проблемы, так и OK, будет достаточно, чтобы сработала корреляция. Таким образом, сообщение о восстановлении с приложения A может случайно закрыть сообщение об ошибке от приложения B.
- Фактические теги и значения тегов становятся видны только при срабатывании триггера. Если используемое регулярное выражение ошибочно, оно автоматически заменится на строку *НЕИЗВЕСТНО*. Если изначальное событие о проблеме с *НЕИЗВЕСТНО* пропущено, могут появиться последующие события OK с таким же значением тега *НЕИЗВЕСТНО*, которые могут закрыть события о проблеме, которые они не должны были бы закрывать.
- Если пользователь для значения тега использует макрос {ITEM.VALUE} без функций макросов, то будет применяться ограничение по длине строки в 255 символов. Когда в журнале имеются длинные сообщения и первые 255 символов не конкретизируют проблему, это может привести к одинаковым тегам событий по не связанным проблемам.

### 2. Глобальная корреляция событий

Обзор

Глобальная корреляция событий позволят охватить все метрики, наблюдаемые Zabbix, и создавать корреляции.

Имеется возможность сопоставления событий, которые были созданы совершенно разными триггерами, и применения одних и тех же операций к ним ко всем. Создавая интеллектуальные правила корреляции, можно обезопасить себя от тысяч повторяющихся оповещений и сфокусироваться на реальных причинах проблемы!

Глобальная корреляция событий — мощный механизм, который позволяет вам отвязать себя от логики проблемы и решения, основанной на одном триггере. До сих пор одно событие о проблеме создавалось одним триггером, и мы зависели от этого же триггера в плане решения проблемы. Мы не могли решить проблему, которая была создана одним триггером, при помощи другого триггера. Но с корреляцией событий на основе тегов событий — мы можем.

Например, триггер на журнал может сообщать о проблемах с приложением, в то время как триггер на основе опроса может сообщать, что приложение запущено и работает. Используя теги событий, вы можете пометить триггер на журнал как *состояние: недоступен*, тогда как тегом триггера на основе опроса будет *состояние: доступен*. Затем, в глобальном правиле корреляции вы можете соотнести эти триггеры и назначить соответствующую операцию на эту корреляцию, такую как закрытие старых событий.

В другом случае глобальная корреляция может распознавать похожие триггеры и применять к ним ту же операцию. Что если мы могли бы получать только одно сообщение о проблемах по каждому сетевому порту? Не требуется сообщать их все. Это также возможно при помощи глобальной корреляции событий.

Глобальная корреляция событий настраивается в **правилах корреляции**. Правило корреляции определяет, каким образом новые события о проблемах соотносятся с существующими событиями о проблемах, и что делать в случае совпадения (закрыть новое событие, закрыть старые события, сформировав соответствующие события OK). Если проблема закрыта при помощи глобальной корреляции, об этом сообщается в колонке *Инфо (Info)* в *Мониторинг (Monitoring)* → *Проблемы (Problems)*.

Настройка глобальных правил корреляции доступна только пользователям с уровнем Zabbix Супер-Администратор.

**Attention:**
Корреляцию событий необходимо настраивать очень осторожно, так как некорректная настройка может негативно сказаться на производительности обработки событий или может привести к закрытию большего количества событий, чем предполагалось (в худшем случае будут закрыты все события о проблемах).

Для **безопасной** настройки глобальной корреляции обратите внимание на следующие важные советы:

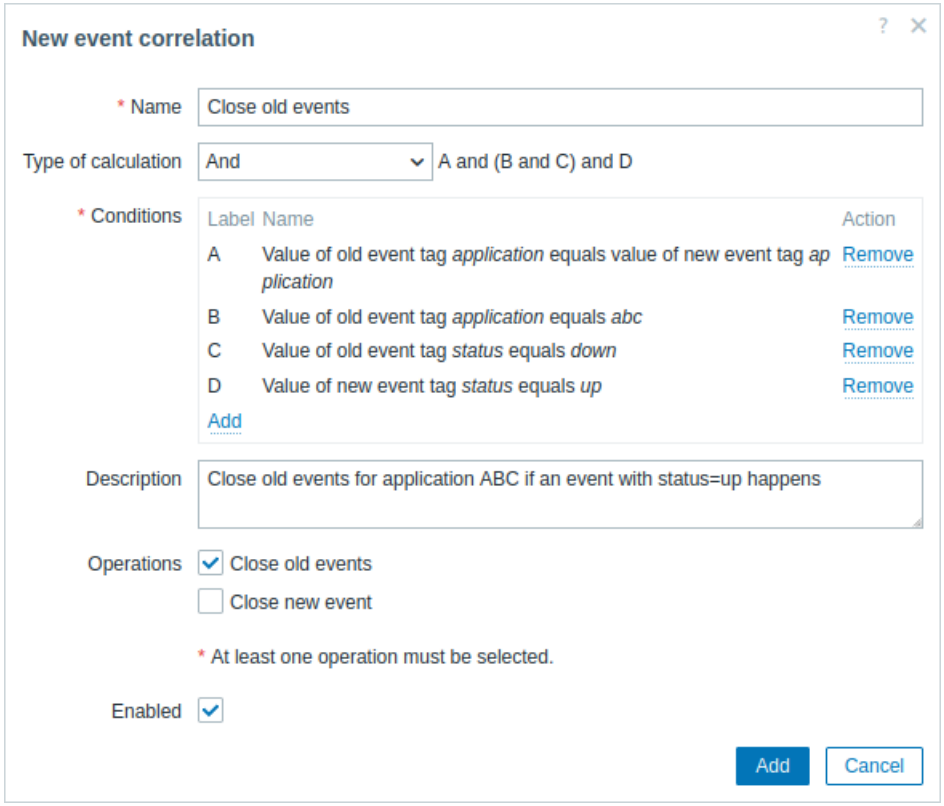
- Уменьшайте область действия корреляции. Всегда указывайте новому событию уникальный тег, который соотносится со старыми событиями, и используйте условие корреляции *Имя тега нового события (New event tag name)*.
- При использовании операции *Закрыть старые события (Close old events)* добавляйте явное условие для старых событий. Всегда добавляйте как минимум одно условие для *Старого события* (например, *Имя тега старого события (Old event tag name)*, *Значение тега старого события (Old event tag value)* или *Пара тегов событий (Event tag pair)*) при выборе параметра **Закрыть старые события (Close old events)** — в противном случае правило может сопоставить и закрыть несвязанные существующие события проблем (в худшем случае, все проблемы). Отдавайте предпочтение типу **Пара тегов событий (Event tag pair)** для сопоставления значений во время выполнения (хост:порт, идентификатор сессии и т. д.) и дополнительно сужайте поиск по узлу сети или группе узлов сети, если это возможно.
- Избегайте использования общих имён тегов, которые в итоге могут быть использованы разными настройками корреляций.
- Ограничьте количество правил корреляции теми, которые вам действительно необходимы.

Смотрите также: известные проблемы.

Конфигурация

Чтобы глобально настроить правила корреляции событий:

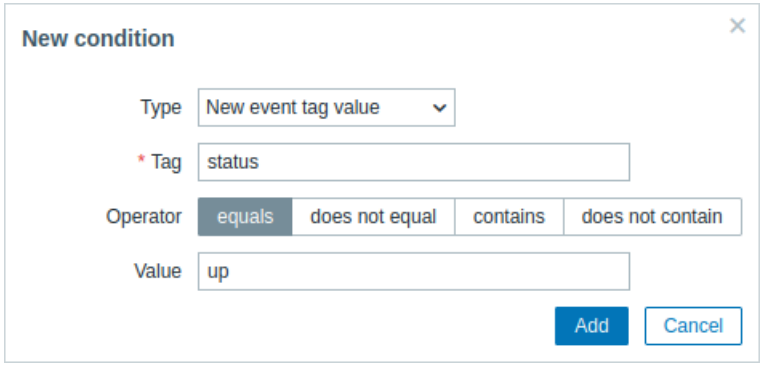
- Перейдите в *Сбор данных* > *Корреляция событий*
- Нажмите *Создать корреляцию событий* справа (или на имя корреляции, чтобы изменить существующее правило)
- Введите параметры правила корреляции в форме




Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Name* | Уникальное имя правила корреляции. |
| *Type of calculation* | Доступны следующие варианты вычисления условий:<br>**And** - должны быть выполнены все условия<br>**Or** - достаточно, чтобы было выполнено одно условие<br>**And/Or** - AND для разных типов условий и OR для одинакового типа условий<br>**Custom expression** - пользовательская формула вычисления для оценки условий действия. Она должна включать все условия (обозначенные заглавными буквами A, B, C, ...) и может содержать пробелы, табуляции, скобки ( ), **and** (с учетом регистра), **or** (с учетом регистра), **not** (с учетом регистра). |
| *Conditions* | Список условий. Подробности по настройке условия см. ниже. |
| *Description* | Описание правила корреляции. |
| *Operations* | Отметьте флажок операции, которую нужно выполнить, когда событие будет скоррелировано. Доступны следующие операции:<br>**Close old events** - закрыть старые события, когда происходит новое событие. Всегда добавляйте условие на основе старого события при использовании операции *Close old events*, иначе могут быть закрыты все существующие проблемы.<br>**Close new event** - закрыть новое событие, когда оно происходит.<br><br>**Warning!**<br>Не оставляйте условия старого/нового события пустыми при использовании *Close old events*/*Close new event*. Если вы выберете операцию *Close old events* без добавления условия, которое соответствует старому событию, Zabbix может сопоставить все существующие старые события и закрыть их. Всегда добавляйте явное условие старого события (например, *Old event tag name* или *Event tag pair*) при использовании *Close old events*. Например, правило, которое использует только *New event condition* и операцию *Close old events*, будет сопоставлять все старые события, которые соответствуют (отсутствующим) критериям старого события, - фактически закрывая старые проблемы. |
| *Enabled* | Если вы отметите этот флажок, правило корреляции будет включено. |

Чтобы настроить подробности нового условия, нажмите Add в блоке Conditions. Откроется всплывающее окно, в котором можно изменить параметры условия.



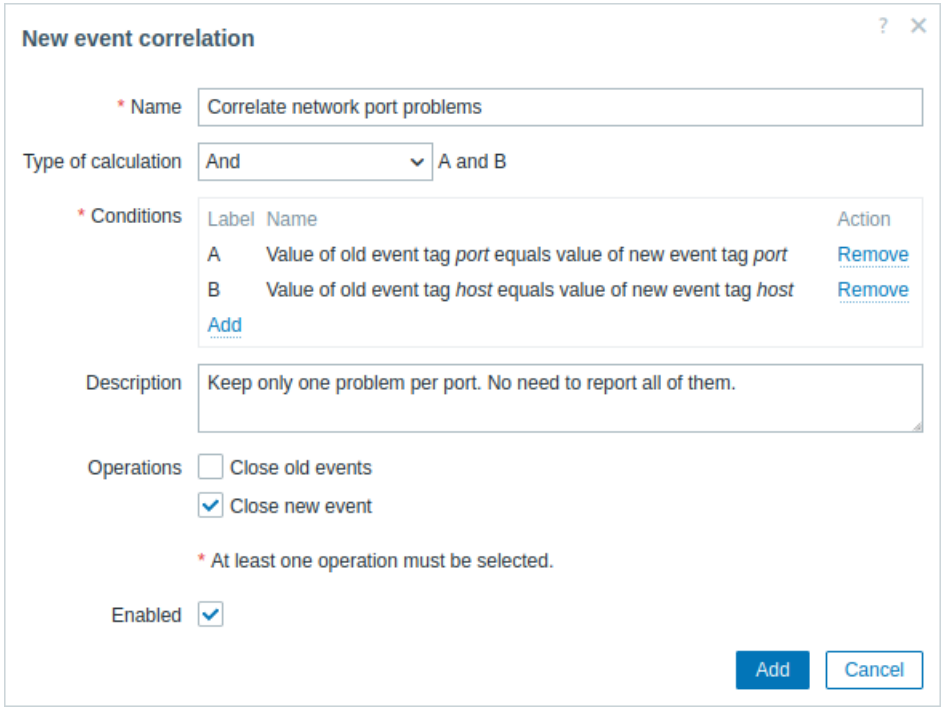

| Parameter | Description |
|---|---|
| *New condition* | Выберите условие для корреляции событий.<br>*Note* что если не указано условие старого события, могут быть сопоставлены и закрыты все старые события. Аналогично, если не указано условие нового события, могут быть сопоставлены и закрыты все новые события.<br>Доступны следующие условия:<br>**Old event tag name** - укажите имя тега старого события для сопоставления.<br>**New event tag name** - укажите имя тега нового события для сопоставления.<br>**New event host group** - укажите группу узлов сети нового события для сопоставления.<br>**Event tag pair** - укажите имя тега нового события и имя тега старого события для сопоставления. В этом случае совпадение будет, если совпадают **значения** тегов в обоих событиях. **Имена** тегов могут не совпадать.<br>Этот вариант полезен для сопоставления значений во время выполнения, которые могут быть неизвестны на момент настройки (см. также Example).<br>**Old event tag value** - укажите имя тега старого события и значение для сопоставления, используя следующие операторы:<br>*equals* - имеет значение тега старого события<br>*does not equal* - не имеет значения тега старого события<br>*contains* - содержит строку в значении тега старого события<br>*does not contain* - не содержит строку в значении тега старого события<br>**New event tag value** - укажите имя тега нового события и значение для сопоставления, используя следующие операторы:<br>*equals* - имеет значение тега нового события<br>*does not equal* - не имеет значения тега нового события<br>*contains* - содержит строку в значении тега нового события<br>*does not contain* - не содержит строку в значении тега нового события |

**Warning:**
Поскольку возможна неправильная настройка, а для **не связанных** проблем могут создаваться похожие теги событий, пожалуйста, ознакомьтесь со случаями, описанными ниже!

- Фактические теги и значения тегов становятся видимыми только при срабатывании триггера. Если используемое регулярное выражение некорректно, оно без предупреждения заменяется строкой *UNKNOWN*. Если исходное проблемное событие со значением тега *UNKNOWN* будет пропущено, могут появиться последующие события OK с тем же значением тега *UNKNOWN*, которые могут закрыть проблемные события, которые не должны были быть закрыты.
- Если пользователь использует макрос {ITEM.VALUE} без функций макросов в качестве значения тега, применяется ограничение в 255 символов. Когда сообщения журнала длинные, а первые 255 символов не содержат специфики, это также может привести к появлению похожих тегов событий для не связанных проблем.

Пример

Остановка повторяющихся событий о проблемах с одного и того же сетевого порта.




Такое глобальное правило корреляции сопоставит проблемы, если у триггера существуют значения тегов *host* и *port*, и они одинаковые как у оригинального события, так и у нового события.

Эта операция будет закрывать новые события о проблемах по тому же самому сетевому порту, оставив открытой только изначальную проблему.

**6. Тегирование**

Обзор

Теги состоят из имени тега и значения тега. При тегировании объектов можете использовать только имя или пару имени вместе со значением (например, `mysql`, `jira`, `target:mysql`, `service:jira` и т. д.).

Теги можно определить для различных объектов:

- шаблоны
- узлы сети
- элементы данных
- веб-сценарии
- триггеры
- услуги
- элементы данных и триггеры из шаблонов
- прототипы узлов сети, элементов данных и триггеров

**Note:**
Ознакомьтесь с официальными руководствами Zabbix для общих рекомендаций [en] по определению тегов, а также конкретными рекомендациями для шаблонов [en], элементов данных [en],триггеров [en] и правил низкоуровневого обнаружения [en].

Теги предназначены для нескольких целей, в основном, для маркировки событий. Когда объект помечен тегом, любое новое событие, относящееся к помеченному объекту, унаследует его теги. Например:

- для помеченных тегами шаблонов — любые проблемы узла сети (созданные триггерами из этого шаблона) унаследуют теги шаблона
- для помеченных тегами узлов сети — любая проблема узла сети унаследует теги узла сети
- для помеченных тегами элементов данных и веб-сценариев — любые проблемы по этому элементу данных или веб-сценарию унаследуют теги элемента данных или веб-сценария
- для помеченных тегами триггеров — любая проблема, созданная этим триггером, унаследует теги триггера

Событие о проблеме наследует все теги со всей цепочки объектов — шаблонов, узлов сети, элементов данных, веб-сценариев, триггеров. Идентичные комбинации : (после раскрытия макросов) объединяются в одну, что позволяет

избегать дублирования.

Пользовательские теги событий дают большую гибкость. Например:

- корреляция событий может быть настроена на основе тегов событий;
- условия действия могут быть настроены на основе тегов событий;
- проблемы элементов данных могут быть сгруппированы на основе тегов событий;
- теги проблем можно использовать для сопоставления проблем с услугами.

Объект можно пометить тегами с одинаковыми именами, но разными значениями (например, `component:memory` и `component:storage`). Аналогично, у объекта может быть тег без значения и тот же тег со значением (например, `database` и `database:postgresql`). Такие теги не считаются дубликатами.

Варианты использования

Некоторые распространенные варианты использования тегирования:

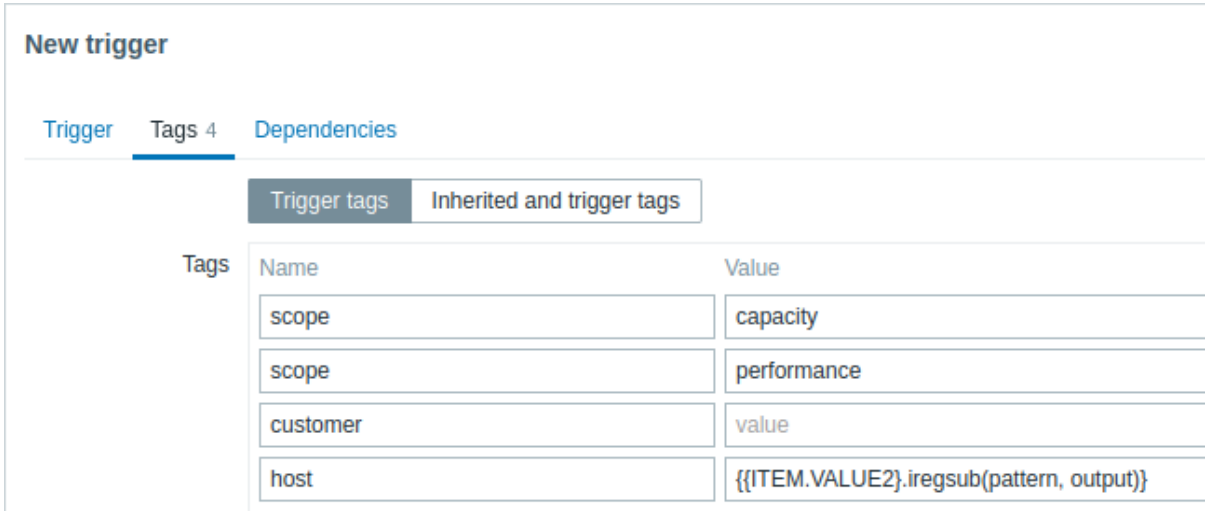
1. Пометка событий триггеров:
   - Определите тег триггера (например, `scope:performance`).
   - Проблемы, созданные этим триггером, будут иметь тег триггера.
2. Пометка проблем, унаследованных от шаблона:
   - Определите тег шаблона (например, `target:mysql`).
   - Проблемы, созданные триггерами из этого шаблона, будут иметь тег шаблона.
3. Пометка проблем узла сети:
   - Определите тег узла сети (например, `service:jira`).
   - Проблемы, созданные триггерами этого узла сети, будут иметь тег узла сети.
4. Фильтрация связанных элементов данных:
   - Определите тег элемента данных (например, `component:cpu`).
   - В *Monitoring* → *Latest data* элементы данных можно фильтровать по тегу `component:cpu`.
5. Использование информации, извлеченной из значения элемента данных, в качестве значения тега:
   - Определите тег с макросом в качестве значения тега (например, `tag-name:{{ITEM.VALUE<N>}.regsub()}` ).
   - В *Monitoring* → *Problems* у проблем значение тега будет преобразовано в данные, извлеченные из значения элемента данных.
6. Определение проблем в файле журнала и их отдельное закрытие:
   - Определите тег триггера для триггера элемента данных мониторинга журнала, который будет извлекать значения из значения элемента данных с помощью макроса (например, `service:{{ITEM.VALUE<N>}.regsub()}` ).
   - В конфигурации триггера настройте корреляцию событий:
     - установите для *PROBLEM event generation mode* значение "Multiple";
     - установите для *OK event closes* значение "All problems if tag values match";
     - задайте тег для сопоставления.
   - Проблемы, созданные триггером элемента данных журнала, будут иметь тег триггера и будут закрываться по отдельности.
7. Фильтрация уведомлений:
   - Определите теги триггеров (например, `scope:security` для trigger1 и `scope:availability` для trigger2).
   - Используйте фильтрацию по тегам в условиях действий, чтобы получать уведомления только по событиям, соответствующим данным тегов.
8. Определение проблем в уведомлениях:
   - Определите теги триггеров.
   - Используйте макрос {EVENT.TAGS} в уведомлении о проблеме.
   - Уведомление о проблеме будет содержать теги триггера, что позволит легче определить, какому приложению/службе принадлежит уведомление.
9. Упрощение задач конфигурации с помощью тегов шаблона:
   - Определите тег триггера шаблона.
   - Триггеры, созданные из этого триггера шаблона, будут иметь его тег.
10. Создание триггеров с тегами из низкоуровневого обнаружения (LLD):
    - Определите тег прототипа триггера с макросом LLD в имени или значении тега (например, `scope:{#FSNAME}`).
    - Триггеры, созданные из прототипа триггера, будут иметь его тег.
11. Сопоставление сервисов с использованием тегов сервисов:
    - Определите теги сервисов.
    - Настройте действия сервисов для сервисов с совпадающими тегами.
    - Кроме того, используйте теги сервисов для привязки сервиса к SLA для расчетов SLA.
12. Связывание сервисов с проблемами с использованием тегов проблем сервисов:
    - Определите тег проблемы в конфигурации сервиса (например, `target:mysql`).

- Проблемы с совпадающим тегом будут автоматически сопоставлены с сервисом, а статус сервиса изменится в соответствии с настроенными правилами расчета статуса сервиса.

13. Подавление проблем, когда узел сети находится в режиме обслуживания:
    - Определите теги в конфигурации периода обслуживания.
    - Проблемы с заданными тегами будут подавляться.
14. Предоставление доступа группам пользователей:
    - Определите теги в конфигурации группы пользователей.
    - Пользователи в группе пользователей смогут просматривать только проблемы с заданными тегами.

Настройка

Теги можно указать на отдельной вкладке, например, в настройках триггера:




Поддержка макросов

Встроенные и пользовательские макросы в тегах определяются в момент возникновения события. Пока событие не произошло, эти макросы будут отображаться в веб-интерфейсе Zabbix как неразрешенные.

Макросы низкоуровневого обнаружения определяются в процессе обнаружения.

Следующие макросы могут использоваться в именах и значениях тегов триггера:

- встроенные макросы {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} и {HOST.ID}
- встроенные макросы {INVENTORY.*} (для ссылки на значения инвентаризации узла сети с одного или нескольких узлов сети в выражении триггера)
- пользовательские макросы и пользовательские макросы с контекстом (контекст может включать макросы низкоуровневого обнаружения)
- макросы низкоуровневого обнаружения (только в тегах прототипа триггера)

Следующие макросы могут использоваться в именах и значениях тегов шаблона, узла сети и элемента данных/сценария web:

- встроенные макросы {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} и {HOST.ID}
- встроенные макросы {INVENTORY.*}
- пользовательские макросы
- макросы низкоуровневого обнаружения (только в тегах прототипа узла сети и прототипа элемента данных)

Следующие макросы могут использоваться в уведомлениях на основе триггера:

- встроенные макросы {EVENT.TAGS} и {EVENT.RECOVERY.TAGS} (эти макросы будут преобразованы в список тегов события или тегов события восстановления, разделенный запятыми)
- встроенные макросы {EVENT.TAGSJSON} и {EVENT.RECOVERY.TAGSJSON} (эти макросы будут преобразованы в массив JSON, содержащий объекты тегов события objects или объекты тегов события восстановления)

Извлечение подстроки в тегах к триггерам

Поддерживается извлечение подстроки для заполнения имени тега или значения тега при использовании функции макросов. Функция применяет регулярное выражение к значению, полученному поддерживаемым макросом. Например:

```
{{ITEM.VALUE}.regsub(    ,    )}
{{ITEM.VALUE}.iregsub(    ,    )}
```

```
{{#LLDMACRO}.regsub(    ,    )}
{{#LLDMACRO}.iregsub(    ,    )}
```

Имя тега и значение будут обрезаны до 255 символов, если их длина превысит 255 символов после раскрытия макросов.

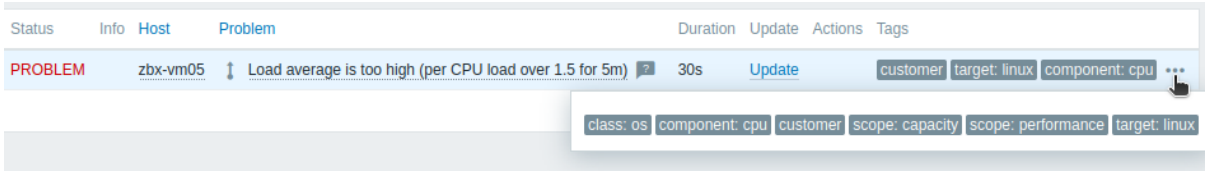
Смотрите также: Использование функций макросов в макросах низкоуровневого обнаружения для маркировки событий.

Просмотр тегов событий

Маркировку, если задана, можно увидеть вместе с новыми событиями в:

- *Мониторинг* → *Проблемы*
- *Мониторинг* → *Проблемы* → *Детали события*
- *Панели* → виджет *Проблемы*

Порядок и количество отображаемых тегов определяется опциями *Приоритет отображения тегов (Tag display priority)* и *Отображать теги (Show tags)* в разделе *Мониторинг* → *Проблемы* либо виджете панели *Проблемы*. Обратите внимание, что можно отобразить максимум три тега; если тегов больше, то при наведении указателя мыши на многоточие все теги отобразятся во всплывающем окне.




## 7. Визуализация

Обзор

Визуализация предоставляет различные способы отображения данных мониторинга, собранных Zabbix. Графики показывают динамику изменений и метрики, карты сети отображают топологию инфраструктуры, а панели объединяют настраиваемые виджеты для предоставления всестороннего представления данных мониторинга.

Инструменты визуализации

- Графики — просматривайте значения элементов данных на простых автоматически сгенерированных графиках, создавайте собственные графики или сравнивайте несколько элементов данных в ситуационных графиках.
- Карты сети — создавайте визуальные представления сетевой инфраструктуры с настраиваемыми элементами и индикаторами.
- Панели — настраивайте макеты, комбинируя различные виджеты панели, которые визуализируют данные в разных форматах.

### 1. Графики

Обзор

При наличии большого потока данных, поступающих в Zabbix, для пользователей намного проще, если они могут взглянуть на визуальное представление о происходящем, а не только на числа.

Именно здесь на помощь приходят графики. Графики позволяют охватить поток данных с одного взгляда, соотнести проблемы, обнаружить, когда что-то началось, или наглядно показать, что что-то может стать проблемой.

Zabbix предоставляет пользователям:

- встроенные простые графики по данным одного элемента данных
- возможность создавать более сложные пользовательские графики
- доступ к быстрому сравнению нескольких элементов данных с использованием ситуационных графиков
- современные настраиваемые векторные графики и круговые диаграммы

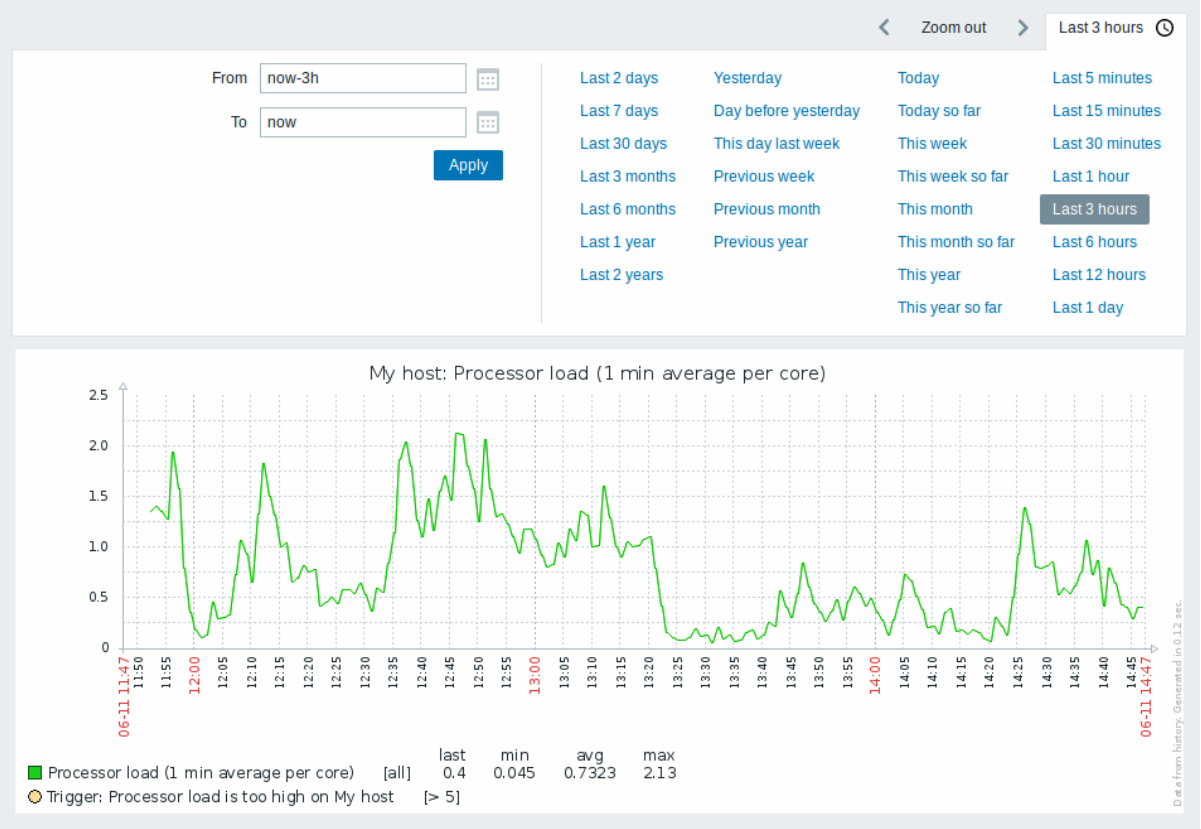
#### 1. Простые графики

Обзор

Простые графики предоставляются для визуализации данных, собранных элементами данных.

Для просмотра простых графиков от пользователя не требуется никаких действий по настройке. Они свободно предоставляются Zabbix.

Просто перейдите в *Monitoring* > *Latest data* и нажмите ссылку Graph для соответствующего элемента данных, после чего будет отображен график.




**Note:**
Простые графики предоставляются для всех числовых элементов данных. Для текстовых элементов данных в *Monitoring* > *Latest data* доступна ссылка History.

Селектор периода времени

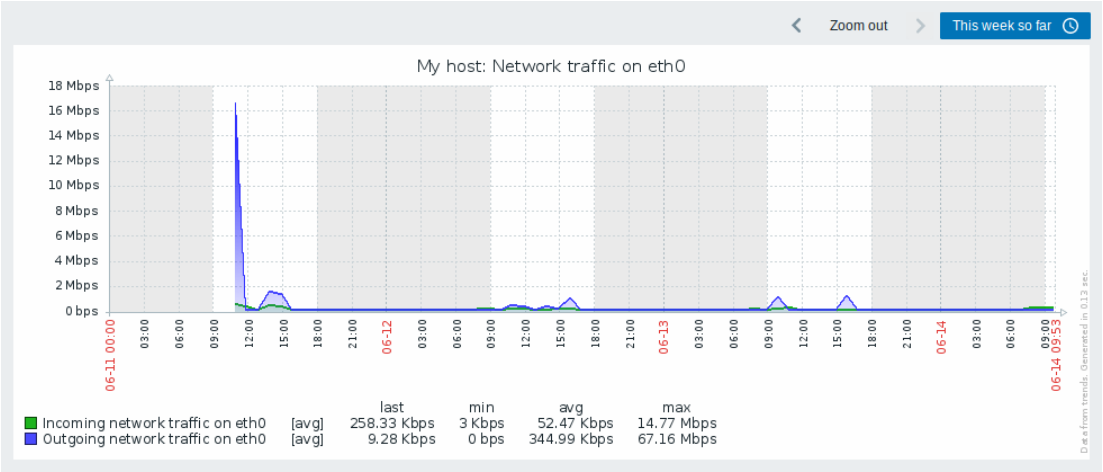
Селектор *Период времени* над графиком позволяет выбирать наиболее популярные периоды одним щелчком мыши. Для получения дополнительной информации смотрите *Селектор периода времени*.

Недавние данные vs длительные периоды

Для самых последних данных рисуется **одиночная** линия, соединяющая каждое полученное значение. Одиночная линия рисуется до тех пор, пока доступен хотя бы один горизонтальный пиксель для одного значения.

Для данных, показывающих длительный период, рисуется **три линии** — тёмно-зелёная линия отображает среднее значение, тогда как светло-розовая и светло-зелёная линии отображают максимальное и минимальное значения в данный момент времени. Пространство между максимумами и минимумами заполняется жёлтым цветом.

Рабочее время (рабочие дни) отображается на графиках белым фоном, а нерабочее время отображается серым (в теме *Оригинальная синяя*, по умолчанию в веб-интерфейсе).

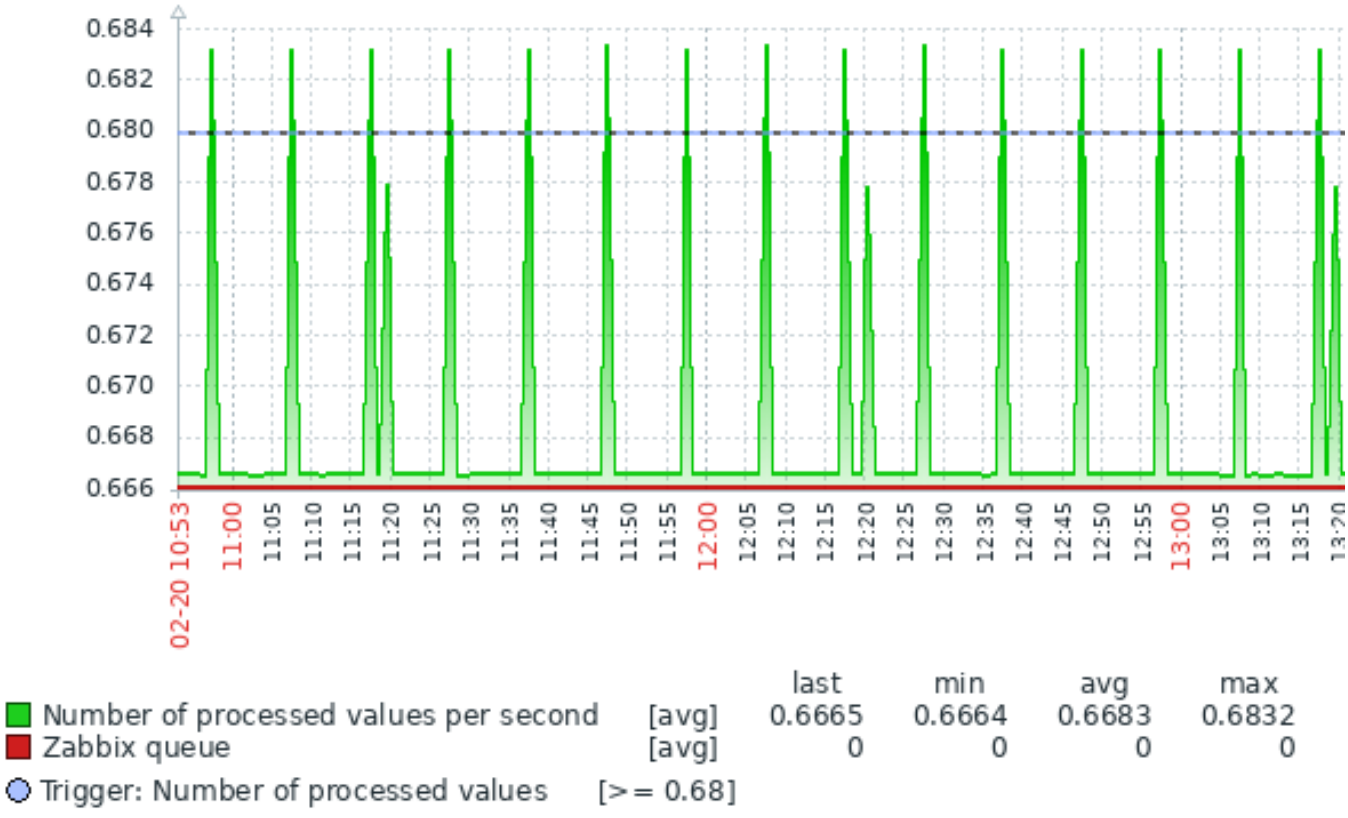



Рабочее время всегда отображается на простых графиках, тогда как отображение рабочего времени на пользовательских графиках является предпочтением пользователя.

Рабочее время не отображается, если график показывает период более 3 месяцев.

Линии триггеров

Простые триггеры отображаются линиями с чёрным пунктиром поверх цвета важности триггера — обратите внимание на синюю линию на графике и на информацию о триггере, отображаемую в легенде. На графике может отображаться до 3 линий триггеров; если триггеров больше, тогда приоритет отдается триггерам с меньшей важностью. В простых графиках триггеры отображаются всегда, в то время как отображение триггеров в пользовательских графиках является предпочтением пользователя.




Формирование на основе истории/трендов

Графики могут строиться на основе либо истории, либо трендов элемента данных.

Для пользователей, у которых включен режим отладки веб-интерфейса, в правом нижнем углу графика отображается серная вертикальная подпись, указывающая, откуда поступают данные.

На то, будет ли использоваться история или тренды, влияют несколько факторов:

- срок хранения истории элемента данных. Например, история элемента данных может храниться 14 дней. В этом случае любые данные старше четырнадцати дней будут браться из трендов.

- плотность данных на графике. Если количество секунд, отображаемых в одном горизонтальном пикселе графика, превышает 3600/16, отображаются данные трендов (даже если история элемента данных по-прежнему доступна за тот же период).
- если тренды отключены, для построения графика используется история элемента данных — если она доступна за этот период.

Отсутствие данных

Для элементов данных с обычным интервалом обновления на графике ничего не отображается, если данные элемента не собираются.

Однако для элементов данных типа trapper и элементов данных с заданным запланированным интервалом обновления (при этом обычный интервал обновления установлен в 0) проводится прямая линия до первого собранного значения и от последнего собранного значения до конца графика; линия проходит соответственно на уровне первого/последнего значения.

Переключение к сырым данным

Выпадающее меню в верхнем правом углу выше графика позволяет переключаться с простого графика на список *Значения/500 последних значений (Values/500 latest values)*. Эта функция может быть полезна для просмотра числовых значений, создающих график.

Представленные здесь значения — «сырые», то есть без единиц измерения и без постобработки значений. Однако, преобразования значений здесь также применяются.

Известные проблемы

Смотрите известные проблемы по графикам.

### 2. Пользовательские графики

Обзор

Пользовательские графики, как следует из названия, предлагают возможность индивидуальной настройки.

В то время как простые графики хороши для просмотра данных по одному элементу данных, они не дают возможности индивидуальной настройки.

Таким образом, если вы захотите изменить стиль графика или способ отрисовки линий, или сравнить несколько элементов данных, например, входящий и исходящий трафик на одном графике, то вам необходим пользовательский график.

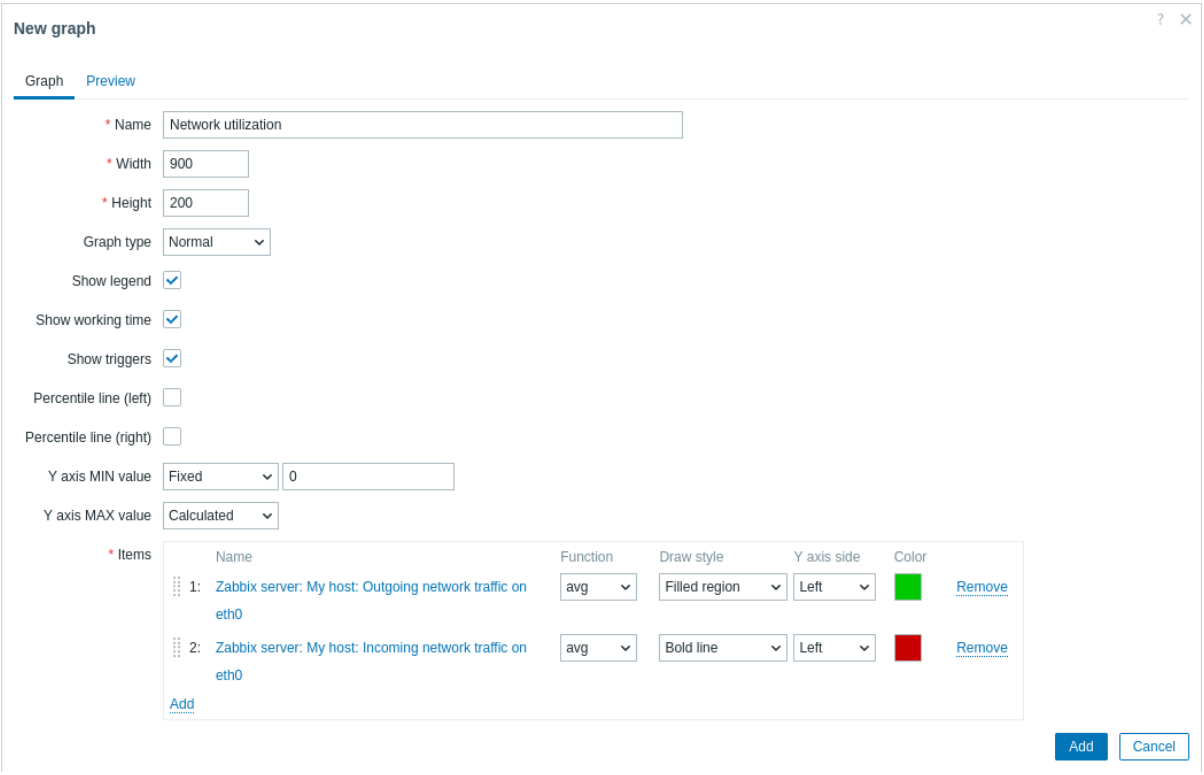
Пользовательские графики настраиваются вручную.

Они могут быть созданы для одного узла сети, для нескольких узлов сети или для отдельного шаблона.

Настройка пользовательских графиков

Чтобы создать пользовательский график, выполните следующие действия:

- Перейдите в *Сбор данных > Узлы сети (или шаблоны)*
- Нажмите *Графики* в строке рядом с нужным узлом сети или шаблоном
- На экране графиков нажмите *Создать график*
- Отредактируйте атрибуты графика




Все обязательные поля ввода отмечены красной звездочкой.

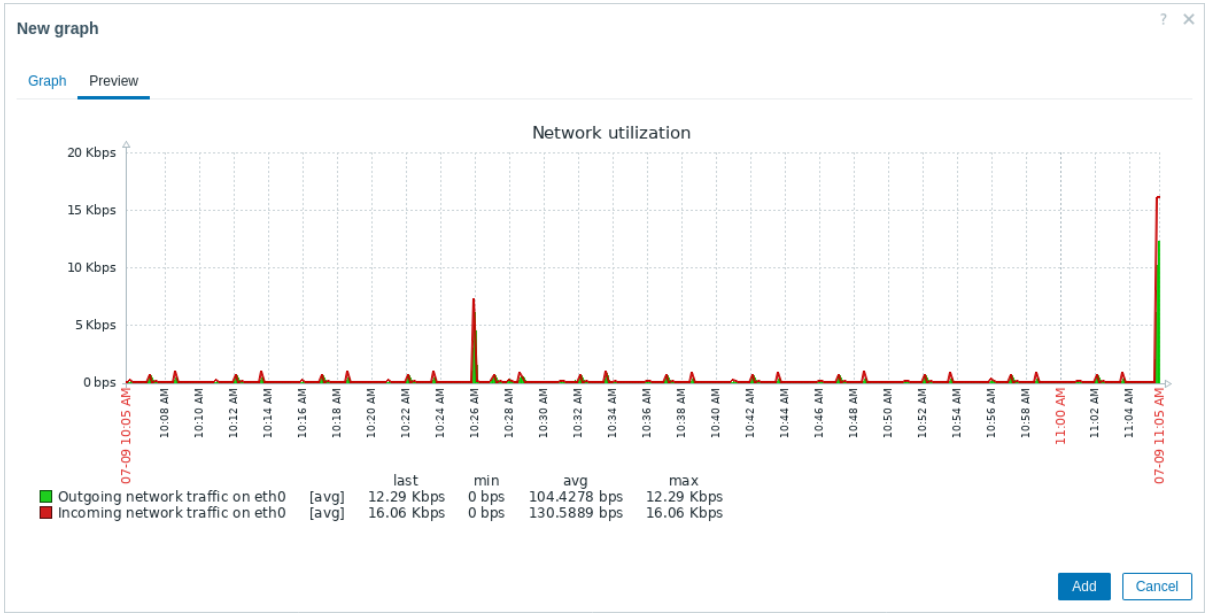
Атрибуты графика:

| Параметр | Описание |
|---|---|
| *Имя* | Уникальное имя графика.<br>В этом поле поддерживаются выражения макросов, но только с функциями `avg`, `last`, `min` и `max` и с временем в качестве параметра (например, `{?avg(/host/key,1h)}`). Макросы {HOST.HOST<1-9>} поддерживаются для использования внутри этого макроса, ссылаясь на первый, второй, третий и т. д. узел сети в графике, например `{?avg(/{HOST.HOST2}/key,1h)}`. Обратите внимание, что ссылка на первый узел сети с помощью этого макроса избыточна, так как на первый узел сети можно сослаться неявно, например `{?avg(//key,1h)}`. |
| *Ширина* | Ширина графика в пикселях (только для предварительного просмотра и круговых/развернутых графиков). |
| *Высота* | Высота графика в пикселях. |
| *Тип графика* | Тип графика:<br>**Normal** - обычный график, значения отображаются в виде линий<br>**Stacked** - накопительный график, отображаются заполненные области<br>**Pie** - круговой график<br>**Exploded** - "развернутый" круговой график, доли отображаются как "вырезанные" из круга |
| *Показывать легенду* | Если этот флажок установлен, будет отображаться легенда графика. |
| *Показывать рабочее время* | Если выбрано, нерабочие часы будут отображаться на сером фоне. Этот параметр недоступен для круговых и развернутых круговых графиков. |
| *Показывать триггеры* | Если выбрано, простые триггеры будут отображаться в виде линий с черными штрихами поверх цвета уровня серьезности триггера. Этот параметр недоступен для круговых и развернутых круговых графиков. |
| *Линия процентиля (левая)* | Отображать процентиль для левой оси Y. Если, например, задан 95% процентиль, линия процентиля будет находиться на уровне, ниже которого располагаются 95 процентов значений. Отображается ярко-зеленой линией. Доступно только для обычных графиков. |
| *Линия процентиля (правая)* | Отображать процентиль для правой оси Y. Если, например, задан 95% процентиль, линия процентиля будет находиться на уровне, ниже которого располагаются 95 процентов значений. Отображается ярко-красной линией. Доступно только для обычных графиков. |

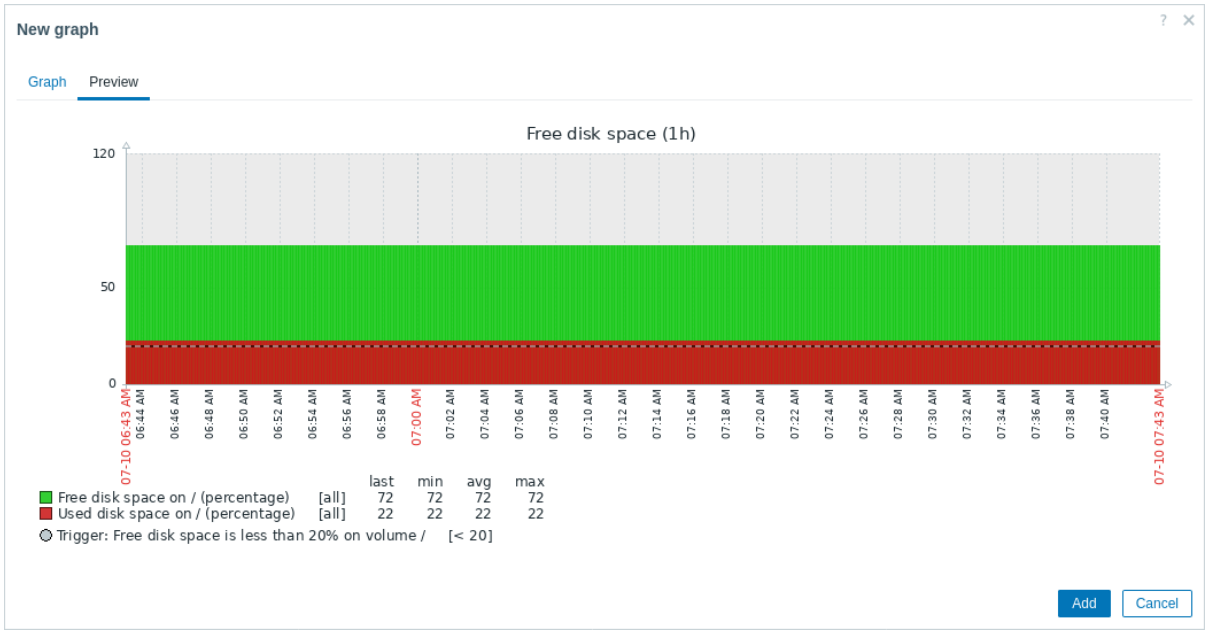
| Параметр | Описание |
|---|---|
| *MIN значение оси Y* | Минимальное значение оси Y:<br>**Calculated** - минимальное значение оси Y будет вычислено автоматически.<br>**Fixed** - фиксированное минимальное значение оси Y.<br>**Item** - последнее значение выбранного элемента данных будет минимальным значением.<br><br>Этот параметр недоступен для круговых и развернутых круговых графиков. |
| *MAX значение оси Y* | Максимальное значение оси Y:<br>**Calculated** - максимальное значение оси Y будет вычислено автоматически.<br>**Fixed** - фиксированное максимальное значение оси Y.<br>**Item** - последнее значение выбранного элемента данных будет максимальным значением<br><br>Этот параметр недоступен для круговых и развернутых круговых графиков. |
| *3D вид* | Включить 3D-стиль. Только для круговых и развернутых круговых графиков. |
| *Элементы данных* | Элементы данных, данные которых должны отображаться на этом графике. Нажмите *Добавить*, чтобы выбрать элементы данных. Также можно выбрать различные параметры отображения (функция, стиль рисования, отображение на левой/правой оси, цвет). |
| *Порядок сортировки (0→100)* | Порядок отрисовки. 0 будет обработан первым. Можно использовать для отображения линий или областей позади (или перед) другими.<br>Вы можете перетаскивать элементы с помощью значка в начале строки, чтобы задать порядок сортировки или определить, какой элемент отображается перед другим. |
| *Имя* | Имя выбранного элемента данных отображается как ссылка. При нажатии на ссылку открывается список других доступных элементов данных. |
| *Тип* | Тип (доступно только для круговых и развернутых круговых графиков):<br>**Simple** - значение элемента данных отображается пропорционально на круговой диаграмме<br>**Graph sum** - значение элемента данных представляет весь круговой график<br>Обратите внимание, что цвет элемента ”graph sum” будет виден только в той мере, в какой он не занят элементами ”proportional”. |
| *Функция* | Выберите, какие значения будут отображаться, если для одного элемента данных существует более одного значения на вертикальный пиксель графика:<br>**all** - отображать все возможные значения (минимум, максимум, среднее) на графике. Обратите внимание, что для более коротких периодов этот параметр не влияет; только для более длительных периодов, когда плотность данных в одном вертикальном пикселе графика увеличивается, ’all’ начинает отображать минимальные, максимальные и средние значения. Эта функция доступна только для типа графика *Normal*. См. также: Создание графиков из истории/трендов.<br>**avg** - отображать средние значения<br>**last** - отображать последние значения. Эта функция доступна только если в качестве типа графика выбран *Pie/Exploded pie*.<br>**max** - отображать максимальные значения<br>**min** - отображать минимальные значения |
| *Стиль рисования* | Выберите стиль рисования (доступно только для обычных графиков; для накопительных графиков всегда используется заполненная область) для применения к данным элемента - *Line*, *Bold line*, *Filled region*, *Dot*, *Dashed line*, *Gradient line*. |
| *Сторона оси Y* | Выберите сторону оси Y, на которой будут отображаться данные элемента - *Left*, *Right*. |
| *Цвет* | Выберите цвет, который будет применен к данным элемента. |

Предпросмотр графика

На вкладке *Предпросмотр (Preview)* отображается предпросмотр графика, так что вы сразу сможете увидеть, что создаёте.




Обратите внимание, что предварительный просмотр не будет отображать какие-либо данные по элементам данных из шаблона.




В этом примере, обратите внимание на жирную пунктирную линию, которая отображает уровень триггера, и на информацию о триггере, которая отображается в легенде.

**Note:**
Отображается не более 3 линий триггеров. Если триггеров больше, приоритет отображения отдается триггерам с более низкой важностью.<br><br> Если высота графика меньше 120 пикселей, то триггеры в легенде к графику отображаться не будут.

### 3. Ситуационные графики

Обзор

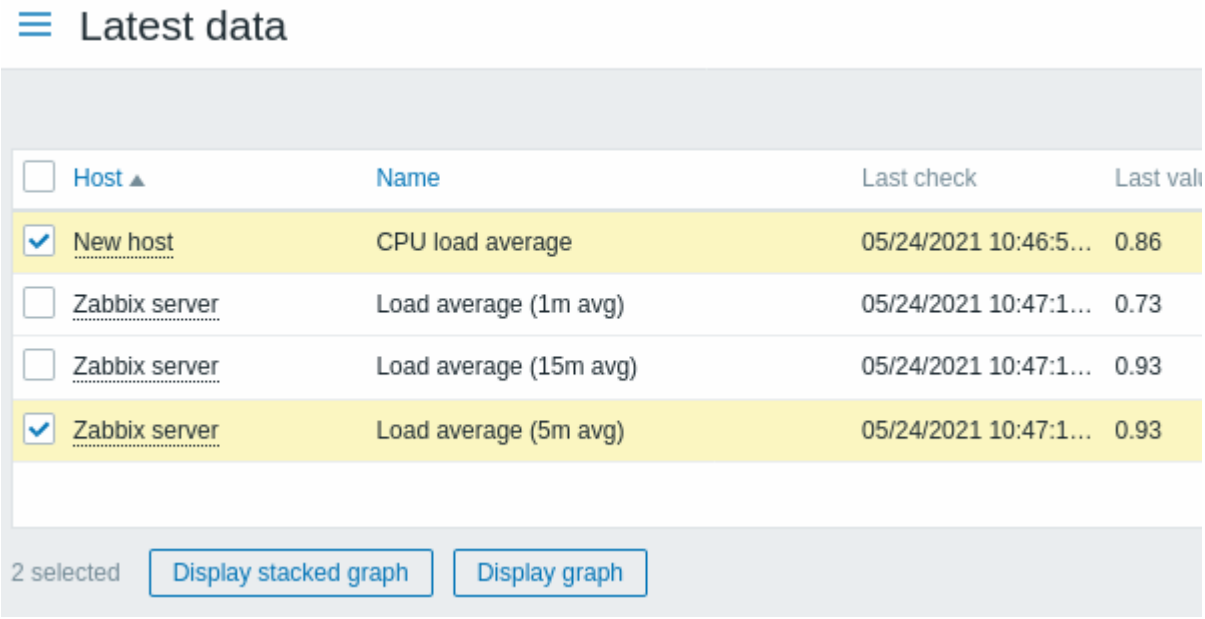
В то время как простой график — отличное средство доступа к данным одного элемента данных, а пользовательские графики предлагают варианты настройки, ни один их них не позволяет быстро создавать графики для сравнения нескольких элементов данных с минимальными усилиями и без последующей поддержки этого графика.

С целью исправить эту проблему, имеется возможность создавать ситуационные графики по нескольким элементам данных на лету.

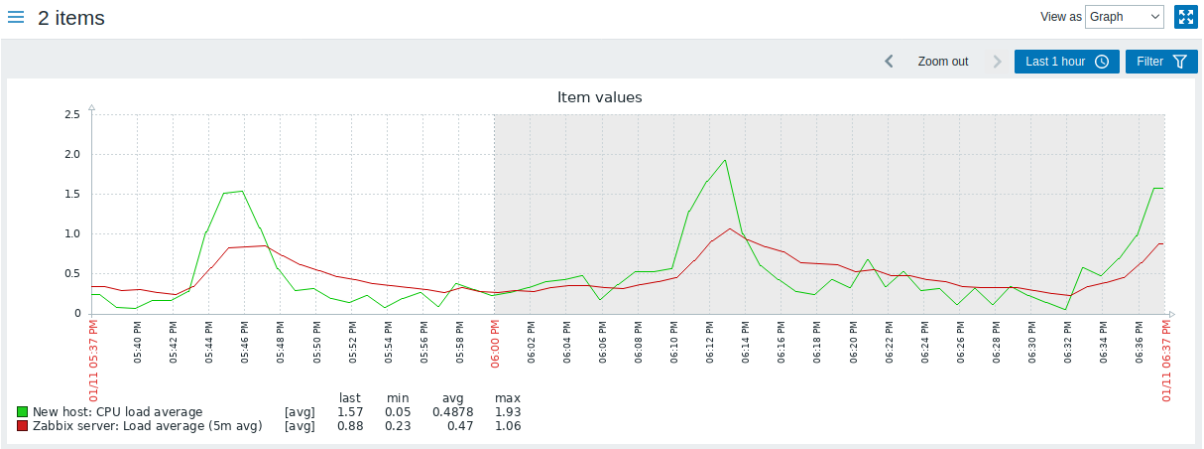
Конфигурация

Чтобы создать ad-hoc график, выполните следующие действия:

- Перейдите в *Monitoring* → *Latest data*
- Используйте фильтр, чтобы отобразить нужные вам элементы данных
- Отметьте флажками элементы данных, которые вы хотите отобразить на графике
- Нажмите кнопки *Display stacked graph* или *Display graph*

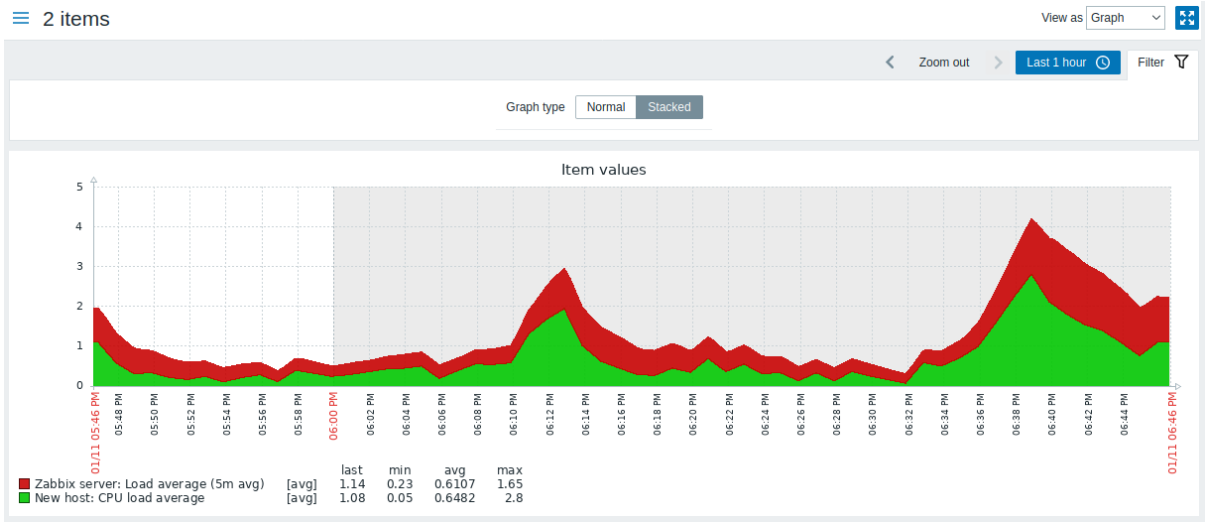



Ваш график будет создан мгновенно:




Обратите внимание: чтобы не отображать слишком много линий на графике, показывается только среднее значение для каждого элемента данных (линии минимального/максимального значения не отображаются). Триггеры и информация о триггерах на графике не отображаются.

В окне созданного графика доступен *Time period* selector, а также возможность переключаться с обычного линейного графика на stacked-график и обратно.




### 2. Карты сети

Обзор

Если у вас есть сеть, за которой вы присматриваете, то вы, возможно, захотите где-либо иметь обзор вашей инфраструктуры. Для этого вы можете создать карты в Zabbix — сетей или всего того, чего пожелаете.

Все пользователи могут создавать карты сети. Карты могут быть публичными (доступными всем пользователям) или приватными (доступными выбранным пользователям).

Перейдём к настройке карты сети.

### 1. Настройка карты сети

Обзор

Настройка карты в Zabbix требует, чтобы сначала вы создали карту, задав ее общие параметры, а затем начали заполнять саму карту элементами и их связями.

Вы можете заполнить карту элементами, которые представляют собой узел сети, группу узлов сети, триггер, изображение или другую карту.

Для представления элементов карты используются значки. Вы можете определить информацию, которая будет отображаться рядом со значками, и задать особый способ отображения недавних проблем. Вы можете связать значки и определить информацию, которая будет отображаться на связях.

Вы можете добавлять пользовательские URL-адреса, доступные при щелчке по значкам. Таким образом, вы можете связать значок узла сети со свойствами узла сети или значок карты с другой картой.

Количество проблем на картах отображается только для первопричинных проблем.

Карты управляются в *Monitoring* > *Maps*, где их можно настраивать, управлять ими и просматривать. В представлении мониторинга вы можете щелкать значки и использовать ссылки на некоторые скрипты и URL-адреса.

Сетевые карты основаны на векторной графике (SVG).

Публичные и приватные карты

Все пользователи Zabbix (включая пользователей не администраторов) могут создавать карты сети. Карты сети имеют владельца — пользователя, который их создал. Карты сети можно сделать публичными или приватными.

- *Публичные (Public)* карты видны всем пользователям; однако, пользователи должны иметь права на чтение по крайней мере одного элемента карты сети, чтобы её увидеть. Публичные карты можно редактировать, если пользователь / группа пользователей имеет права чтения-записи на эту карту и по крайней мере права на чтение всех элементов на соответствующей карте, включая триггеры в связях.
- *Приватные (Private)* карты видны только своим владельцам и пользователям/группам пользователей, с которыми поделился владелец карты. Обычные пользователи (не Супер-администраторы) могут назначать общий доступ только тем группам, в которые они входят сами, и пользователям этих групп. Пользователи с уровнем Администратор могут видеть приватные карты вне зависимости от того, являются ли они их владельцами или принадлежат ли списку пользователей, с которыми поделились картой. Приватные карты могут редактироваться

владельцами и в случае, если у пользователя/группы пользователей имеются права чтения-записи на карту, а также хотя бы права на чтение всех элементов соответствующей карты, включая триггеры в связях.

Элементы карты, к которым у пользователя отсутствует доступ на чтение, отображаются с серой иконкой, и вся текстовая информация по элементу скрыта. Однако, подпись к триггеру остаётся видимой, даже если у пользователя нет прав на триггер.

Для добавления элемента на карту у пользователя также должны быть права как минимум на чтение этого элемента.

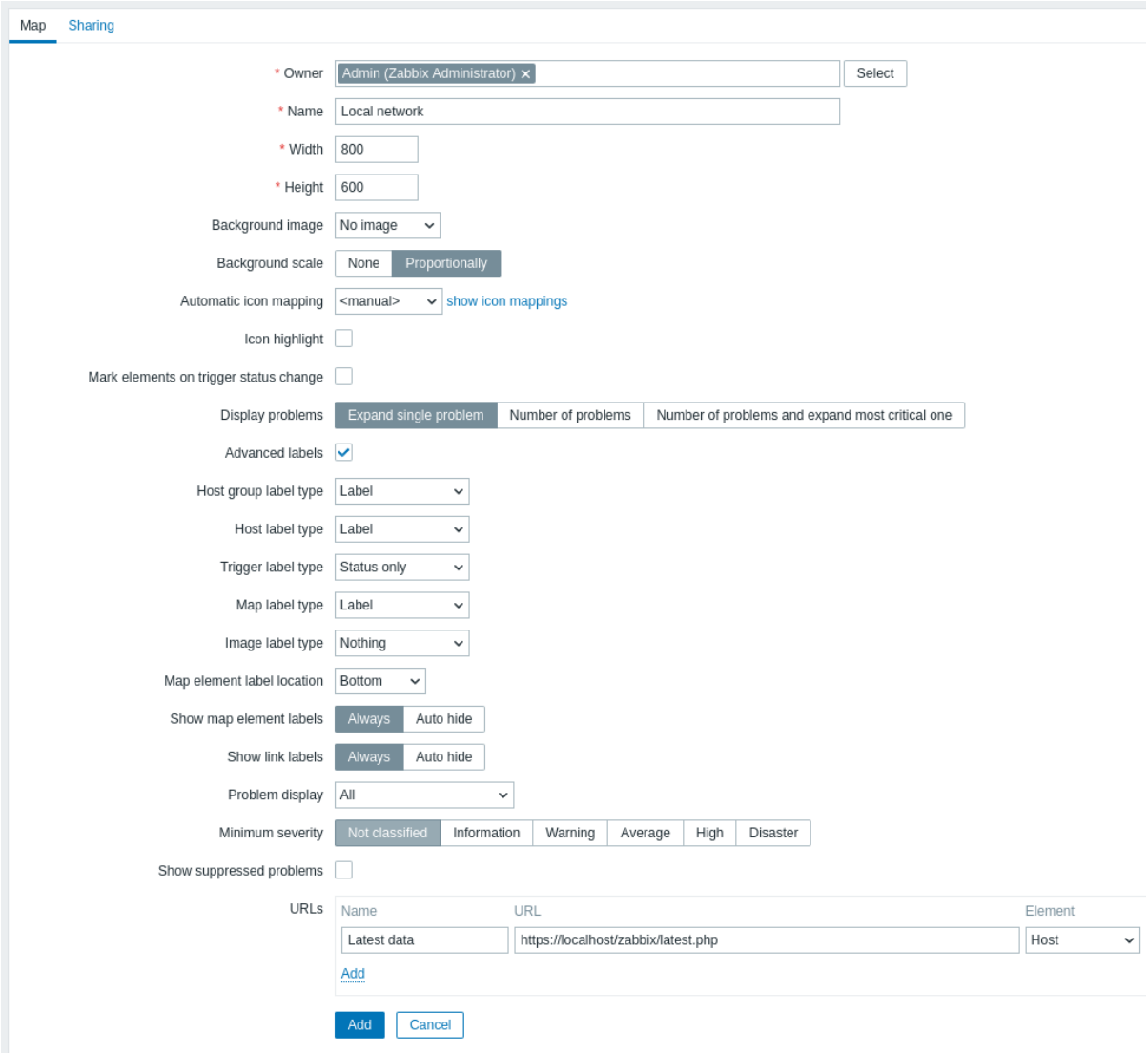
Создание карты

Чтобы создать карту, выполните следующие действия:

1. Перейдите в *Monitoring* > *Maps*.
2. Перейдите в представление со всеми картами.
3. Нажмите *Create map*.

Вы также можете использовать кнопку *Clone* в форме настройки существующей карты, чтобы создать новую карту. Эта карта будет иметь все свойства существующей карты, включая общие атрибуты макета, а также элементы существующей карты.

Вкладка **Map** содержит общие атрибуты карты:




Все обязательные поля ввода отмечены красной звездочкой.

Общие атрибуты карты:

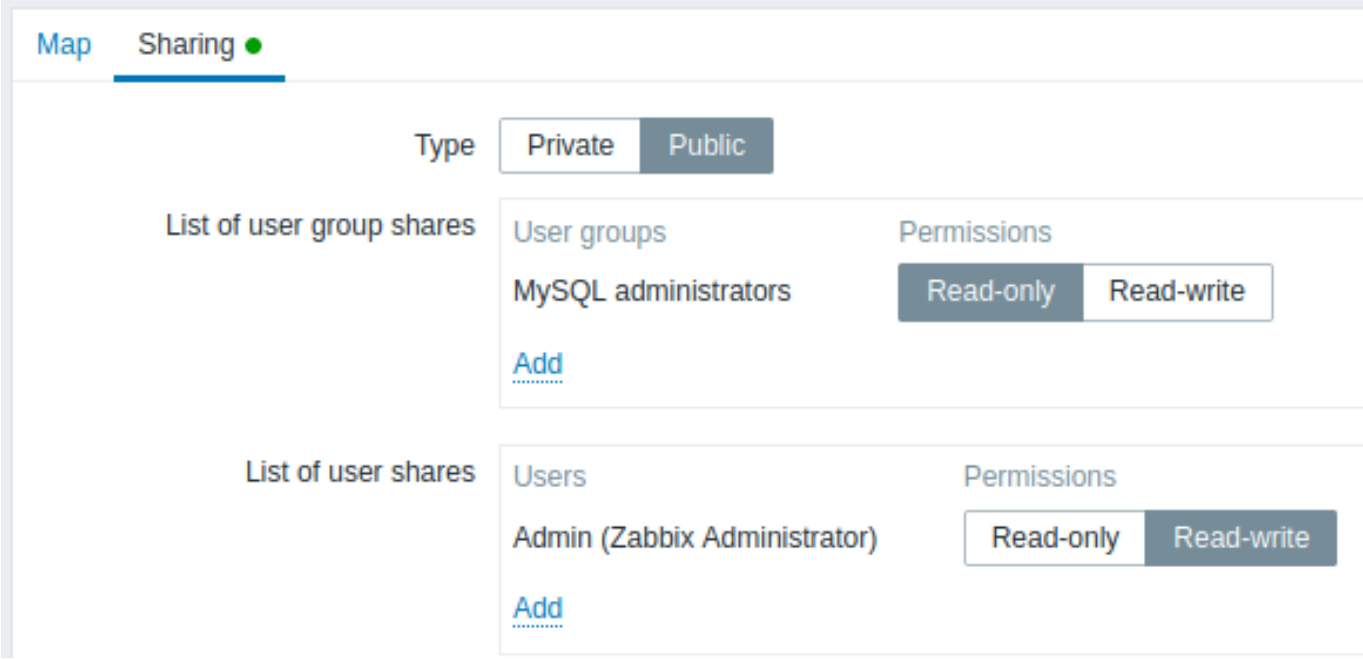
| Parameter | Description |
|---|---|
| *Owner* | Имя владельца карты. |
| *Name* | Уникальное имя карты. |
| *Width* | Ширина карты в пикселях. |
| *Height* | Высота карты в пикселях. |

| Parameter | Description |
|---|---|
| *Background image* | Использовать фоновое изображение:<br>**No image** - без фонового изображения (белый фон)<br>**Image** - выбранное изображение, которое будет использоваться в качестве фонового изображения. Вы можете использовать географическую карту или любое другое изображение, чтобы улучшить внешний вид вашей карты. |
| *Background scale* | Использовать масштабирование фона:<br>**None** - без масштабирования;<br>**Proportionally** - заполнить весь фон карты, сохраняя пропорции изображения (по умолчанию). |
| *Automatic icon mapping* | Можно включить автоматическое сопоставление значков, настроенное в *Administration > General > Icon mapping*. Сопоставление значков позволяет сопоставлять определенные значки с определенными полями инвентаризации узла сети. |
| *Icon highlight* | Если вы отметите этот флажок, элементы карты будут подсвечиваться.<br>Элементы с активным триггером получат круглую подложку того же цвета, что и триггер с наивысшей серьезностью. Кроме того, вокруг круга будет отображаться толстая зеленая линия, если все проблемы подтверждены.<br>Элементы со статусом ”disabled” или ”in maintenance” получат квадратную подложку серого и оранжевого цвета соответственно.<br>См. также: Viewing maps |
| *Mark elements on trigger status change* | Недавнее изменение статуса триггера (недавняя проблема или ее устранение) будет выделено маркерами (красными треугольниками, направленными внутрь) на трех сторонах значка элемента, свободных от подписи. Маркеры отображаются в течение 30 минут. |
| *Display problems* | Выберите, как отображать проблемы для элемента карты:<br>**Expand single problem** - если проблема только одна, отображается ее название. В противном случае отображается общее количество проблем.<br>**Number of problems** - отображается общее количество проблем<br>**Number of problems and expand most critical one** - отображается название наиболее критичной проблемы и общее количество проблем.<br>”Наиболее критичная” определяется на основе серьезности проблемы и, если она одинакова, ID события проблемы (сначала отображается больший ID или более поздняя проблема). Для *trigger map element* это определяется на основе серьезности проблемы и, если она одинакова, положения триггера в списке триггеров. Если у одного триггера несколько проблем, будет отображена самая последняя из них. |
| *Advanced labels* | Если вы отметите этот флажок, вы сможете задавать отдельные типы подписей для отдельных типов элементов. |
| *Host group label type* | Тип подписи, используемый для элемента карты:<br>**Label** - подпись элемента карты<br>**IP address** - IP-адрес<br>**Element name** - имя элемента (например, имя узла сети)<br>**Status only** - только статус (OK или PROBLEM)<br>**Nothing** - подписи не отображаются |
| *Host label type*<br>*Trigger label type*<br>*Map label type*<br>*Image label type* | |
| *Map element label location* | Расположение подписи относительно элемента карты:<br>**Bottom** - под элементом карты<br>**Left** - слева<br>**Right** - справа<br>**Top** - над элементом карты |
| *Show map element labels* | Выберите, как отображать подписи элементов карты:<br>**Always** - всегда отображать (по умолчанию);<br>**Auto hide** - скрывать подпись, если на нее не наведены курсор или она не выбрана. |
| *Show link labels* | Выберите, как отображать подписи связей:<br>**Always** - всегда отображать (по умолчанию);<br>**Auto hide** - скрывать подпись, если на нее не наведены курсор или она не выбрана. |

| Parameter | Description |
|---|---|
| *Problem display* | Отображать количество проблем как:<br>**All** - будет отображаться полное количество проблем<br>**Separated** - количество неподтвержденных проблем будет отображаться отдельно от общего количества проблем<br>**Unacknowledged only** - будет отображаться только количество неподтвержденных проблем |
| *Minimum severity* | Проблемы ниже выбранного минимального уровня серьезности не будут отображаться на карте.<br>Например, если выбрано *Warning*, изменения с триггерами уровня *Information* и *Not classified* не будут отражаться на карте. |
| *Show suppressed problems* | Отметьте флажок, чтобы отображать проблемы, которые в противном случае были бы подавлены (не показаны) из-за обслуживания узла сети. |
| *URLs* | Здесь можно определить URL-адреса для каждого типа элемента, где:<br>**Name** - подпись для URL;<br>**URL** - URL-адрес (до 2048 символов);<br>**Element** - тип элемента (например, *Host*, *Host group* и т. д.).<br>Они будут отображаться как ссылки, когда пользователь щелкает по элементу в режиме просмотра карты.<br>В именах и значениях URL карты можно использовать макросы. Полный список см. в разделе supported macros и найдите 'map URL names and values'. |

Общий доступ

Вкладка **Общий доступ** содержит тип карты, а также параметры общего доступа (группы пользователей, пользователи) для приватных карт:




| Parameter | Description |
|---|---|
| *Type* | Выберите тип карты:<br>**Private** - карта видна только выбранным группам пользователей и пользователям<br>**Public** - карта видна всем |
| *List of user group shares* | Выберите группы пользователей, которым доступна карта.<br>Можно разрешить доступ только для чтения или для чтения и записи. |
| *List of user shares* | Выберите пользователей, которым доступна карта.<br>Можно разрешить доступ только для чтения или для чтения и записи. |

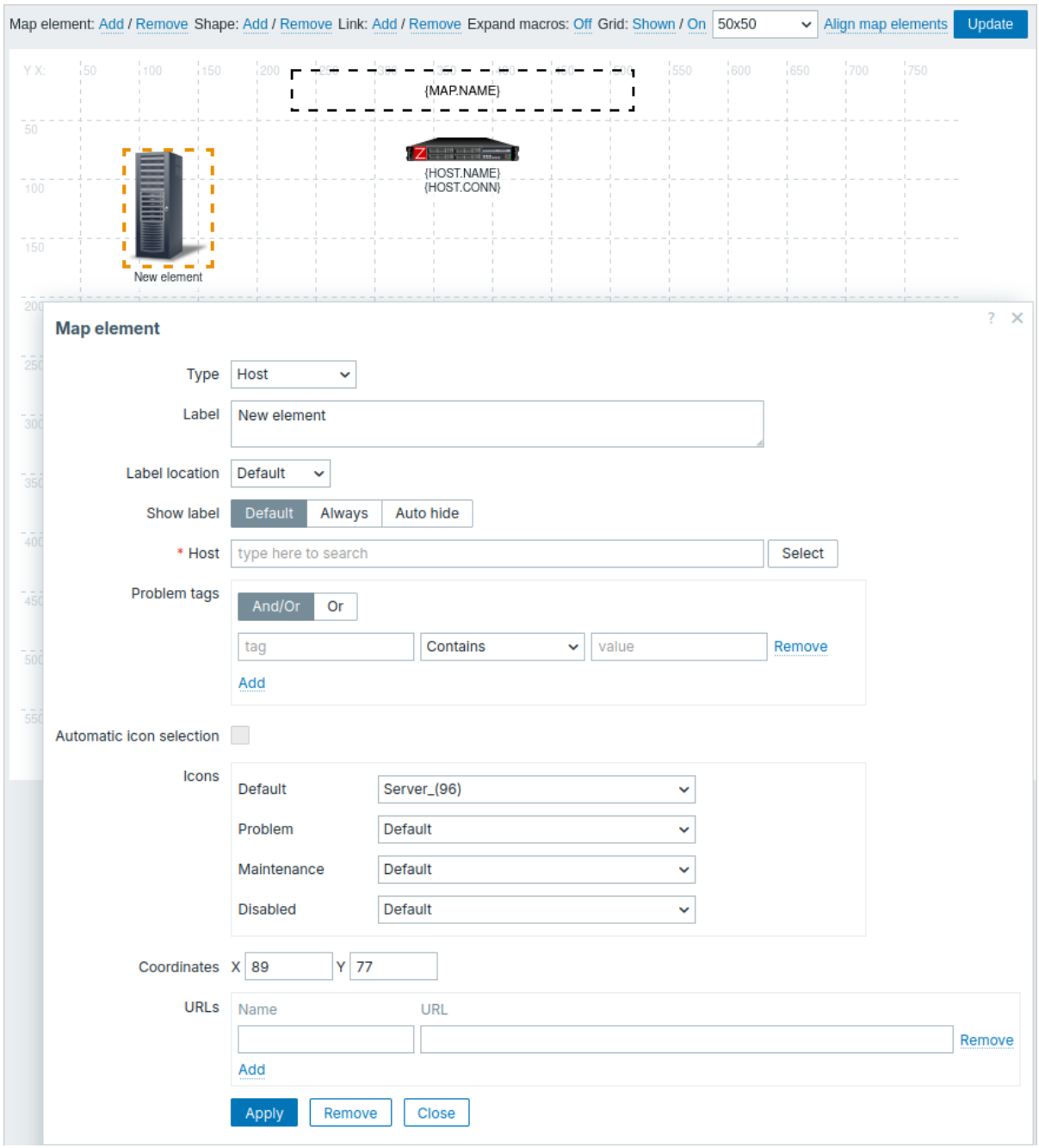
Когда вы нажимаете *Add* для сохранения этой карты, создается пустая карта с именем, размерами и определенными настройками. Теперь нужно добавить несколько элементов. Для этого нажмите *Edit* в списке карт, чтобы открыть область редактирования.

Добавление элементов

Чтобы добавить элемент, нажмите *Add* рядом с `Map element`. Новый элемент появится в левом верхнем углу карты. Перетащите его в любое удобное место.

Обратите внимание: если для параметра Grid включено значение "On", элементы всегда будут выравниваться по сетке (можно выбрать разные размеры сетки в раскрывающемся списке, а также скрыть/показать сетку). Если вы хотите размещать элементы где угодно без выравнивания, переключите параметр в "Off". (Позже можно выровнять случайно размещенные элементы по сетке, нажав *Align map elements*.)

Теперь, когда у вас уже есть несколько элементов, вы, возможно, захотите начать различать их, задавая им имена и т. д. При нажатии на элемент отображается форма, в которой можно задать тип элемента, указать имя, выбрать другой значок и т. д.




Атрибуты элемента карты:

| Parameter | Description |
|---|---|
| *Type* | Тип элемента:<br>**Host** - значок, отображающий состояние всех триггеров выбранного узла сети<br>**Map** - значок, отображающий состояние всех элементов карты<br>**Trigger** - значок, отображающий состояние одного или нескольких триггеров<br>**Host group** - значок, отображающий состояние всех триггеров всех узлов сети, входящих в выбранную группу<br>**Image** - значок, не связанный ни с каким ресурсом |
| *Label* | Подпись значка, любая строка.<br>Можно использовать макросы и многострочные строки.<br>В этом поле поддерживаются макросы выражений macros, но только с функциями `avg`, `last`, `min` и `max`, с временем в качестве параметра (например, `{?avg(/host/key,1h)}`).<br>Полный список поддерживаемых макросов см. в разделе supported macros и найдите 'map element labels'. |
| *Label location* | Расположение подписи относительно значка:<br>**Default** - расположение подписи по умолчанию для карты;<br>**Bottom** - под значком;<br>**Left** - слева;<br>**Right** - справа;<br>**Top** - над значком. |
| *Show label* | Выберите, как отображать подпись элемента:<br>**Default** - использовать настройки *Show map element labels* из конфигурации карты (по умолчанию);<br>**Always** - всегда отображать (по умолчанию);<br>**Auto hide** - скрывать подпись, если на нее не наведены курсор или она не выбрана. |
| *Host* | Выберите узел сети; либо введите имя узла сети. Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет показан выпадающий список подходящих узлов сети. Прокрутите список вниз, чтобы выбрать один из них. Нажмите значок удаления (x), чтобы снять выбор.<br>Это поле доступно только для элементов типа *Host*. |
| *Map* | Выберите карту; либо введите имя карты. Это поле поддерживает автодополнение, поэтому при вводе имени карты будет показан выпадающий список подходящих карт. Прокрутите список вниз, чтобы выбрать одну из них. Нажмите значок удаления (x), чтобы снять выбор.<br>Это поле доступно только для элементов типа *Map*. |
| *Triggers* | Выберите один или несколько триггеров в поле *New triggers* ниже и нажмите *Add*. Либо начните вводить имя триггера и выберите один из выпадающего списка подходящих триггеров. Нажмите значок удаления (x), чтобы снять выбор.<br>Это поле доступно только для элементов типа *Trigger*.<br>Порядок выбранных триггеров можно изменить, но только в пределах одной и той же критичности триггеров. Множественный выбор триггеров также влияет на разрешение макросов {HOST.*} как в режиме редактирования, так и в режиме просмотра.<br>В режиме редактирования первые отображаемые макросы {HOST.*} будут разрешаться в зависимости от первого триггера в списке (на основе критичности триггера).<br>Режим просмотра зависит от параметра Display problems в общих атрибутах карты:<br>- Если выбран режим *Expand single problem*, первые отображаемые макросы {HOST.*} будут разрешаться в зависимости от последнего обнаруженного проблемного триггера (независимо от критичности) или первого триггера в списке (если проблем не обнаружено).<br>- Если выбран режим *Number of problems and expand most critical one*, первые отображаемые макросы {HOST.*} будут разрешаться в зависимости от критичности триггера. |
| *Host group* | Выберите группу узлов сети; либо введите группу узлов сети. Это поле поддерживает автодополнение, поэтому при вводе имени группы узлов сети будет показан выпадающий список подходящих групп узлов сети. Прокрутите список вниз, чтобы выбрать одну из них. Нажмите значок удаления (x), чтобы снять выбор.<br>Это поле доступно только для элементов типа *Host group*. |

| Parameter | Description |
|---|---|
| *Problem tags* | Укажите теги, чтобы ограничить число проблем, отображаемых в виджете. Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны несколько операторов:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия<br>Это поле доступно только для элементов типа *Host* и *Host group*. |
| *Automatic icon selection* | Если флажок установлен, для определения отображаемого значка будет использоваться сопоставление значков. |
| *Icons* | Можно выбрать разные значки для элемента в следующих случаях: *Default*, *Problem*, *Maintenance* и *Disabled*.<br>Это поле доступно только для элементов типа *Host*. |
| *Coordinates* | Координаты элемента карты по осям **X** и **Y**. |
| *URLs* | Здесь можно задать URL-адреса, относящиеся к элементу, где:<br>**Name** - подпись URL;<br>**URL** - URL-адрес (до 2048 символов).<br>Они будут отображаться как ссылки, когда пользователь нажмет на элемент в режиме просмотра карты. Если у элемента есть собственные URL-адреса и для его типа на уровне карты определены URL-адреса, они будут объединены в одном меню.<br>В именах и значениях элементов карты можно использовать макросы. Полный список см. в разделе supported macros и найдите 'map URL names and values'. |

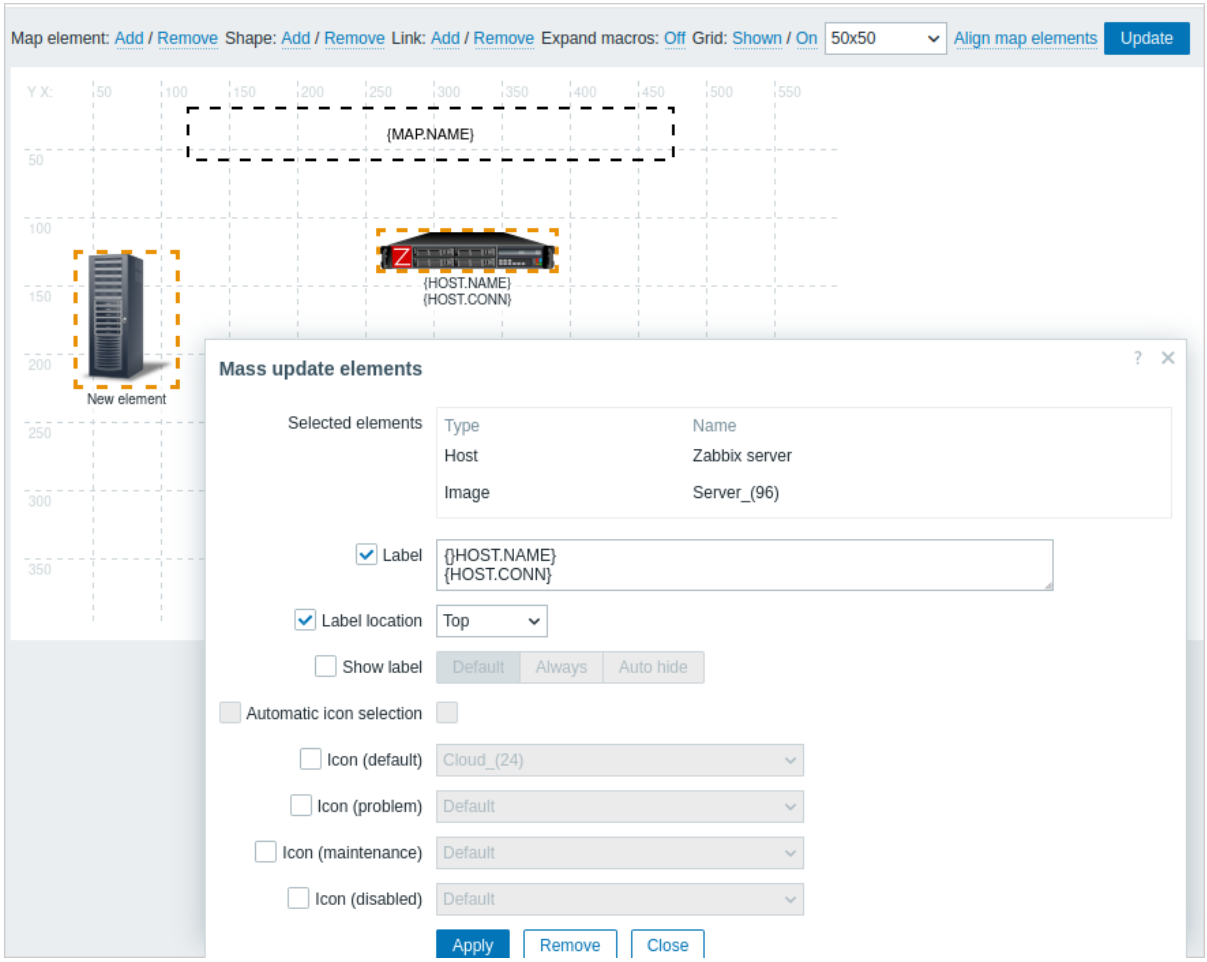
**Attention:**
Добавленные элементы не сохраняются автоматически. Если вы перейдете на другую страницу, все изменения могут быть потеряны.
Поэтому рекомендуется нажать кнопку **Update** в правом верхнем углу. После нажатия изменения сохраняются независимо от того, что вы выберете во всплывающем окне далее.
Выбранные параметры сетки также сохраняются для каждой карты.

Выбор элементов

Для выбора нескольких элементов, выберите сначала один элемент, затем нажмите и удерживайте *Ctrl*, чтобы выбрать другие.

Вы также можете выбрать несколько элементов, растягивая прямоугольник в области редактирования и тем самым выбирая все элементы, находящиеся в нём.

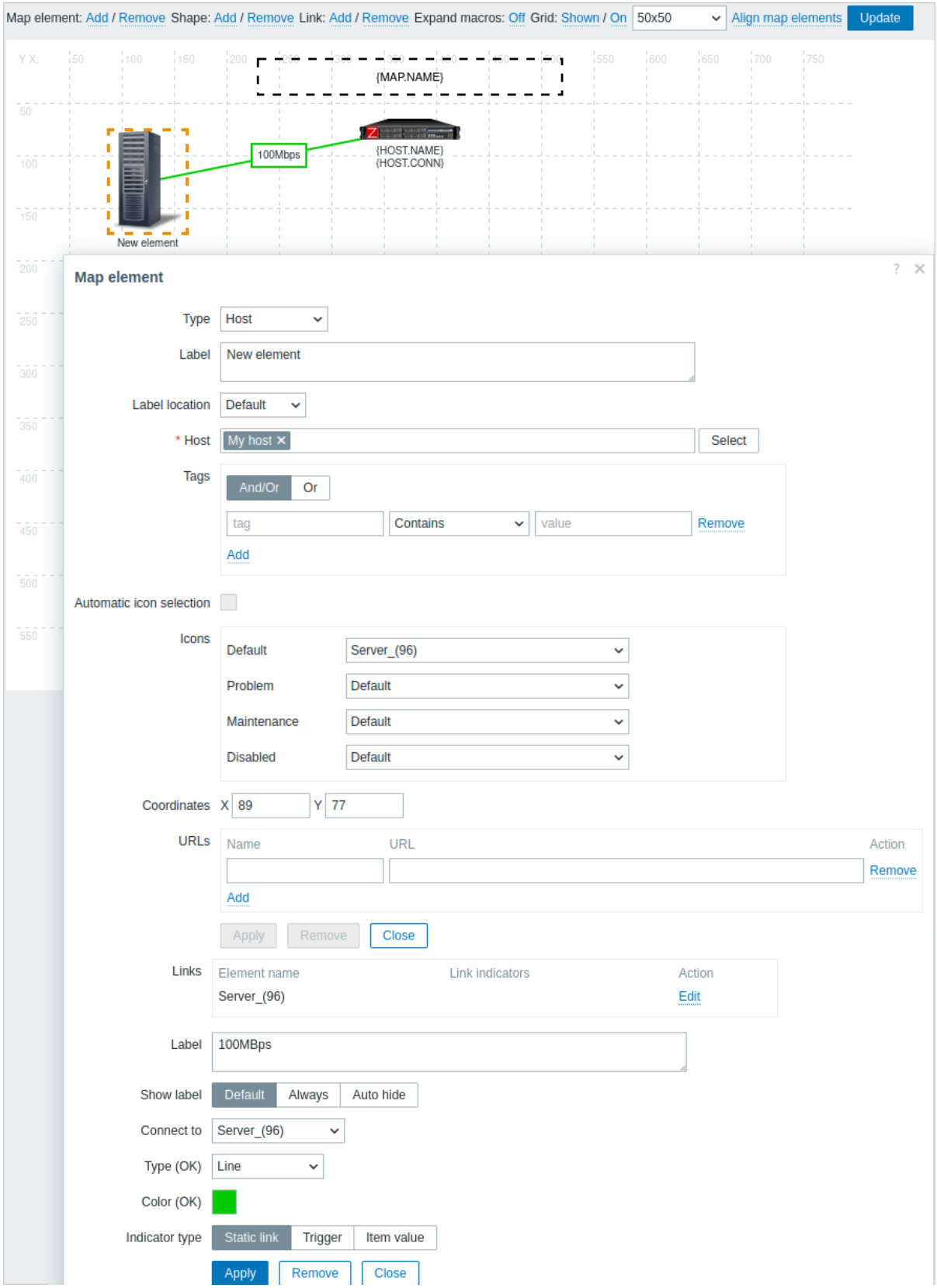
Как только вы выберете более одного элемента, диалог свойств элемента переключится в режим массового обновления, так что вы сможете изменить атрибуты выбранных элементов за раз. Чтобы это сделать, отметьте атрибут, используя флажок, и укажите для него новое значение. Здесь вы можете использовать макросы (такие как, скажем, {HOST.NAME} для подписи элемента).




Связывание элементов

После того как вы разместили на карте несколько элементов, можно приступать к их связыванию. Чтобы связать два элемента, сначала нужно выделить их. После выбора элементов нажмите *Добавить* рядом с *Link*.

После создания связи в форме отдельного элемента появляется дополнительный раздел *Links*. Нажмите *Edit*, чтобы изменить атрибуты связи.




Атрибуты связи:

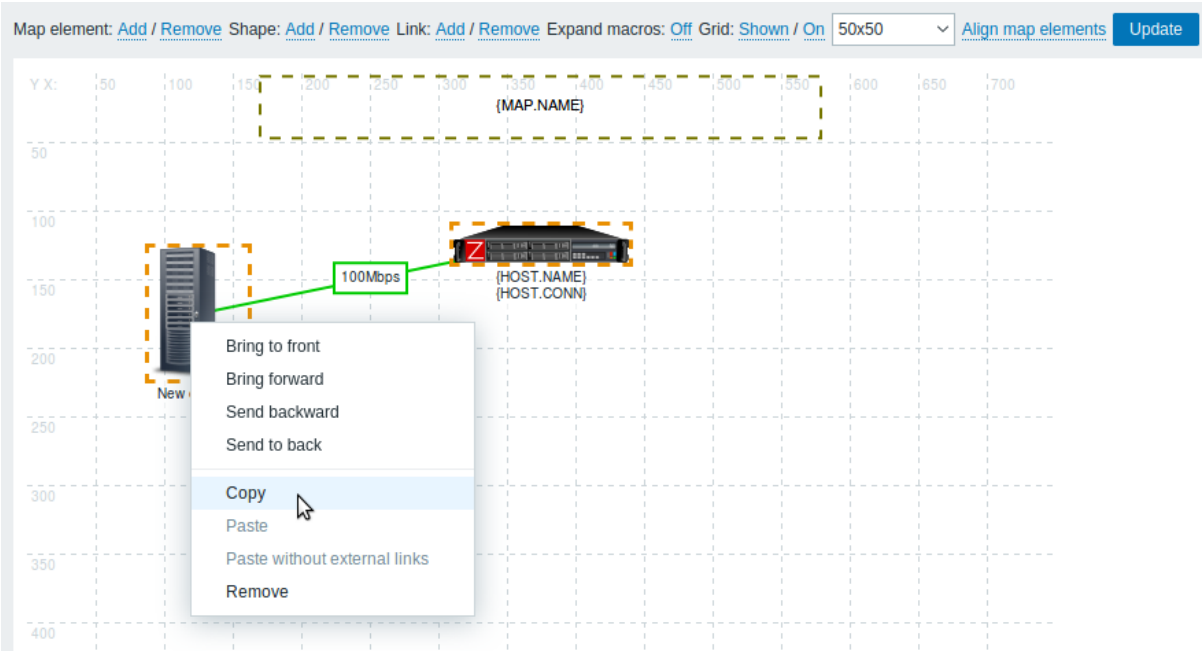
| Parameter | Description |
|---|---|
| *Label* | Метка, которая будет отображаться над связью.<br>В этом поле поддерживаются выражения macros, но только с функциями `avg`, `last`, `min` и `max`, с временем в качестве параметра (например, `{?avg(/host/key,1h)}`). |

| Parameter | Description |
|---|---|
| *Show label* | Выберите, как отображать метку связи:<br>**Default** - использовать настройки *Show map element labels* из конфигурации карты (по умолчанию);<br>**Always** - всегда отображать (по умолчанию);<br>**Auto hide** - скрывать метку, если на нее не наведено указатель мыши и она не выбрана. |
| *Connect to* | Элемент, к которому подключается связь. |
| *Type (OK)* | Стиль связи по умолчанию. Выберите стиль: *Line*, *Bold line*, *Dot* или *Dashed line*. |
| *Color (OK)* | Цвет связи по умолчанию. Щелкните образец цвета, чтобы выбрать другой цвет. |
| *Indicator type* | Выберите тип индикатора связи:<br>**Static link** - без индикаторов;<br>**Trigger** - разрешить индикаторы связи на основе триггеров;<br>**Item value** - разрешить индикаторы связи на основе значений элемента данных. |
| *Item* | Выберите элемент данных. Значение этого элемента данных будет влиять на стиль связи.<br>Это поле доступно, когда для *Indicator type* выбрано *Item value*. |
| *Indicators* | Список link indicators.<br>Это поле доступно, когда для *Indicator type* задано значение *Trigger* или *Item value*.<br>Если выбрано *Trigger*, в качестве индикаторов связи можно назначать триггеры. Когда триггер находится в состоянии *Problem*, к связи будет применен его стиль (выбранный цвет и тип линии).<br>Если выбрано *Item value* и в поле *Item* выбран элемент данных, в качестве индикаторов связи можно назначать значения элемента данных. Когда значение элемента данных достигает указанного порога (для числового элемента данных) или совпадает с шаблоном (для текстового элемента данных), к связи применяется его стиль (выбранный цвет и тип линии). |

Перемещение и копирование элементов

Несколько выбранных элементов можно **переместить** в другое место на карте, щелкнув один из выбранных элементов, удерживая нажатой кнопку мыши и перемещая курсор в нужное место.

Один или несколько элементов можно **скопировать**, выделив элементы, а затем щелкнув выбранный элемент правой кнопкой мыши и выбрав *Copy* в меню.

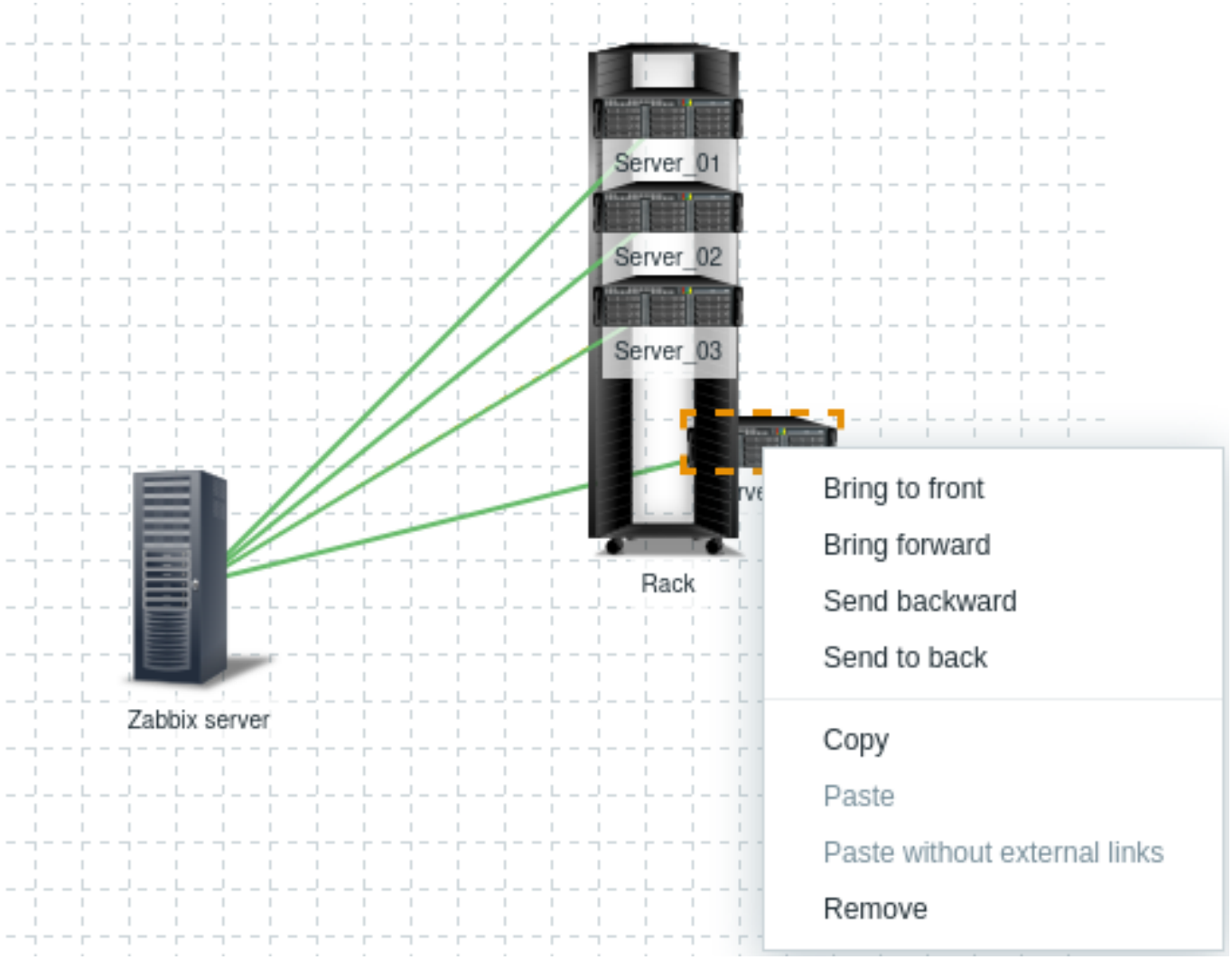



Чтобы вставить элементы, щелкните область карты правой кнопкой мыши и выберите *Paste* в меню. Параметр *Paste without external links* вставит элементы, сохранив только связи между выбранными элементами.

Копирование и вставка работают в пределах одного окна браузера. Сочетания клавиш не поддерживаются.

Порядок элементов

Чтобы переместить один элемент перед другим (или наоборот), щелкните элемент правой кнопкой мыши и выберите *Переместить вперед/Переместить на передний план* или *Переместить назад/Переместить на задний план*.

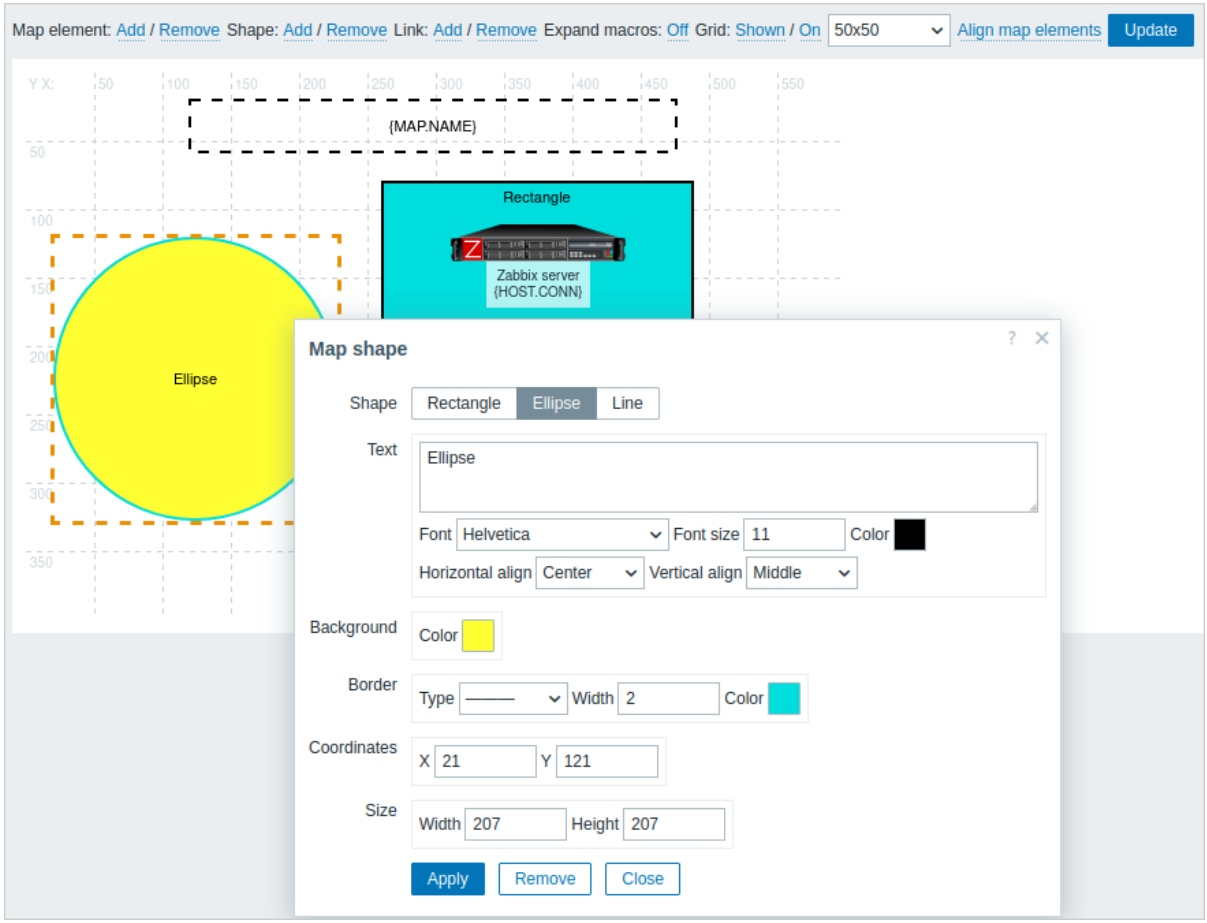



Добавление фигур

Помимо элементов карты, можно также добавлять фигуры. Фигуры не являются элементами карты; это лишь визуальное представление. Например, фигуру в виде прямоугольника можно использовать в качестве фона для группировки нескольких узлов сети. Можно добавлять фигуры в виде прямоугольника и эллипса.

Чтобы добавить фигуру, нажмите *Добавить* рядом с полем Shape. Новая фигура появится в левом верхнем углу карты. Перетащите ее в любое нужное место.

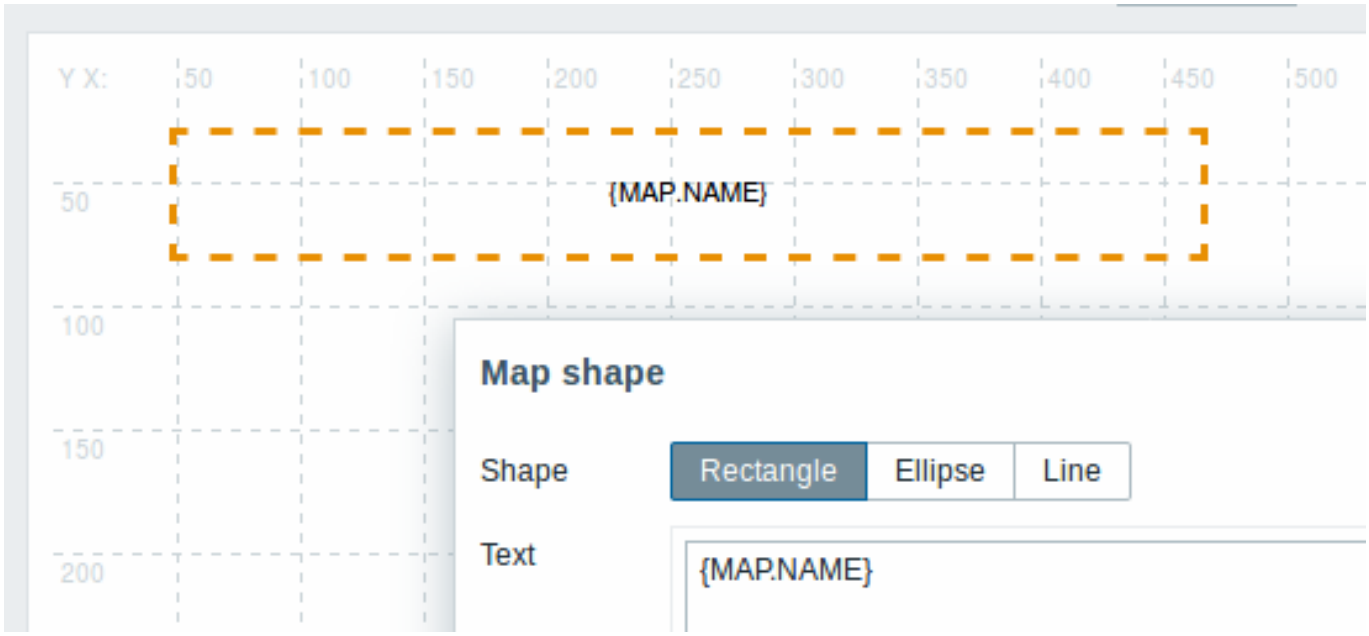
Новая фигура добавляется со стандартными цветами. При нажатии на фигуру отображается форма, в которой можно настроить ее внешний вид, добавить текст и т. д.




Чтобы выбрать фигуры, выберите одну из них, а затем, удерживая *Ctrl*, выберите остальные. При выборе нескольких фигур общие свойства можно массово обновлять так же, как и для элементов.

В фигуры можно добавлять текст. В тексте поддерживаются выражения macros, но только с функциями `avg`, `last`, `min` и `max`, с временем в качестве параметра (например, `{?avg(/host/key,1h)}`).

Чтобы отображать только текст, фигуру можно сделать невидимой, удалив ее границу (выберите 'None' в поле *Border*). Например, обратите внимание, что макрос {MAP.NAME}, видимый на скриншоте выше, на самом деле является фигурой-прямоугольником с текстом, который можно увидеть при нажатии на макрос:




{MAP.NAME} подставляет настроенное имя карты при просмотре карты.

Если в тексте используются гиперссылки, при просмотре карты они становятся кликабельными.

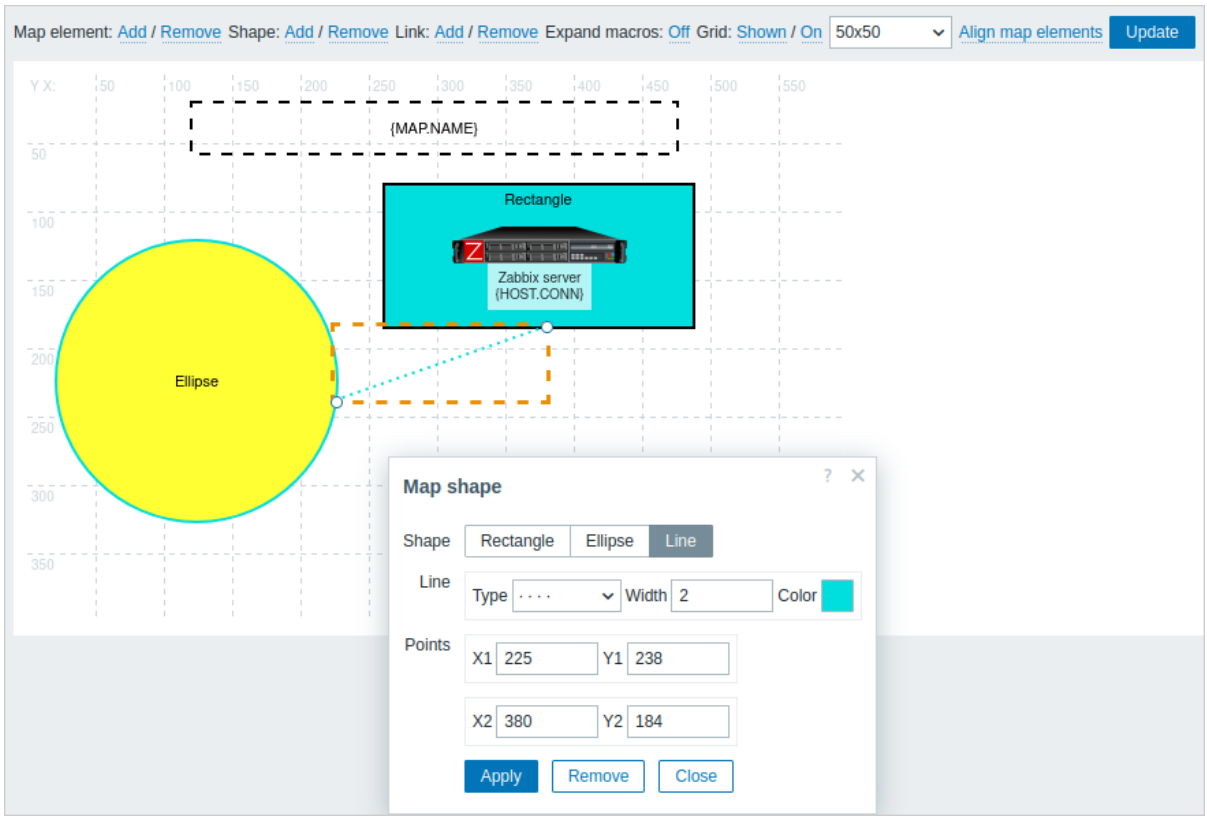
Перенос строк в тексте внутри фигур всегда включен. Однако внутри эллипса строки переносятся так, как если бы эллипс был прямоугольником. Перенос по словам не реализован, поэтому длинные слова (слова, которые не помещаются

в фигуру) не переносятся, а маскируются (на странице редактирования карты) или обрезаются (на других страницах с картами).

Добавление линий

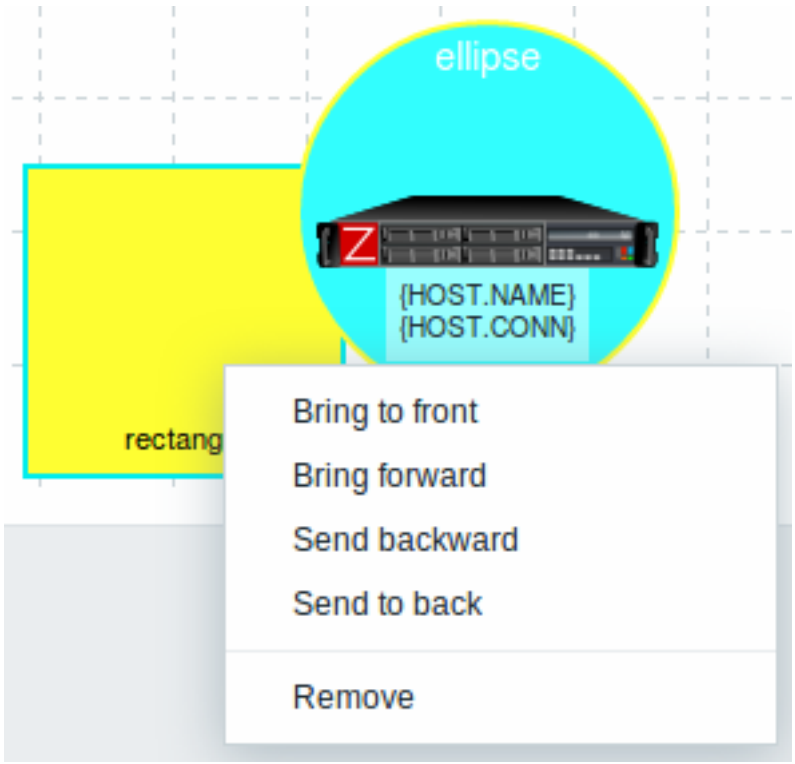
Помимо фигур, также можно добавлять линии. Линии можно использовать для связывания элементов или фигур на карте.

Чтобы добавить линию, нажмите *Add* рядом с Shape. Новая фигура появится в левом верхнем углу карты. Выберите ее и нажмите *Line* в форме редактирования, чтобы преобразовать фигуру в линию. Затем настройте свойства линии, такие как тип линии, ширина, цвет и т. д.




Упорядочивание фигур и линий

Чтобы переместить одну фигуру перед другой (или наоборот), щелкните фигуру правой кнопкой мыши, чтобы открыть меню фигур карты.

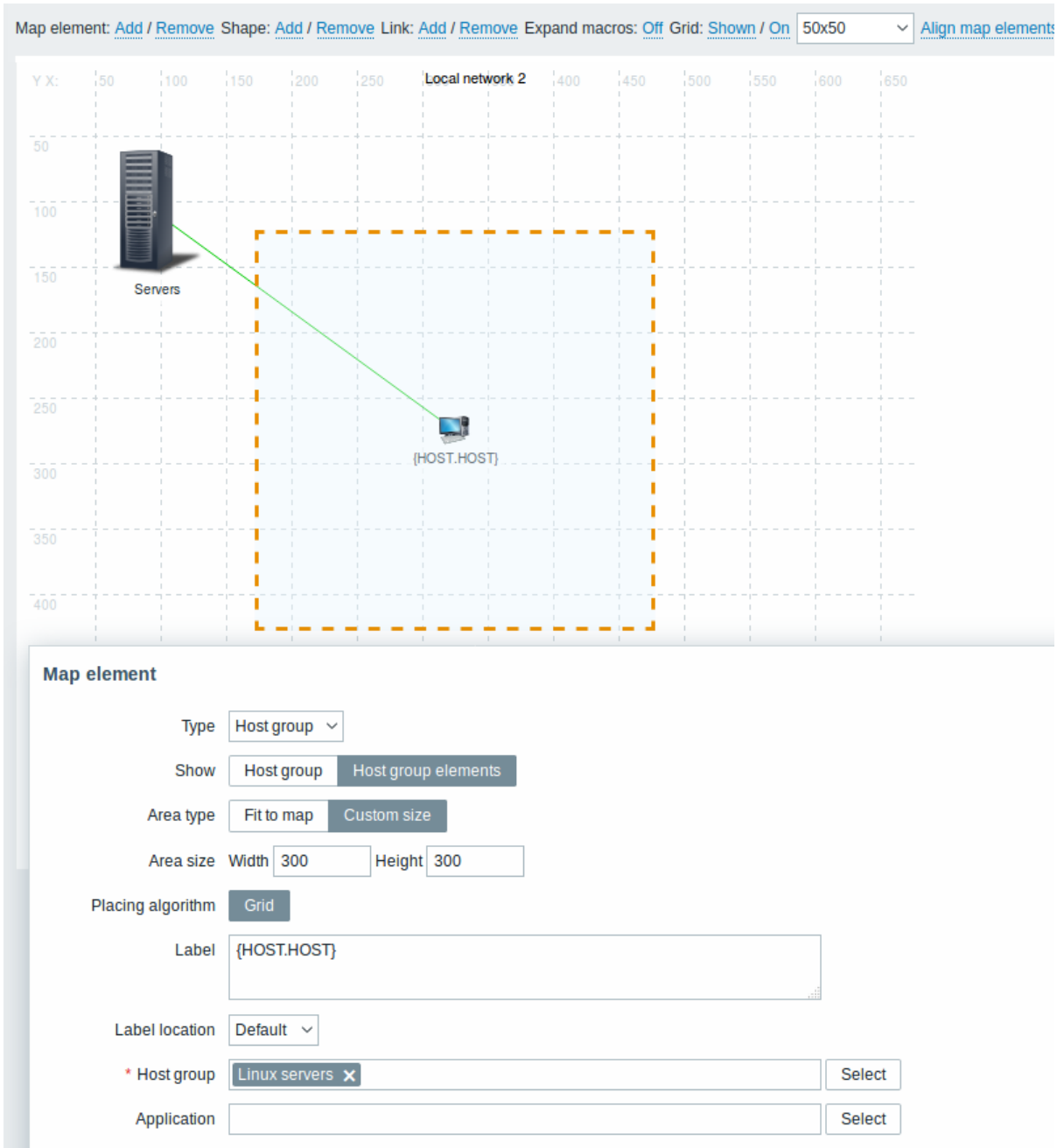


### 2. Элементы групп узлов сети

Обзор

Этот раздел объясняет, как добавить элемент типа *Группа узлов сети* при настройке карты сети.

Конфигурация




Все обязательные поля ввода отмечены красной звездочкой.

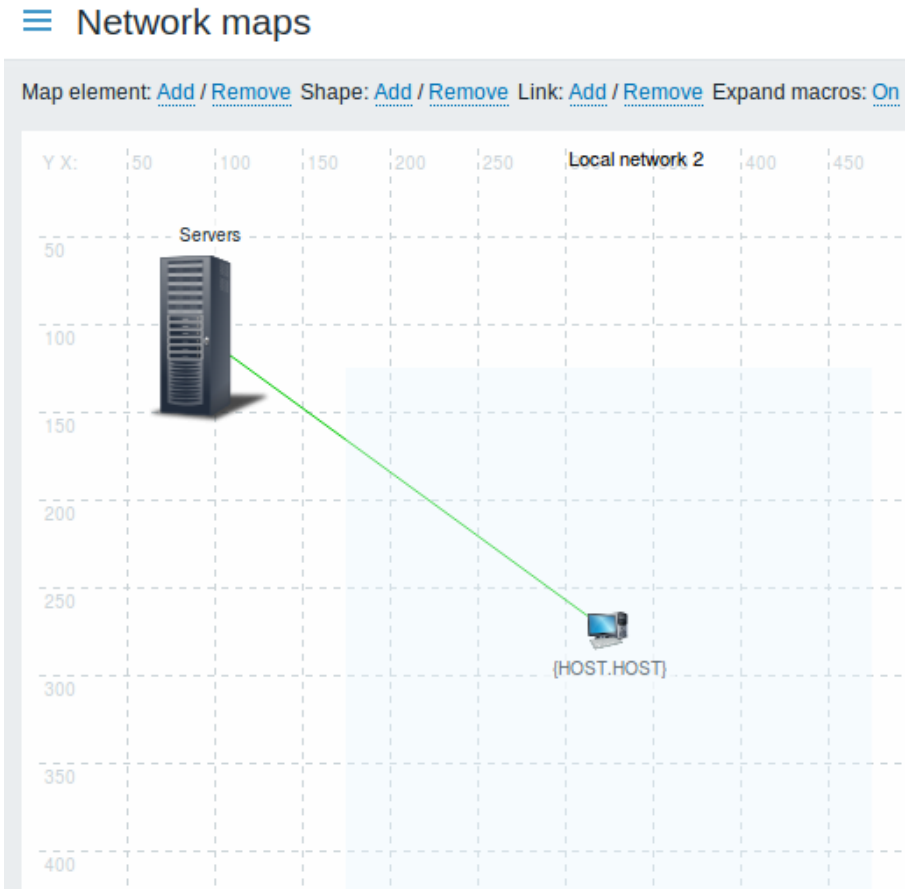
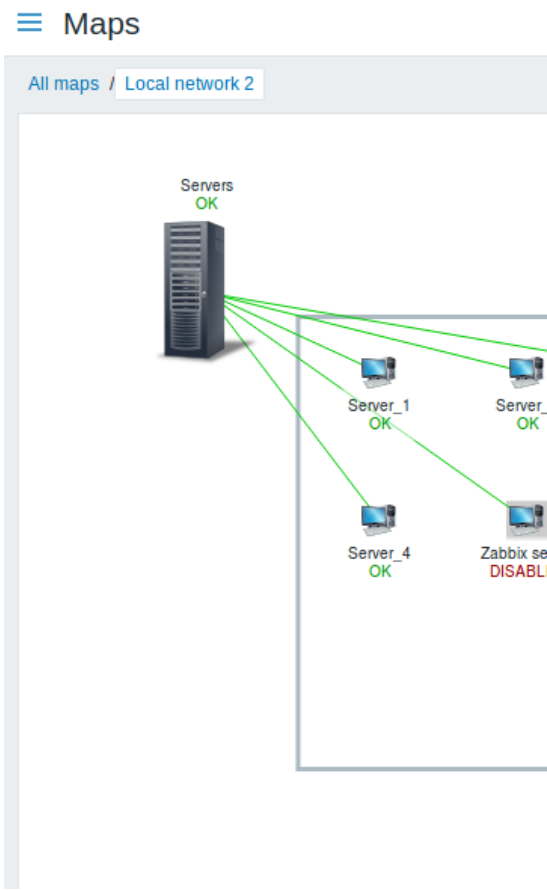
Эта таблица содержит параметры, типичные для элемента типа *Host group*:

| Parameter | Description |
|---|---|
| *Type* | Выберите тип элемента:<br>**Host group** - Значок, представляющий состояние всех триггеров всех узлов сети, входящих в выбранную группу. |
| *Show* | Параметры отображения:<br>**Host group** - При выборе этой опции будет отображаться один значок с соответствующей информацией о конкретной группе узлов сети.<br>**Host group elements** - При выборе этой опции будет отображаться несколько значков с соответствующей информацией о каждом отдельном элементе (узле сети) данной группы узлов сети. |

| Parameter | Description |
|---|---|
| *Area type* | Этот параметр доступен, если для *Show* установлено значение *Host group elements*:<br>**Fit to map** - Все элементы группы узлов сети равномерно размещаются на карте.<br>**Custom size** - Ручная настройка области карты для отображения всех элементов группы узлов сети. |
| *Area size* | Этот параметр доступен, если для *Area type* установлено значение *Custom size*:<br>**Width** - Значение в пикселях, задающее ширину области карты.<br>**Height** - Значение в пикселях, задающее высоту области карты. |
| *Placing algorithm* | **Grid** - Единственный доступный вариант отображения всех элементов группы узлов сети. |
| *Label* | Подпись значка, любая строка.<br>Macros и многострочные строки можно использовать в подписях.<br>Если тип элемента карты — *Host group*, указание определенных макросов влияет на отображение карты, показывая соответствующую информацию о каждом отдельном узле сети. Например, если используется макрос {HOST.IP}, в режиме редактирования карты будет отображаться только сам макрос {HOST.IP}, а в режиме просмотра карты будет включен и отображен уникальный IP-адрес каждого узла сети. |

Просмотр элементов группы узлов сети

Эта опция доступна, если выбрана опция отображения *Элементы группы узлов сети*. При выборе *Элементы группы узлов сети* в качестве опции *показа* сначала вы увидите только один значок для группы узлов сети. Однако после сохранения карты и перехода к просмотру карты вы увидите, что карта включает все элементы (узлы сети) определенной группы узлов сети:

| Вид редактирования карты | Просмотр карты |
|---|---|
|  |  |



Обратите внимание, как используется макрос {HOST.HOST}. При редактировании карты имя макроса не разрешается, тогда как в режиме просмотра карты отображаются все уникальные имена узлов сети.

**3. Индикаторы связей**

Обзор

Вы можете назначать индикаторы для ссылки между элементами на карте сети.

Индикаторы могут основываться на триггерах или значениях элементов данных. Можно отображать различный стиль и цвет ссылки:

- когда триггеры переходят в состояние проблемы;
- когда значение элемента данных:
  - достигает порогового значения (для числовых элементов данных);
  - соответствует регулярному выражению (для текстовых элементов данных).

При настройке ссылки вы задаете тип и цвет ссылки по умолчанию. Если назначить индикаторы для ссылки, становится возможным сделать стиль и цвет ссылки зависящими от статуса триггера или значения элемента данных.

Например, если любой из назначенных триггеров перейдет в состояние проблемы, стиль и цвет ссылки изменятся соответствующим образом. Допустим, ваша ссылка по умолчанию была зеленой линией. Теперь, когда триггер находится в состоянии проблемы, ссылка может стать жирной красной линией (если вы так ее настроили).

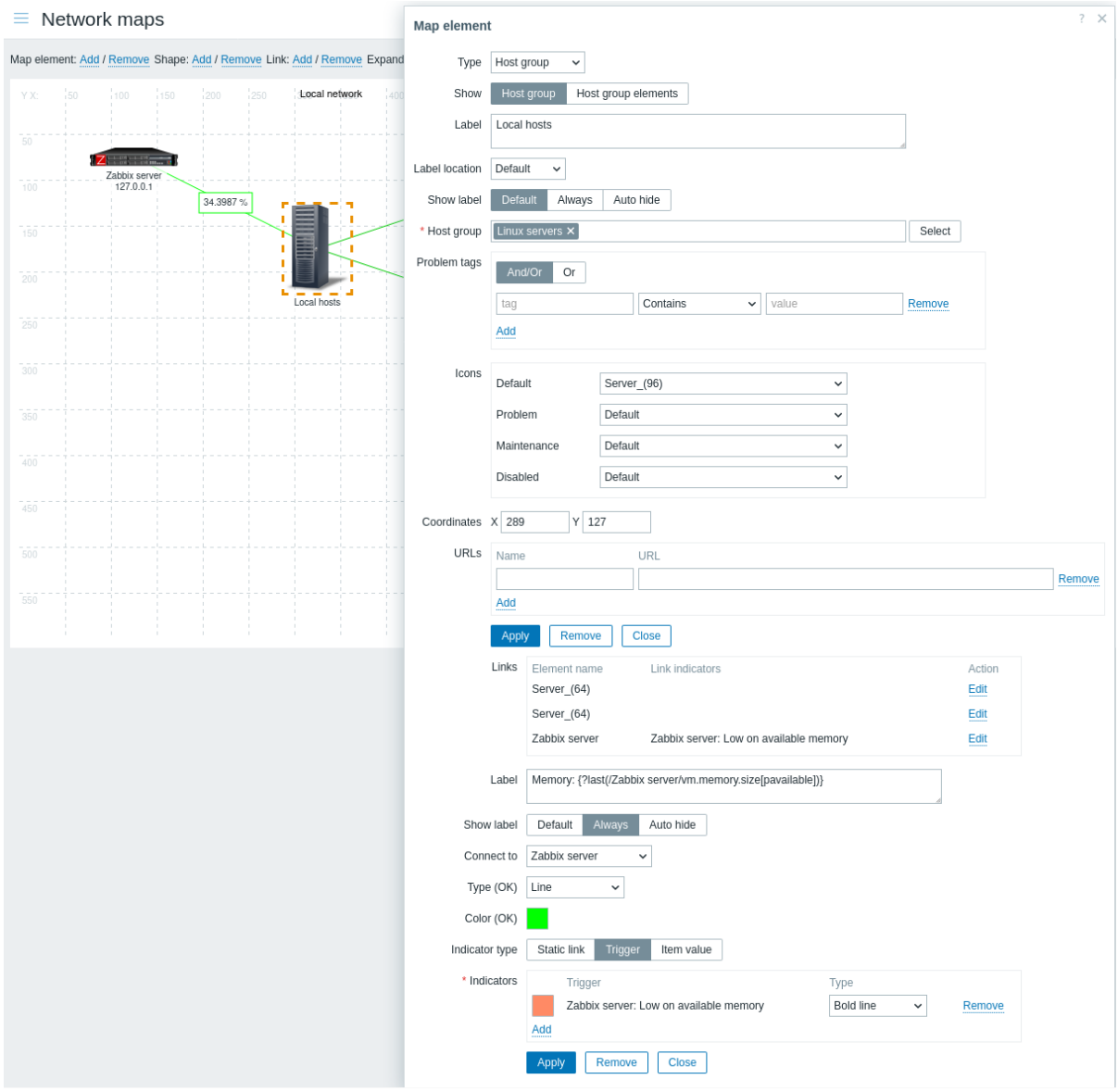
Аналогично, если значение элемента данных достигает заданного порога или соответствует заданному регулярному выражению, стиль ссылки может измениться соответствующим образом.

Конфигурация

Триггеры

Чтобы назначить триггеры в качестве индикаторов связи, выполните следующие действия:

- выберите элемент карты
- нажмите *Edit* в разделе *Links* для нужной связи
- выберите *Trigger* в качестве типа индикатора
- нажмите *Add* в блоке *Indicators* и выберите один или несколько триггеров

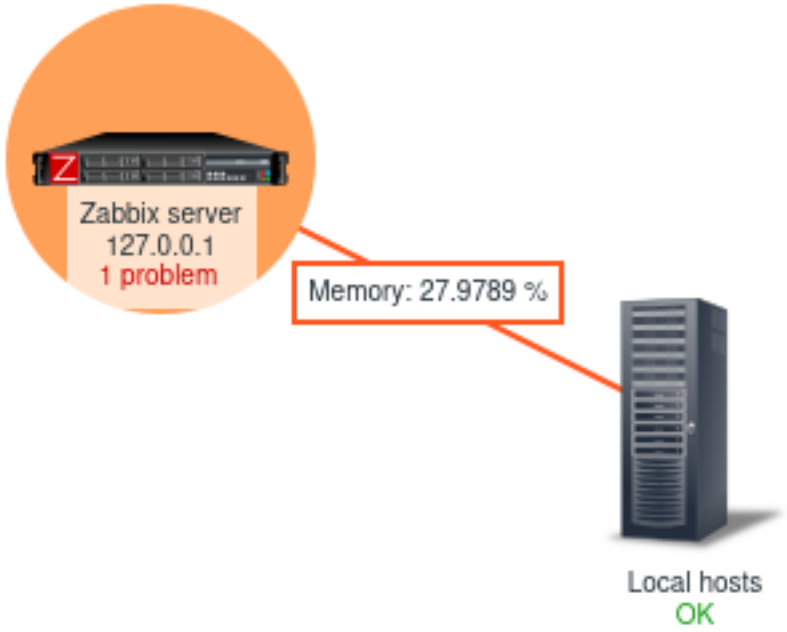


Все обязательные поля ввода отмечены красной звездочкой.

Добавленные триггеры можно увидеть в списке *Indicators*.

Вы можете задать тип связи и цвет для каждого индикатора прямо из списка. После этого нажмите *Apply*, закройте форму и нажмите *Update* , чтобы сохранить изменения карты.

В *Monitoring* → *Maps* соответствующий тип связи и цвет будут отображаться, если **trigger** перейдет в состояние проблемы.



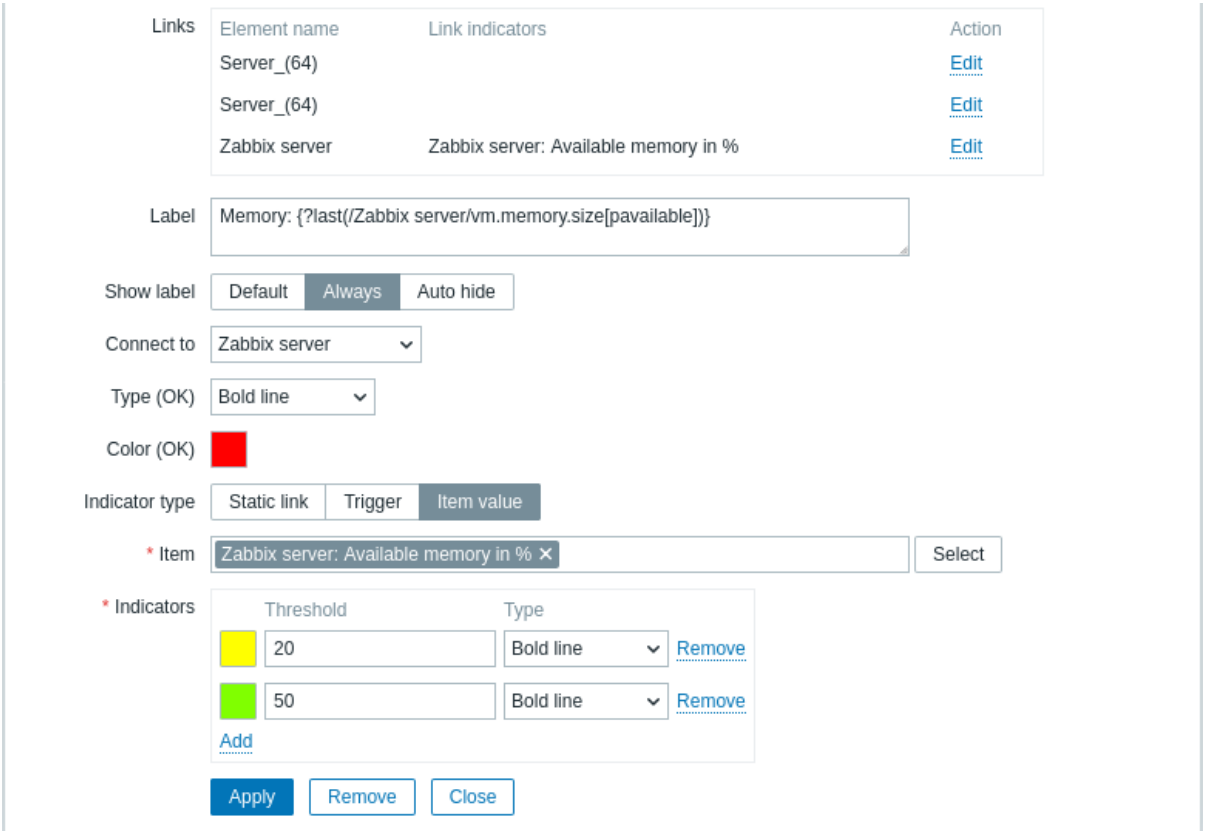

**Note:**
Если несколько триггеров перейдут в состояние проблемы, стиль и цвет связи будут определяться проблемой с наивысшей серьезностью. Если нескольким триггерам с одинаковой серьезностью назначена одна и та же связь карты, приоритет будет у триггера с наименьшим ID. Также обратите внимание, что:

1. Настройки *Minimum trigger severity* и *Show suppressed problem* в конфигурации карты влияют на то, какие проблемы будут учитываться.
2. В случае триггеров с несколькими проблемами (multiple problem generation) каждая проблема может иметь серьезность, отличающуюся от серьезности триггера (измененной вручную), может иметь разные теги (из-за макросов) и может быть подавлена.

Значения элементов данных

Чтобы назначить значения элементов данных в качестве индикаторов связи, выполните следующие действия:

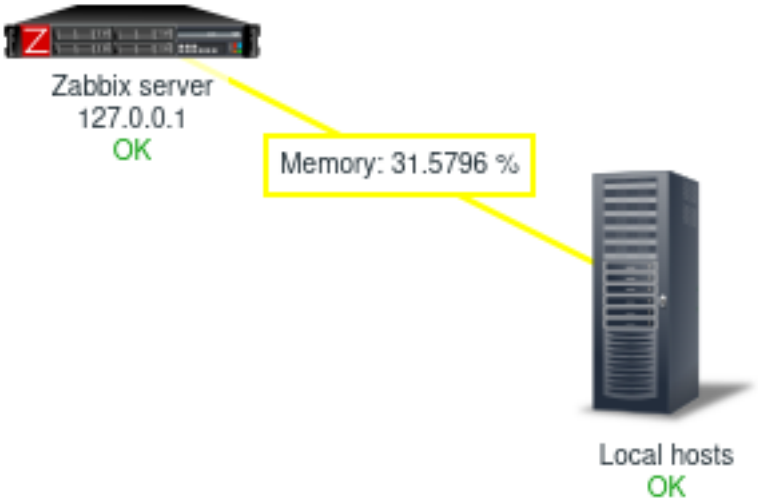
- выберите элемент карты
- нажмите *Edit* в разделе *Links* для нужной связи
- выберите *Item value* в качестве типа индикатора
- выберите элемент данных
- добавьте один или несколько порогов или шаблонов значений элемента данных в блоке *Indicators*




Добавленные пороги/шаблоны элементов данных можно увидеть в списке *Indicators*.

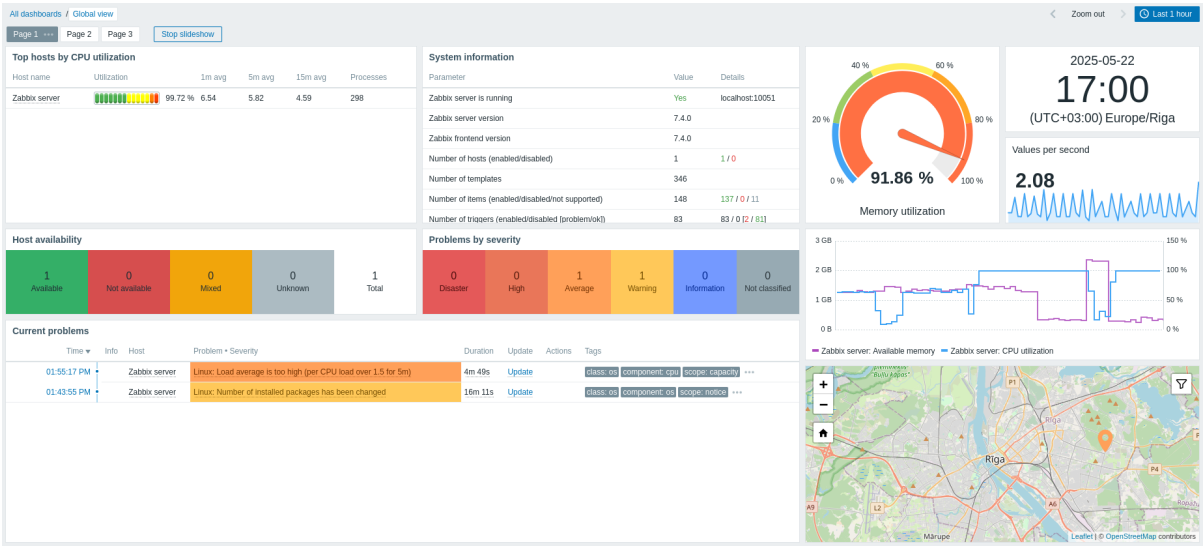
Вы можете задать тип связи и цвет для каждого индикатора прямо из списка. После этого нажмите *Apply*, закройте форму и нажмите *Update* , чтобы сохранить изменения карты.

В *Monitoring* → *Maps* соответствующий тип связи и цвет будут отображаться, если **item value** достигает заданного порога (для числовых типов данных) или соответствует шаблону регулярного выражения (для текстовых типов данных).




### 3. Панели

Панели — как глобальные панели, так и панели узлов сети — обеспечивают мощную платформу визуализации с такими виджетами и инструментами как: современные графики, карты, слайд-шоу и многое другое.




## 8. Шаблоны и группы шаблонов

Обзор

Использование шаблонов — отличный способ сократить рабочую нагрузку и оптимизировать настройки Zabbix. Шаблон — это набор сущностей, которые можно удобно применить к множеству узлов сети.

Сущностями могут быть:

- элементы данных
- триггеры
- графики
- панели
- правила низкоуровневого обнаружения
- веб-сценарии

Поскольку в реальной жизни многие узлы сети идентичны или довольно схожи, естественно следует, что набор сущностей (элементы данных, триггеры, графики...) созданные для одного узла сети, могут быть полезны для других узлов сети. Конечно можно было бы скопировать их на каждый новый узел сети, но это потребовало бы большого количества ручной работы. Вместо этого можно скопировать все такие сущности в один шаблон, а затем применить его к необходимому набору узлов сети.

Когда шаблон присоединён к узлу сети, все сущности (элементы, триггеры, графики,...) шаблона добавляются к узлу сети. Шаблоны назначаются каждому отдельному узлу сети напрямую (а не группе узлов сети).

Шаблоны часто используются для группировки сущностей в для определённых служб или приложений (таких как Apache, MySQL, PostgreSQL, Postfix...) и затем применяются к узлам сети, на которых работают эти службы.

Еще одно преимущество использования шаблонов — когда что-то нужно изменить для всех узлов сети. Изменение чего-либо на уровне шаблона один раз распространяется на все присоединённые к этому шаблону узлы сети.

Шаблоны организованы в группы шаблонов.

Перейдём к созданию и настройке шаблона.

## 9. Готовые шаблоны

Обзор

Zabbix предоставляет растущий набор предварительно настроенных шаблонов, чтобы упростить и ускорить настройку объектов мониторинга.

Все готовые шаблоны доступны в *Сбор данных* > *Шаблоны*.

При обновлении Zabbix существующие шаблоны не обновляются автоматически, чтобы не перезаписать пользовательские изменения. Чтобы обновить существующие шаблоны или добавить новые, см. Обновление шаблонов.

Пожалуйста, используйте боковую панель, чтобы получить информацию о конкретных типах шаблонов и требованиях к эксплуатации.

См. также:

- Связывание шаблона
- Известные проблемы для шаблонов

Обновление шаблона

Чтобы обновить один шаблон или добавить новый после обновления Zabbix:

1. Перейдите в репозиторий Zabbix Git.
2. В селекторе ветки (под меткой *Source*) выберите ветку, соответствующую вашей версии Zabbix (например, *release/7.4* для Zabbix 7.4).
3. Откройте шаблон, который вы хотите добавить или обновить (например, MySQL by Zabbix agent).
4. Скопируйте содержимое файла шаблона (например, `template_db_mysql_agent.yaml`) и сохраните его в локальный файл.
5. В веб-интерфейсе Zabbix перейдите в *Сбор данных* > *Шаблоны* и импортируйте локальный файл.

Также можно обновить все шаблоны и добавить новые сразу:

1. Скачайте пакет шаблонов в зависимости от вашего веб-сервера: `templates_1M_7.4.7z` (Nginx) или `templates_2M_7.4.7z` (Apache).
2. Распакуйте пакет. Он содержит YAML-файлы (каждый с несколькими шаблонами) и индекс (`index-1M_74.md`), в котором перечислены шаблоны, включенные в каждый YAML-файл.
3. В веб-интерфейсе Zabbix перейдите в *Сбор данных* > *Шаблоны* и импортируйте YAML-файлы по одному.

В качестве альтернативы можно импортировать все YAML-файлы сразу с помощью скрипта `import_templates.sh`. Для работы скрипта требуются утилиты `curl` и `jq`, а также используется метод Zabbix API `configuration.import`. При запуске скрипта укажите каталог, содержащий YAML-файлы, а также URL и токен аутентификации вашего Zabbix API.

> **Warning:**
> Скрипт `import_templates.sh` перезаписывает все существующие шаблоны без возможности просмотра или отмены изменений шаблонов.

Пример:

```
./import_templates.sh /tmp/templates_2M_7.4 https://example.com/zabbix/api_jsonrpc.php d8d6b5c78ee2a8333db
```

Совместимость шаблонов с мастером узлов сети

Полный список готовых шаблонов, которые на данный момент совместимы с мастером узлов сети, см. в файле `hw_compatible_templates.txt`.

Вы также можете сделать свои пользовательские шаблоны совместимыми с мастером узлов сети:

1. В веб-интерфейсе Zabbix перейдите в *Сбор данных* > *Шаблоны* и экспортируйте пользовательский шаблон.
2. Отредактируйте экспортированный файл:
   - Добавьте элемент шаблона `wizard_ready` template element, чтобы указать совместимость с мастером узлов сети.
   - При необходимости дополнительно добавьте элемент шаблона `readme` или `config` template element.
3. В веб-интерфейсе Zabbix перейдите в *Сбор данных* > *Шаблоны* и импортируйте обновленный файл шаблона.

#### 1 Операция шаблона Zabbix agent

Шаги для обеспечения корректной работы шаблонов, которые собирают метрики с помощью агента Zabbix:

1. Убедитесь, что агент Zabbix установлен на узле сети. Для активных проверок также убедитесь, что адрес сервера/прокси Zabbix добавлен в параметр `ServerActive` файла конфигурации агента.
2. Привяжите шаблон к целевому узлу сети (если шаблон недоступен в вашей установке Zabbix, сначала может потребоваться импортировать шаблон).
3. При необходимости настройте значения макросов шаблона.
4. Настройте контролируемый экземпляр так, чтобы он разрешал обмен данными с Zabbix.

Подробное описание шаблона, включая полный список макросов, элементов данных и триггеров, доступно в файле README шаблона (открывается при нажатии на имя шаблона).

Доступны следующие шаблоны:

- Apache by Zabbix agent
- Apache by Zabbix agent active
- HAProxy by Zabbix agent
- IIS by Zabbix agent
- IIS 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
- PHP-FPM by Zabbix agent
- PHP-FPM by Zabbix agent active
- PostgreSQL by Zabbix agent
- PostgreSQL by Zabbix agent active
- RabbitMQ cluster by Zabbix agent

#### 2 Работа шаблонов Zabbix agent 2

Шаги для обеспечения корректной работы шаблонов, которые собирают метрики с помощью Zabbix agent 2:

1. Убедитесь, что агент 2 установлен на узле сети и что установленная версия содержит необходимый плагин. В некоторых случаях сначала может потребоваться обновить агент 2.
2. Привяжите шаблон к целевому узлу сети (если шаблон недоступен в вашей установке Zabbix, сначала может потребоваться импортировать шаблон).
3. При необходимости настройте значения макросов шаблона. Обратите внимание, что пользовательские макросы можно использовать для переопределения параметров конфигурации.
4. Настройте систему, которая находится под мониторингом, чтобы разрешить обмен данными с Zabbix.

**Attention:**
Шаблоны Zabbix agent 2 работают совместно с плагинами. Хотя базовую настройку можно выполнить, просто изменив пользовательские макросы, более глубокую настройку можно выполнить, настроив плагин напрямую. Например, если плагин поддерживает именованные сеансы, можно отслеживать несколько сущностей одного типа (например, MySQL1 и MySQL2), указав для каждой сущности в файле конфигурации именованный сеанс с собственными URI, именем пользователя и паролем.

Подробное описание шаблона, включая полный список макросов, элементов данных и триггеров, доступно в файле README шаблона (открывается при нажатии на имя шаблона).

Доступны следующие шаблоны:

- 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

#### 3 Операция шаблона HTTP

Шаги для обеспечения корректной работы шаблонов, которые собирают метрики с помощью HTTP agent:

1. Создайте узел сети в Zabbix и укажите IP-адрес или DNS-имя объекта мониторинга в качестве основного интерфейса. Это необходимо для корректного разрешения макроса {HOST.CONN} в элементах данных шаблона.
2. Привяжите шаблон к узлу сети, созданному на шаге 1 (если шаблон недоступен в вашей установке Zabbix, сначала может потребоваться импортировать шаблон).
3. При необходимости скорректируйте значения макросов шаблона.
4. Настройте контролируемый экземпляр так, чтобы он разрешал обмен данными с Zabbix.

Подробное описание шаблона, включая полный список макросов, элементов данных и триггеров, доступно в файле README шаблона (открывается при нажатии на имя шаблона).

Доступны следующие шаблоны:

- 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
- ClickHouse by HTTP
- Cloudflare by HTTP
- CockroachDB by HTTP
- Control-M enterprise manager by HTTP
- Control-M server by HTTP
- DELL PowerEdge R720 by HTTP
- DELL PowerEdge R740 by HTTP
- DELL PowerEdge R820 by HTTP
- DELL PowerEdge R840 by HTTP
- Elasticsearch Cluster by HTTP
- Envoy Proxy by HTTP
- Etcd by HTTP
- FortiGate by HTTP
- GitHub repository by HTTP
- GitLab 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
- 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
- OpenStack by HTTP
- OpenWeatherMap by HTTP
- Oracle Cloud by HTTP
- Palo Alto PA-440 by HTTP
- PHP-FPM 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

#### 4 Операции с шаблоном IPMI

Шаблоны IPMI не требуют какой-либо специальной настройки. Чтобы начать мониторинг, свяжите шаблон с целевым узлом сети (если шаблон недоступен в вашей установке Zabbix, сначала может потребоваться импортировать шаблон).

Подробное описание шаблона, включая полный список макросов, элементов данных и триггеров, доступно в файле README шаблона (открывается при нажатии на имя шаблона).

Доступный шаблон:

- Chassis by IPMI

#### 5 Работа шаблонов JMX

Шаги для обеспечения корректной работы шаблонов, собирающих метрики по JMX:

1. Убедитесь, что Zabbix Java gateway установлен и правильно настроен.
2. Привяжите шаблон к целевому узлу сети. На узле сети должен быть настроен интерфейс JMX.<br> Если шаблон недоступен в вашей установке Zabbix, возможно, сначала потребуется импортировать шаблон.
3. При необходимости скорректируйте значения макросов шаблона.
4. Настройте экземпляр, который отслеживается, чтобы он разрешал обмен данными с Zabbix.

Подробное описание шаблона, включая полный список макросов, элементов данных и триггеров, доступно в файле README шаблона (открывается при нажатии на имя шаблона).

Доступны следующие шаблоны:

- 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

#### 6 Операция шаблона ODBC

Шаги для обеспечения корректной работы шаблонов, которые собирают метрики через мониторинг ODBC:

1. Убедитесь, что необходимый драйвер ODBC установлен на сервере Zabbix или прокси.
2. Привяжите шаблон к целевому узлу сети (если шаблон недоступен в вашей установке Zabbix, сначала может потребоваться импортировать шаблон).
3. При необходимости настройте значения макросов шаблона.
4. Настройте контролируемый экземпляр так, чтобы он разрешал обмен данными с Zabbix.

Подробное описание шаблона, включая полный список макросов, элементов данных и триггеров, доступно в файле README шаблона (открывается при нажатии на имя шаблона).

Доступны следующие шаблоны:

- MSSQL by ODBC
- MySQL by ODBC
- Oracle by ODBC
- PostgreSQL by ODBC

#### 7. Унифицированные шаблоны для сетевых устройств

Обзор

Для обеспечения мониторинга сетевых устройств, таких как коммутаторы и роутеры, мы создали две так называемые модели: для самого сетевого устройства (в основном его шасси) и для сетевых интерфейсов.

Шаблоны поставляются для многих семейств сетевых устройств. Все шаблоны охватывают (где возможно получить эти элементы данных с устройства):

- Мониторинг ошибок шасси (источники питания, вентиляторы, температуру и общее состояние)
- Мониторинг производительности шасси (элементы данных CPU и памяти)
- Сбор инвентарных данных шасси (серийных номеров, имени модели, версии прошивки)
- Мониторинг сетевых интерфейсов при помощи IF-MIB и EtherLike-MIB (состояние интерфейса, загрузка трафика на порту, состояние дуплекса для Ethernet)

Если шаблон недоступен в вашей инсталляции Zabbix, вам может потребоваться сначала импортировать шаблон.

Если вы импортируете новые готовые шаблоны, вы, возможно, также захотите обновить глобальное регулярное выражение `@Network interfaces for discovery` на:

```
: ^Software Loopback Interface
: ^(In)?[lL]oop[bB]ack[0-9._]*$
: ^NULL[0-9.]*$
: ^[lL]o[0-9.]*$
: ^[sS]ystem$
: ^Nu[0-9.]*$
```

для фильтрации loopback и нулевых интерфейсов в большинстве систем.

Устройства

Список семейств устройств, для которых доступны шаблоны:

| Template name | Vendor | Device family | Known models | OS | MIBs used | **Tags** |
|---|---|---|---|---|---|---|
| *Alcatel Timetra TiMOS SNMP* | Alcatel | Alcatel Timetra | ALCATEL SR 7750 | TiMOS | TIMETRA-SYSTEM-MIB,TIMETRA-CHASSIS-MIB | Certified |

| Template name | Vendor | Device family | Known models | OS | MIBs used | Tags |
|---|---|---|---|---|---|---|
| *Aruba CX 8300s by SNMP* | HPE Aruba Network-ing | Aruba CX Switch Series | 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 | Brocade FC switches | 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 | Next Genera-tion Firewall | - | 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 |
| *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 and 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 |

| Template name | Vendor | Device family | Known models | OS | MIBs used | Tags |
|---|---|---|---|---|---|---|
| *D-Link DES_DGS Switch SNMP* | D-Link | DES/DGX switches | 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<br>Fault<br>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<br>FORTINET-FORTIGATE-MIB<br>FORTINET-CORE-MIB<br>SNMPv2-MIB<br>IF-MIB<br>ENTITY-MIB | Performance<br>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 | HP Enterprise Switch | HP ProCurve J4900B Switch 2626, HP J9728A 2920-48G Switch | | STATISTICS-MIB,NETSWITCH-MIB,HP-ICF-CHASSIS,ENTITY-MIB,SEMI-MIB | Certified |
| *Huawei AR600 by SNMP* | Huawei | Huawei NetEngine AR600 Series Routers | 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 |

| Template name | Vendor | Device family | Known models | OS | MIBs used | **Tags** |
|---|---|---|---|---|---|---|
| *Huawei OceanStor Dorado by SNMP* | Huawei | Huawei OceanStor, Huawei OceanStor Dorado | Huawei OceanStor Dorado 3000/5000/6000/8000/18000/5300/5500/5600/6800/18500/18800 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 | Intel/QLogic Infini-band devices | Infiniband 12300 | | ICS-CHASSIS-MIB | Fault Inventory |
| *Juniper SNMP* | Juniper | MX,SRX,EX models | Juniper MX240, Juniper EX4200-24F | JunOS | JUNIPER-MIB | Certified |
| *Juniper MX NETCONF* | Juniper | MX models | Juniper MX204 Edge Router | JunOS 24.2R1-S1.10 | *Not applicable (uses NETCONF over SSH)* | Certified |
| *Juniper MX SNMP* | Juniper | MX models | 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 | Mellanox Infini-band devices | SX1036 | MLNX-OS | HOST-RESOURCES-MIB,ENTITY-MIB,ENTITY-SENSOR-MIB,MELLANOX-MIB | Certified |

| Template name | Vendor | Device family | Known models | OS | MIBs used | **Tags** |
|---|---|---|---|---|---|---|
| *MikroTik CCR<device model> SNMP* | MikroTik | MikroTik Cloud Core Routers (CCR series) | Separate dedicated templates are available for 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 |

| Template name | Vendor | Device family | Known models | OS | MIBs used | Tags |
|---|---|---|---|---|---|---|
| *MikroTik CRS<device model> SNMP* | MikroTik | MikroTik Cloud Router Switches (CRS series) | Separate dedicated templates are available for 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 | MikroTik Cloud Smart Switches (CSS series) | Separate dedicated templates are available for 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 | Separate dedicated templates are available for MikroTik hEX, MikroTik hEX lite, MikroTik hEX PoE, MikroTik hEX PoE lite, MikroTik hEX S | RouterOS | MIKROTIK-MIB,HOST-RESOURCES-MIB | Certified |

| Template name | Vendor | Device family | Known models | OS | MIBs used | Tags |
|---|---|---|---|---|---|---|
| *MikroTik netPower <device model> SNMP* | MikroTik | MikroTik net-Power | Separate dedicated templates are available for 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 Power-Box | Separate dedicated templates are available for MikroTik PowerBox, MikroTik PowerBox Pro | RouterOS | MIKROTIK-MIB,HOST-RESOURCES-MIB | Certified |
| *MikroTik RB<device model> SNMP* | MikroTik | MikroTik RB series routers | Separate dedicated templates are available for 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 | MikroTik RouterOS devices | MikroTik CCR1016-12G, MikroTik RB2011UAS-2HnD, MikroTik 912UAG-5HPnD, MikroTik 941-2nD, MikroTik 951G-2HnD, MikroTik 1100AHx2 | RouterOS | MIKROTIK-MIB,HOST-RESOURCES-MIB | Certified |
| *Netgear Fastpath SNMP* | Netgear | Netgear Fastpath | M5300-28G | | FASTPATH-SWITCHING-MIB,FASTPATH-BOXSERVICES-PRIVATE-MIB | Fault Inventory |

| Template name | Vendor | Device family | Known models | OS | MIBs used | Tags |
|---|---|---|---|---|---|---|
| *QTech QSW SNMP* | QTech | Qtech devices | Qtech QSW-2800-28T | - | QTECH-MIB,ENTITY-MIB | Performance Inventory |
| *Stormshield SNS by SNMP* | Stormshield | Stormshield Network Security (SNS) firewalls | 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 | Ubiquiti AirOS wireless devices | 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 |

Дизайн шаблонов

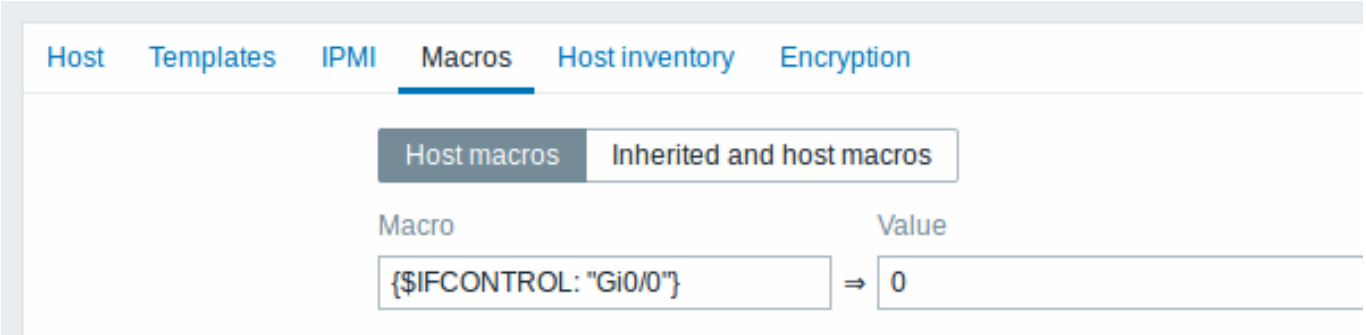
Шаблоны разработаны с учетом следующего:

- Пользовательские макросы используются где только возможно, поэтому триггеры можно подстроить;
- Низкоуровневые обнаружения используются где только возможно для минимизации количества неподдерживаемых элементов данных;
- Все шаблоны зависят от Template ICMP Ping, таким образом все устройства так же проверяются по протоколу ICMP;
- Элементы данных не используют MIB — в элементах данных и правилах обнаружения используются только SNMP OID. Поэтому для работы шаблонов нет необходимости загружать какие-либо MIB в Zabbix;
- Сетевые интерфейсы loopback при обнаружении фильтруются, также как и интерфейсы с ifAdminStatus = down(2)
- Используются 64-битные счётчики из IF-MIB::ifXTable, где это возможно. Если они не поддерживаются, по умолчанию вместо них используются 32-битные счётчики.

У всех обнаруженных сетевых интерфейсов имеется триггер, который следит за его рабочим состоянием (каналом связи), например:

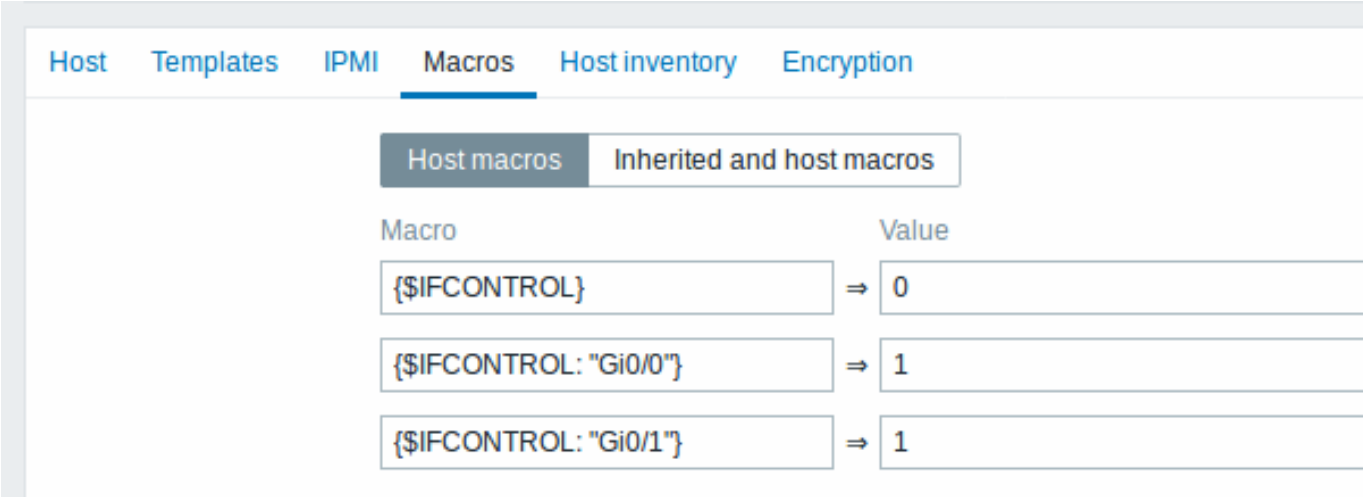
```
{$IFCONTROL:"{#IFNAME}"}=1 and last(/Alcatel Timetra TiMOS SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])=
```

- Если вы не хотите мониторить это условие по какому-либо отдельному интерфейсу, создайте пользовательский макрос с контекстом со значением 0. Например:




где Gi0/0 является {#IFNAME}. Таким образом, триггер больше не будет использоваться для этого конкретного интерфейса.

- Вы также можете изменить поведение по умолчанию по всем триггерам (не срабатывать) и активировать этот триггер только по ограниченному количеству интерфейсов, например, только по магистральным каналам связи:




Теги

- Performance (Прозводительность) — MIB'ы семейства устройств дают возможность мониторинга элементов данных CPU и памяти;
- Fault (Ошибки) — MIB'ы семейства устройств дают возможность мониторинга по крайней мере одного датчика температуры;
- Inventory (Инвентарные данные) — MIB'ы семейства устройств дают возможность сбора по крайней мере серийного номера устройства и его наименование модели;
- Certified (Сертифицированный) — охватываются все три основные категории выше.

**8 Операция шаблона VMware**

Обзор

Zabbix включает набор готовых к использованию шаблонов для мониторинга виртуальных машин, предназначенных для гипервизоров VMware vCenter и ESXi. Доступные шаблоны входят в два отдельных набора шаблонов:

- VMware - использует данные UUID для соответствующих макросов
- VMware FQDN - использует данные FQDN для соответствующих макросов

Шаблоны, предназначенные для мониторинга гипервизоров VMware vCenter или ESXi, содержат предварительно настроенные правила низкоуровневого обнаружения, а также различные встроенные проверки для мониторинга виртуальных установок.

Для корректной работы шаблона *VMware FQDN* каждая отслеживаемая виртуальная машина должна иметь уникальное имя ОС, соответствующее правилам FQDN. Кроме того, на каждой машине должны быть установлены VMware Tools/Open Virtual Machine tools. Если эти требования выполнены, рекомендуется использовать шаблон *VMware FQDN*. Шаблон *VMware FQDN* доступен начиная с Zabbix 5.2, вместе с появлением возможности создавать узлы сети с пользовательскими интерфейсами. <br><br> Также доступен классический шаблон *VMware*, который можно использовать, если требования к FQDN не выполняются. Однако у шаблона *VMware* есть известная проблема. Узлы сети для обнаруженных виртуальных машин создаются с именами, сохраненными в vCenter (например, "VM1", "VM2" и т. д.). Если на этих узлах сети установлен агент Zabbix и включена активная авто-регистрация агента Zabbix, процесс авто-регистрации будет считывать имена

узлов сети такими, какими они были зарегистрированы при запуске (например, ”vm1.example.com”, ”vm2.example.com” и т. д.). Это может привести к созданию новых узлов сети для уже существующих виртуальных машин (поскольку совпадения имен не найдены), что приводит к появлению дублирующихся узлов сети с разными именами.

Обратите внимание:

- Шаблон *VMware Hypervisor* можно вручную привязать к узлу сети, а также использовать в обнаружении;
- Шаблон *VMware Guest* не следует вручную привязывать к узлу сети, и его можно использовать только в обнаружении.

Конфигурация макросов узла сети

Чтобы использовать простые проверки VMware, на узле сети должны быть определены следующие пользовательские макросы:

- {$VMWARE.URL} - URL SDK службы VMware (vCenter или гипервизор ESXi) (https://servername/sdk)
- {$VMWARE.USERNAME} - имя пользователя службы VMware
- {$VMWARE.PASSWORD} - пароль пользователя {$VMWARE.USERNAME} службы VMware

**10. Оповещения на события**

Обзор

Предположим, мы уже настроили несколько элементов данных и триггеров и теперь мы получаем некоторые события, которые появляются как результат изменения состояния триггеров; настало время рассмотреть некоторые действия.

Начнем с того, что мы не хотим наблюдать за триггерами или за списком событий всё время. Было бы неплохо получать оповещения, если происходит что-то значительное (например, проблемы). Кроме того, когда возникают проблемы, мы бы хотели видеть, что все заинтересованные персоны проинформированы об этом.

Вот почему отправка оповещений является одной из ключевых задач, предлагаемой Zabbix. Можно указать, кто и когда должен быть оповещён при наступлении определённого события.

Чтобы была возможность отправлять и получать оповещения от Zabbix, вы должны:

- настроить какие-нибудь способы оповещения
- настроить действия, которые отправляют сообщения по одному из заданных способов оповещения

Действия состоят из *условий* и *операций*. В общем смысле, когда условия удовлетворяются, выполняются операции. Две основные операции — это отправка сообщений (оповещений) и выполнение удалённых команд.

Для событий, созданных обнаружениями и автоматической регистрацией, доступны несколько дополнительных операций. Они включают в себя добавление и удаление узлов сети, присоединение шаблонов и др.

**1. Способы оповещений**

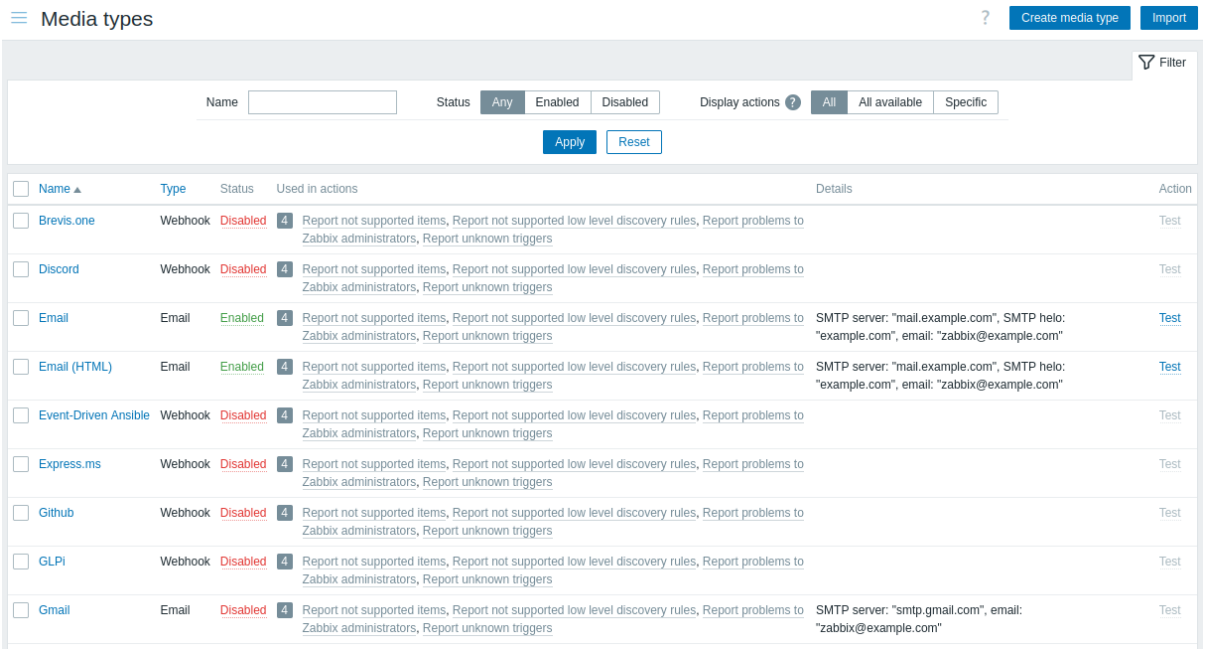
Обзор

Способы оповещений являются каналами доставки при отправке оповещений и предупреждений из Zabbix.

Способы оповещений поддерживают следующие методы доставки:

- Email
- SMS
- Пользовательские скрипты
- Вебхук

Способы оповещений хранятся в *Оповещения → Способы оповещений* (*Alerts → Media types*). Некоторые способы оповещений поставляются в наборе данных по умолчанию. Вам нужно только настроить их параметры, чтобы они заработали.




Чтобы понять, как способы оповещений вписываются в процесс оповещения, давайте рассмотрим три требования для доставки уведомлений от Zabbix конечным пользователям:

1. Необходимо определить операцию действия, которая отсылает оповещения
2. Необходимо определить работающий способ оповещений (например, *Email*, который отсылает уведомления через SMTP)
3. Информация о доставке на уровне пользователя (например, адреса e-mail, номера телефонов и т.п.) должна быть задана в оповещениях пользователей

Тестирование типа оповещения

Чтобы проверить, работает ли настроенный тип оповещения, нажмите *Test* в списке типов оповещений.

Запрос на тестирование будет отправлен на сервер Zabbix. Сервер Zabbix попытается отправить оповещение с использованием указанного типа оповещения и вернет результат во веб-интерфейс. Веб-интерфейс будет ожидать, пока сервер вернет результаты. По умолчанию для тестирования типа оповещения установлен тайм-аут 65 секунд (настраивается в *Administration* > *General* > *Timeouts*).

См. также подробности тестирования для:

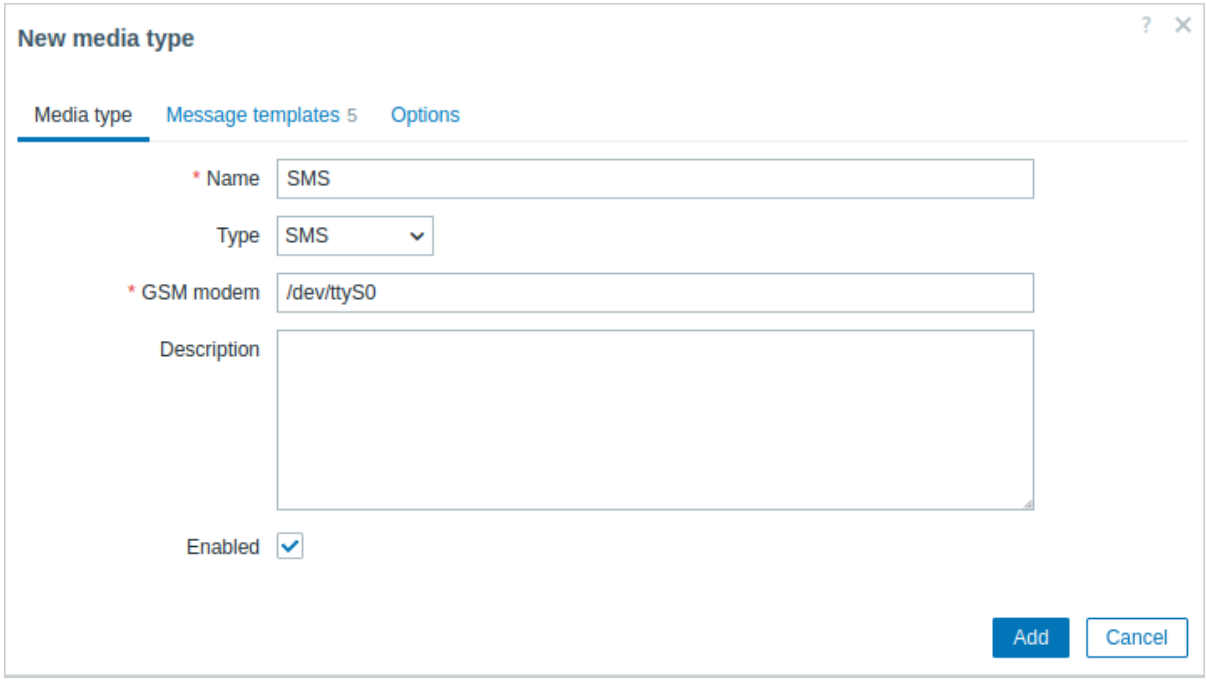
- Email
- Webhook
- Script

Конфигурация

Чтобы создать тип оповещений во веб-интерфейсе Zabbix:

- Перейдите в: *Оповещения* > *Типы оповещений*
- Нажмите *Создать тип оповещений*
- Введите параметры типа оповещений в форме

Некоторые параметры являются **общими** для всех способов доставки.

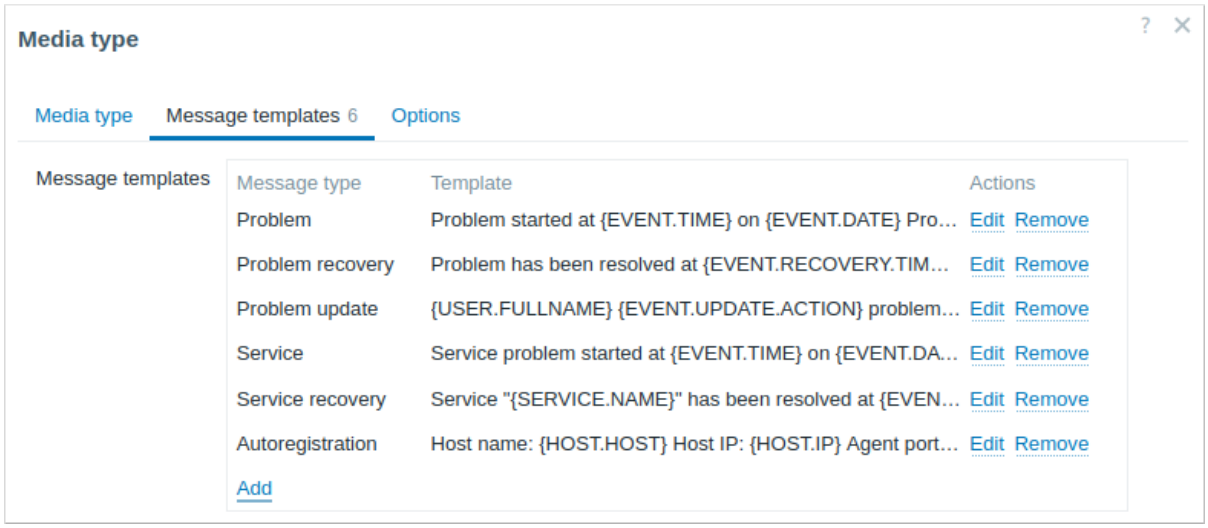



| Parameter | Description |
|---|---|
| *Name* | Имя типа оповещений. |
| *Type* | Выберите способ доставки для типа оповещений. |
| *Description* | Введите описание для типа оповещений. |
| *Enabled* | Установите флажок, чтобы включить тип оповещений. |

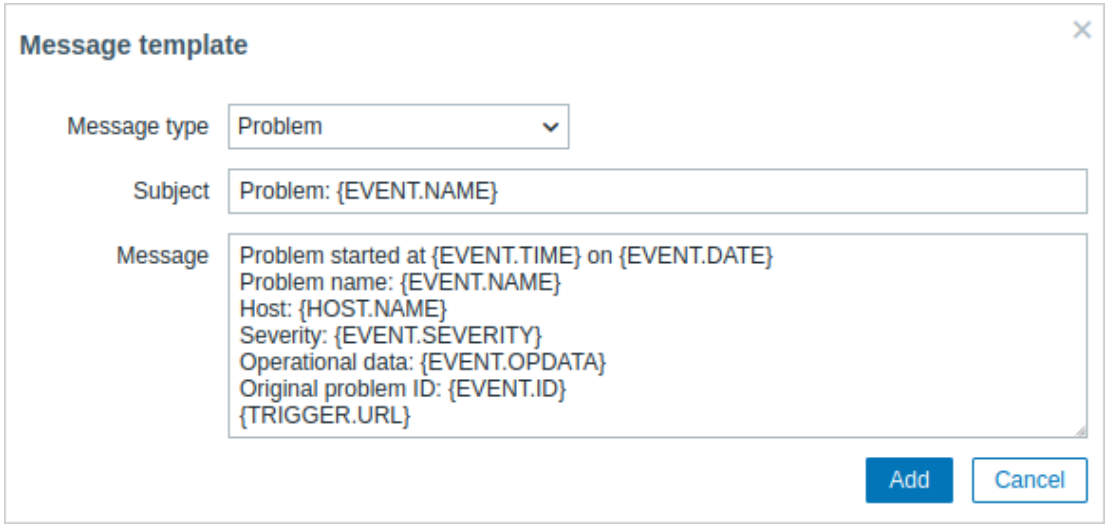
Для параметров, зависящих от способа доставки, см. страницы email, SMS, custom alertscript или вебхук.

Шаблоны сообщений

Вкладка **Шаблоны сообщений** содержит сообщения по умолчанию для каждого типа события (проблема, восстановление проблемы, обнаружение и т. д.).




Нажмите **Добавить**, чтобы определить сообщение по умолчанию (или **Изменить**, чтобы обновить существующее сообщение):




| Parameter | Description |
|---|---|
| *Message type* | Тип события, для которого должно использоваться сообщение по умолчанию.<br>Для каждого типа события можно определить только одно сообщение по умолчанию. |
| *Subject* | Тема сообщения по умолчанию. Тема может содержать макросы. Ограничение — 255 символов.<br>Тема недоступна для типа носителя SMS. |
| *Message* | Сообщение по умолчанию. Его длина ограничена определенным количеством символов в зависимости от типа базы данных (см. Отправка сообщений для получения дополнительной информации).<br>Сообщение может содержать поддерживаемые макросы.<br>В сообщениях о проблеме и обновлении проблемы поддерживаются макросы выражений (например, `{?avg(/host/key,1h)}`). |

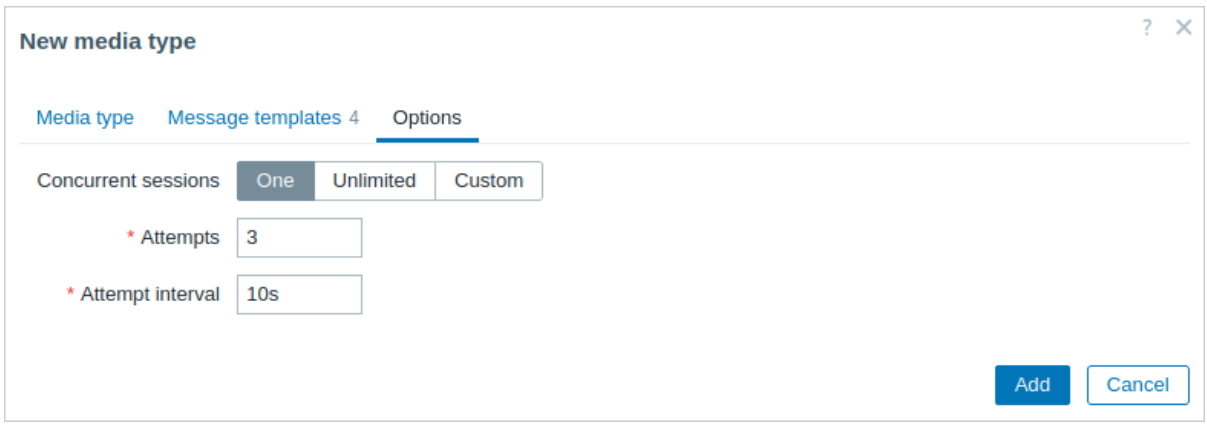
Обратите внимание, что сообщения по умолчанию переопределяются пользовательскими сообщениями, если они определены в операциях действия.

**Warning:**
Определение шаблонов сообщений обязательно для всех способов доставки, включая вебхуки или пользовательские скрипты оповещений, которые не используют сообщения по умолчанию для уведомлений. Например, действие "Send message to Pushover webhook" не сможет отправить уведомления о проблеме, если сообщение о проблеме для Pushover webhook не определено.

Опции

Вкладка **Опции (Options)** содержит настройки обработки оповещений. Этот набор параметров одинаков для всех способов оповещений.

Все способы оповещений обрабатываются параллельно. В то время как максимальное количество одновременных сессий настраивается для каждого способа оповещения по отдельности, общее количество процессов оповещения на сервере может быть ограничено только параметром StartAlerters parameter. Оповещения, сгенерированные одним триггером, обрабатываются последовательно. Несколько оповещений могут обрабатываться одновременно, только если они генерируются несколькими триггерами.



| Параметр | Описание |
|---|---|
| *Одновременные сессии (Concurrent sessions)* | Выберите количество параллельных сессий оповещения для способа оповещения:<br>**Одна (One)** — одна сессия<br>**Без ограничения (Unlimited)** — неограниченное количество сессий<br>**Пользовательское (Custom)** — введите настраиваемое количество сессий<br>Неограниченные/высокие значения означают большее количество параллельных сессий и увеличенную мощность при отправке оповещений. Неограниченное или большое количество сессий следует использовать в больших инфраструктурах, где может потребоваться отправка множества оповещений одновременно.<br>Если требуется отослать больше оповещений, чем имеется одновременных сессий, оставшиеся уведомления будут поставлены в очередь; они не потеряются. |
| *Попыток (Attempts)* | Количество попыток отправки оповещения. Можно указать до 100 попыток; значение по умолчанию «3». Если указано «1», Zabbix отправит оповещение только один раз и не будет повторять попытку в случае неудачной отправки. |
| *Интервал между попытками (Attempt interval)* | Частота попыток повторной отправки уведомления в случае неудачной отправки, в секундах (0-3600). Если указано «0», Zabbix выполнит новую попытку немедленно.<br>Поддерживаются суффиксы времени, например: 5s, 3m, 1h. |

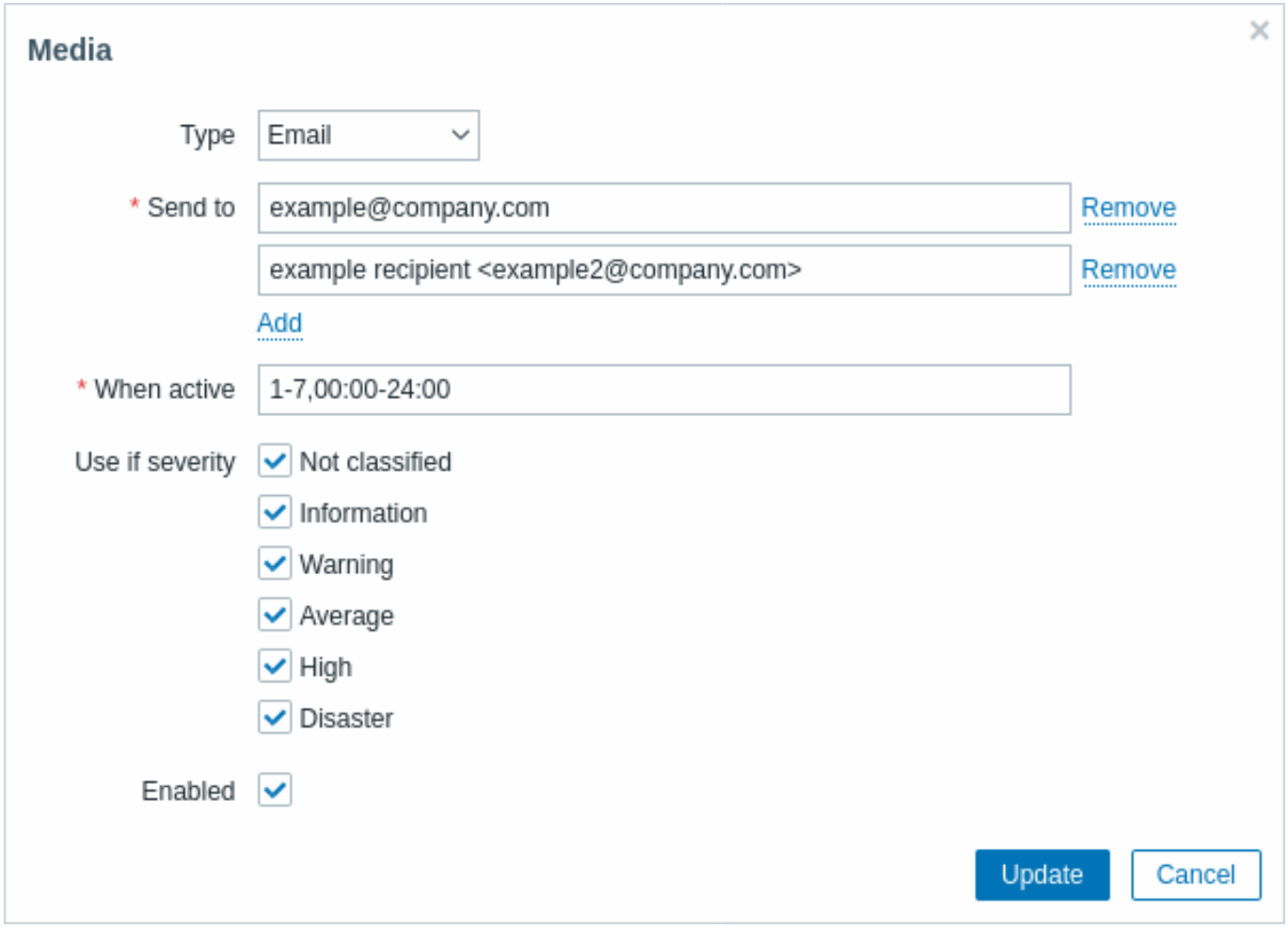
Пользовательские медиа

Если типы медиа определяют **как** будет отправлено уведомление, то пользовательские медиа определяют **куда** должно быть отправлено уведомление.

Пользовательские медиа (например, адрес электронной почты, ID пользователя вебхука и т. д.) должны быть указаны в профиле пользователя независимо от способа доставки. Действие, отправляющее сообщения пользователю *Admin* с использованием вебхука *X*, не сможет доставить их, если параметры доставки вебхука *X* не указаны в профиле пользователя Admin.

Чтобы задать пользовательские медиа:

- Перейдите в *Users* > *Users* и откройте форму свойств пользователя (или перейдите в *User settings* > *Notifications* своего профиля пользователя)
- Нажмите *Add* на вкладке Media




| Parameter | Description |
|---|---|
| *Type* | В раскрывающемся списке содержатся названия включенных типов медиа.<br>Обратите внимание: при редактировании медиа отключенного типа медиа его тип будет отображаться красным цветом. |
| *Send to* | Введите контактную информацию, куда следует отправлять сообщения.<br>Для типа медиа email можно добавить несколько адресов, нажав кнопку Add под полем адреса. В этом случае уведомления будут отправляться на все указанные адреса.<br>Примеры адресов см. в описании параметра *Email* для типа медиа email. |
| *When active* | Вы можете ограничить время отправки сообщений, например, задать только рабочие дни (1-5,09:00-18:00). Обратите внимание, что это ограничение основано на часовом поясе пользователя. Если часовой пояс пользователя изменен и отличается от системного часового пояса, это ограничение может потребовать соответствующей корректировки, чтобы не пропустить важные сообщения.<br>Описание формата см. на странице Спецификация временного периода.<br>Поддерживаются пользовательские макросы. |
| *Use if severity* | Отметьте флажками уровни серьезности триггеров, по которым вы хотите получать уведомления.<br>*Обратите внимание*, что уровень серьезности по умолчанию ('Not classified') **должен быть** отмечен, если вы хотите получать уведомления о событиях, не связанных с триггерами.<br>После сохранения выбранные уровни серьезности триггеров будут отображаться соответствующими цветами серьезности, а невыбранные будут отображаться серым цветом. |
| *Status* | Статус пользовательских медиа.<br>**Enabled** - используется.<br>**Disabled** - не используется. |

**1. Email**

Обзор

Чтобы настроить email в качестве канала доставки сообщений, вам необходимо настроить email как способ оповещений и назначить соответствующие адреса пользователям.

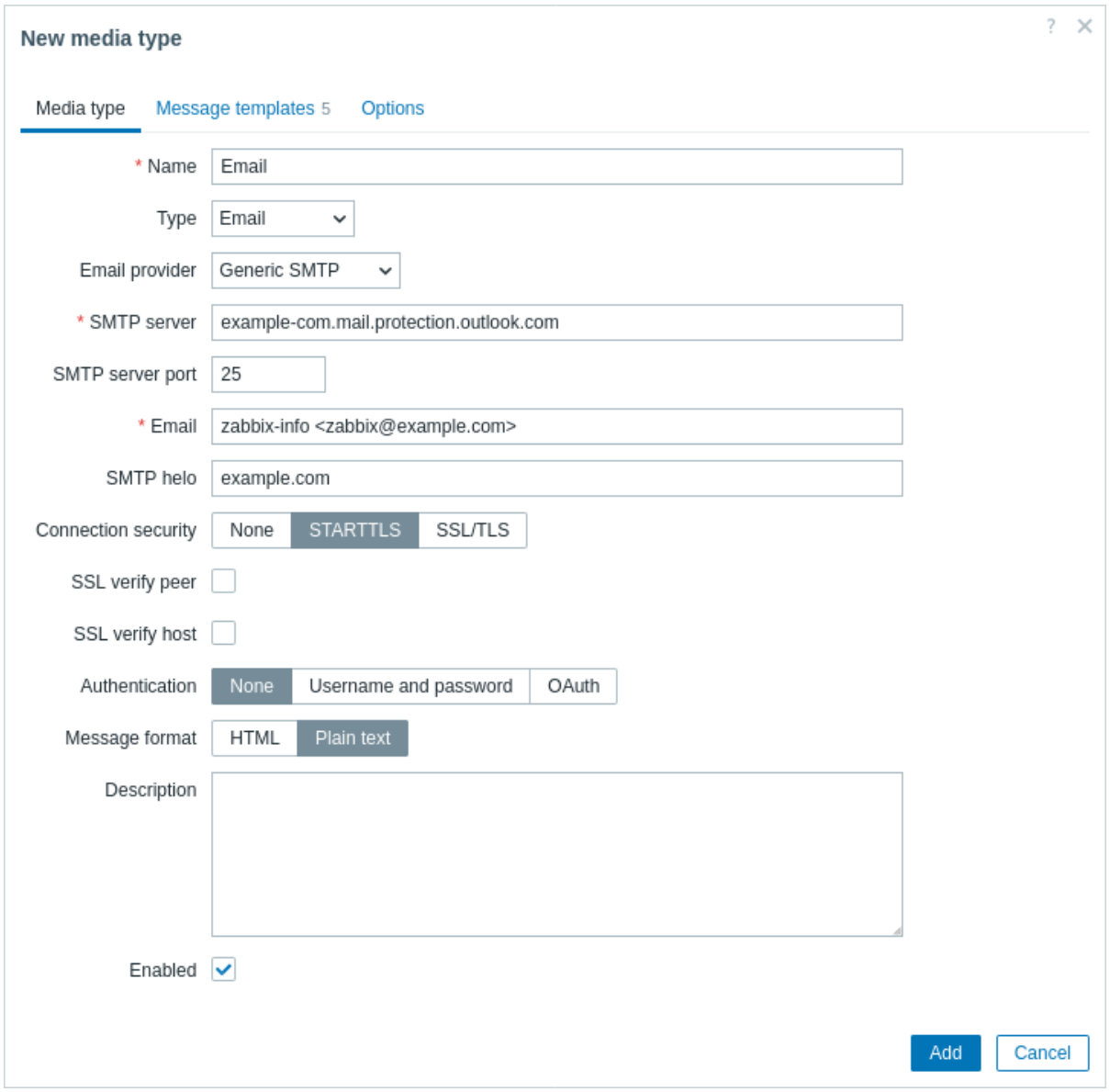
**Note:**
Несколько оповещений по одному событию группируются вместе в одной ветке email сообщений.

Конфигурация

Чтобы настроить email как тип медиа:

1. Перейдите в *Alerts* > *Media types*.
2. Нажмите *Create media type* (или нажмите *Email* в списке предопределенных типов медиа).

Вкладка **Media type** содержит общие атрибуты типа медиа:




Все обязательные поля ввода отмечены красной звездочкой.

**Warning:**
Пароль экспортируется в открытом виде при экспорте конфигурации типа медиа email.

Следующие параметры специфичны для типа медиа email:

| Parameter | Description |
|---|---|
| *Email provider* | Выберите поставщика email: *Generic SMTP*, *Gmail*, *Gmail relay*, *Office365* или *Office365 relay*.<br>Если вы выберете параметры, связанные с Gmail/Office365, вам нужно будет указать только адрес email отправителя и пароль; такие параметры, как *SMTP server*, *SMTP server port*, *SMTP helo* и *Connection security*, будут автоматически заполнены Zabbix. См. также: Automated Gmail/Office365 media types. |
| *SMTP server* | Укажите SMTP-сервер для обработки исходящих сообщений.<br>Это поле доступно, если в качестве поставщика email выбран *Generic SMTP*. |
| *SMTP server port* | Укажите порт SMTP-сервера для обработки исходящих сообщений.<br>Это поле доступно, если в качестве поставщика email выбран *Generic SMTP*. |
| *Email* | Адрес, указанный здесь, будет использоваться как адрес **From** для отправляемых сообщений.<br>Поддерживается добавление отображаемого имени отправителя (например, ”Zabbix_info” в *Zabbix_info <zabbix@company.com>* на скриншоте выше) вместе с фактическим адресом email.<br>В Zabbix для отображаемых имен в email действуют некоторые ограничения по сравнению с тем, что разрешено RFC 5322, как показано в примерах:<br>Допустимые примеры:<br>*zabbix@company.com* (только адрес email, угловые скобки не нужны)<br>*Zabbix_info <zabbix@company.com>* (отображаемое имя и адрес email в угловых скобках)<br>*∑Ω-monitoring <zabbix@company.com>* (символы UTF-8 в отображаемом имени)<br>Недопустимые примеры:<br>*Zabbix HQ zabbix@company.com* (есть отображаемое имя, но нет угловых скобок вокруг адреса email)<br>*”Zabbix\@\<H(comment)Q\>” <zabbix@company.com>* (хотя это допустимо по RFC 5322, экранированные пары и комментарии не поддерживаются в email Zabbix) |
| *SMTP helo* | Укажите корректное значение SMTP helo, обычно это доменное имя.<br>Если поле пустое, будет отправлено доменное имя email (то есть часть после @ в поле *Email*). Если получить доменное имя невозможно, в журнал будет записано предупреждение уровня debug, а в качестве домена для команды HELO будет отправлено имя хоста сервера.<br>Это поле доступно, если в качестве поставщика email выбран *Generic SMTP*. |
| *Connection security* | Выберите уровень безопасности соединения:<br>**None** - не использовать опцию CURLOPT_USE_SSL<br>**STARTTLS** - использовать опцию CURLOPT_USE_SSL со значением CURLUSESSL_ALL<br>**SSL/TLS** - использование CURLOPT_USE_SSL необязательно |
| *SSL verify peer* | Установите флажок, чтобы проверить SSL-сертификат SMTP-сервера.<br>Значение серверной директивы конфигурации ”SSLCALocation” должно быть передано в CURLOPT_CAPATH для проверки сертификата.<br>Это задает параметр cURL CURLOPT_SSL_VERIFYPEER. |
| *SSL verify host* | Установите флажок, чтобы проверить, что поле *Common Name* или поле *Subject Alternate Name* сертификата SMTP-сервера совпадает.<br>Это задает параметр cURL CURLOPT_SSL_VERIFYHOST. |
| *Authentication* | Выберите уровень аутентификации:<br>**None** - параметры cURL не задаются<br>**Username and password** - подразумевает ”AUTH=*”, оставляя выбор механизма аутентификации на усмотрение cURL<br>**OAuth** - аутентификация OAuth<br>Аутентификация OAuth не поддерживается для поставщика email *Office365 relay*. |
| *Username* | Имя пользователя, используемое для аутентификации.<br>Это задает значение CURLOPT_USERNAME.<br>Поддерживаются макросы пользователя. |
| *Password* | Пароль, используемый для аутентификации.<br>Это задает значение CURLOPT_PASSWORD.<br>Поддерживаются макросы пользователя.<br>Значение, введенное здесь, экспортируется как есть при экспорте конфигурации типа медиа. Чтобы не раскрывать конфиденциальные учетные данные в экспортируемых файлах, используйте макрос пользователя (например `{$EMAIL_SMTP_PASSWORD}`) вместо буквального пароля — ссылка на макрос будет экспортирована, а секретное значение должно быть (пере)настроено на целевой системе. |
| *OAuth tokens* | Нажмите *Configure*, чтобы в новом окне настроить параметры для получения OAuth tokens.<br>Это поле доступно только если в поле *Authentication* выбрано ”OAuth”. |

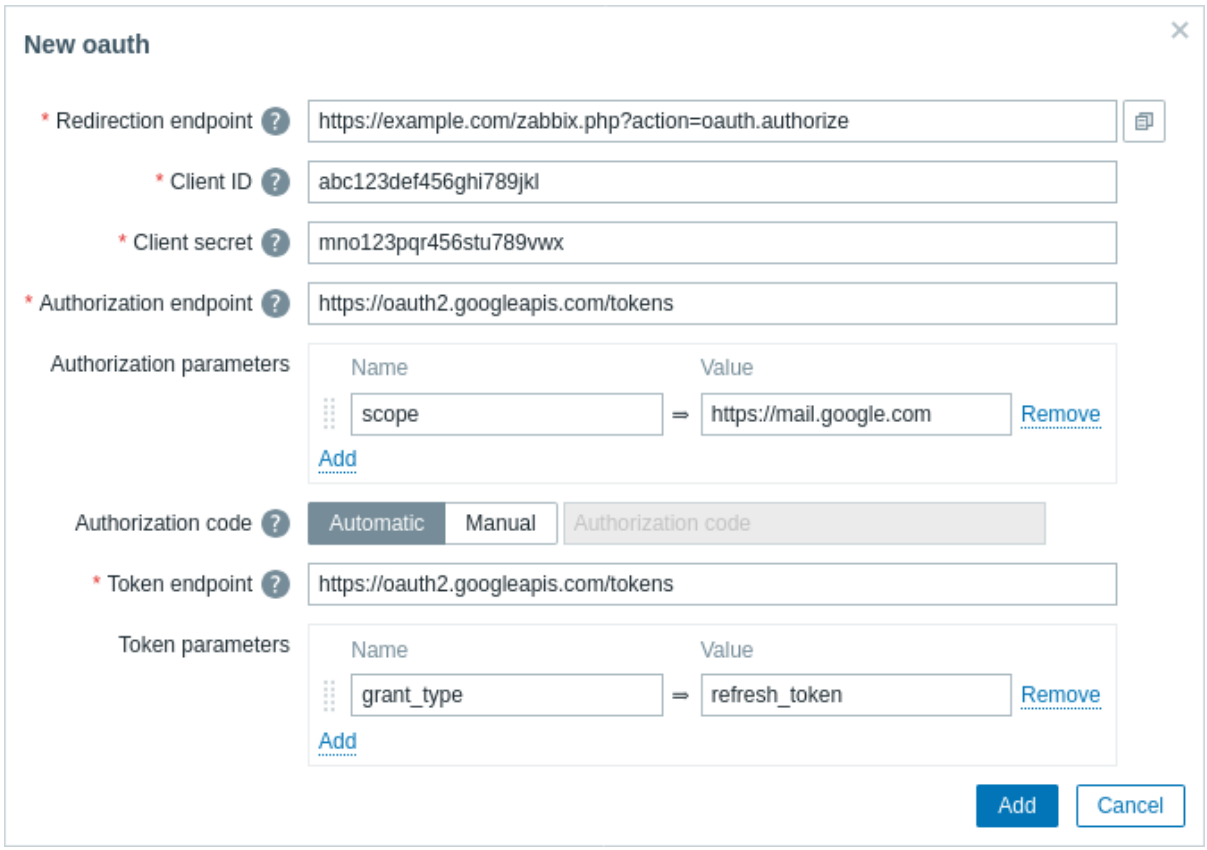
| Parameter | Description |
|---|---|
| *Message format* | Выберите формат сообщения:<br>**HTML** - отправлять как HTML<br>**Plain text** - отправлять как обычный текст |

**Attention:**
Чтобы включить SMTP-аутентификацию, сервер Zabbix должен быть скомпилирован с опцией компиляции `--with-libcurl`, для которой требуется libcurl (см. требования к сервер для подробностей о версиях).

См. также common media type parameters для получения подробностей о настройке сообщений по умолчанию и параметров обработки оповещений.

OAuth tokens

Для получения OAuth tokens требуются следующие параметры:




| Parameter | Description |
|---|---|
| *Redirection endpoint* | Введите URL веб-интерфейса Zabbix, на который служба OAuth будет перенаправлять ответ после авторизации OAuth (в формате `https://<zabbix-frontend-url>/zabbix.php?action=oauth.authorize`).<br>Он автоматически задается для нового типа медиа email, если определен URL веб-интерфейса. |
| *Client ID* | Введите уникальный идентификатор клиентского приложения, зарегистрированного на сервере авторизации OAuth. |
| *Client secret* | Введите закрытый секрет клиентского приложения, зарегистрированного на сервере авторизации OAuth. |
| *Authorization endpoint* | Введите URL сервера авторизации OAuth для запроса авторизации пользователя. |
| *Authorization parameters* | Введите параметры для конечной точки авторизации. |
| *Authorization code* | Введите код авторизации:<br>**Automatic** - код будет получен автоматически через страницу перенаправления<br>**Manual** - введите код вручную, если автоматическое получение не удалось |
| *Token endpoint* | Введите URL сервера авторизации OAuth для обмена кода авторизации на access и refresh tokens. |

| Parameter | Description |
|---|---|
| *Token parameters* | Введите параметры для access token. |

**Note:**
Получение OAuth tokens имеет автоматизированные возможности для почтовых провайдеров **Gmail**, **Gmail relay** и **Office365**. В этом случае требуется указать только значения параметров *Redirection endpoint*, *Client ID* и *Client secret*. Zabbix автоматически заполнит остальные необходимые значения (см. значения URL OAuth по умолчанию для провайдеров).

Получение access и refresh token OAuth

После отправки параметров OAuth:

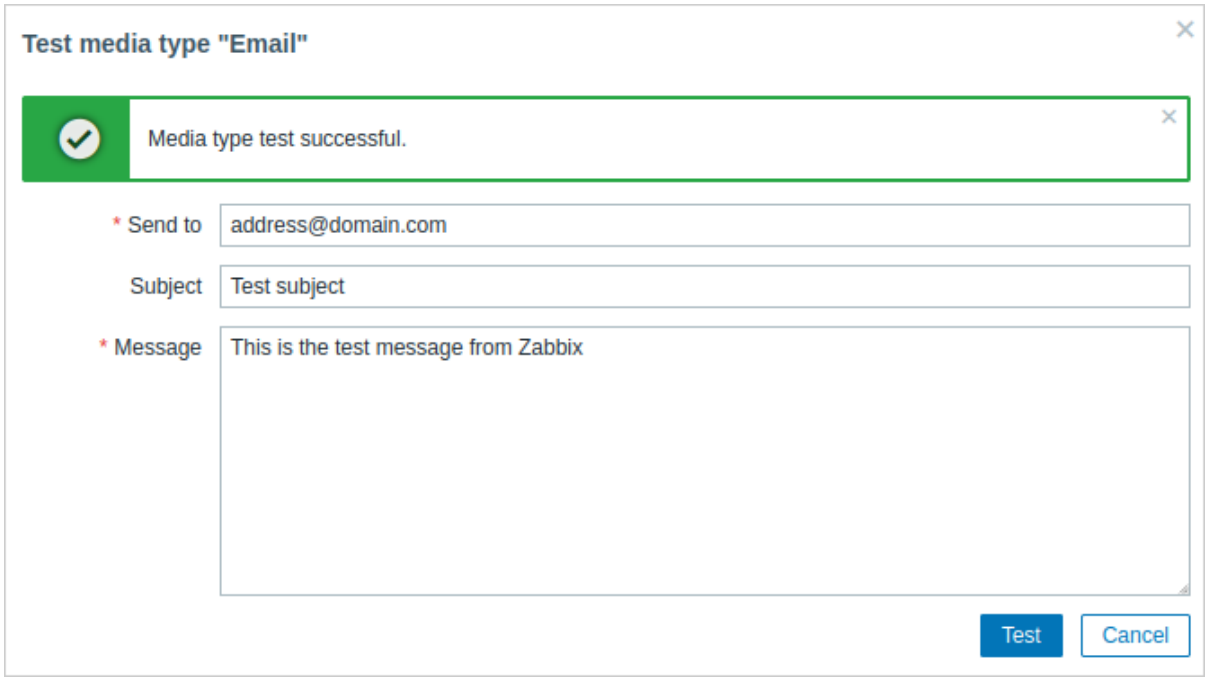
1. Открывается всплывающее окно браузера, которое перенаправляет пользователя к *Authorization endpoint*.
2. Пользователь авторизует Zabbix в службе OAuth.
3. Служба OAuth перенаправляет пользователя на действие веб-интерфейса Zabbix `oauth.authorize` с кодом авторизации и значением scope.
4. В ответ Zabbix отправит запрос к *Token endpoint*, чтобы обменять код авторизации на access и refresh token.

Тестирование

Чтобы проверить, что настроенный тип медиа для электронной почты работает корректно:

1. Найдите нужный тип медиа для электронной почты в списке типов медиа.
2. Нажмите *Test* в последнем столбце списка (откроется окно тестирования).
3. Укажите адрес получателя в поле *Send to*, текст сообщения и, при необходимости, тему.
4. Нажмите *Test*, чтобы отправить тестовое сообщение.

В том же окне будет отображено сообщение об успешном выполнении или ошибке теста:




Оповещение пользователей

После того, как email способ оповещения настроен, перейдите в раздел *Пользователи → Пользователи (Users → Users)* и измените профиль пользователя, назначьте оповещение по email этому пользователю. Шаги по настройке оповещений пользователей, общие для всех способов оповещений, описаны на странице Способы оповещений.

1 Автоматизированные типы медиа Gmail/Office365

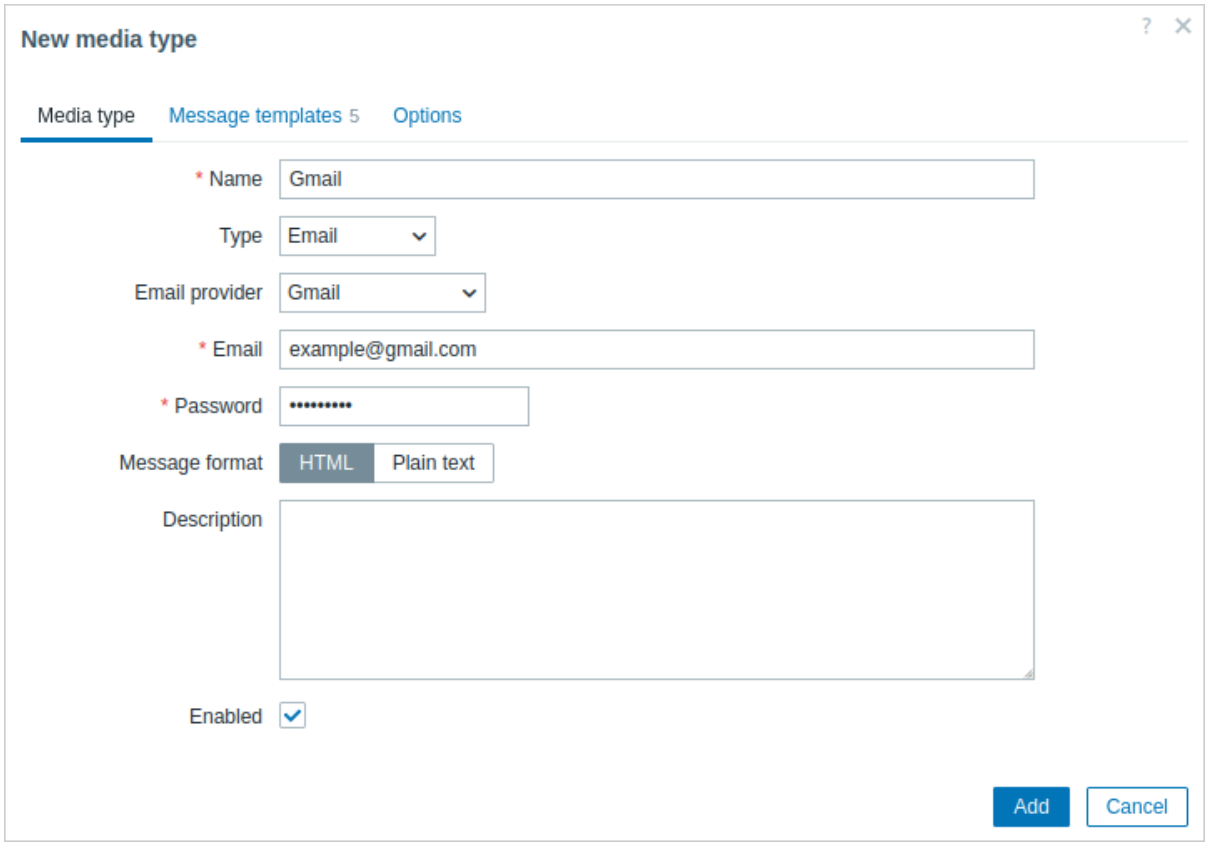
Обзор

Пользователи Gmail или Office365 могут воспользоваться автоматизированными функциями в настройке типа медиа.

Конфигурация

Поле *Поставщик электронной почты* в конфигурации типа медиа для электронной почты позволяет выбрать предварительно настроенные варианты для Gmail и Office 365.

При выборе вариантов, связанных с Gmail/Office365, для создания рабочего типа медиа требуется только указать адрес электронной почты отправителя и пароль.




Как только адрес электронной почты и пароль будут указаны, Zabbix сможет автоматически заполнить все необходимые настройки для типов медиа Gmail/Office365 фактическими/рекомендуемыми значениями, то есть *SMTP server*, *SMTP server port*, *SMTP helo* и *Connection security*. Благодаря этой автоматизации эти поля даже не отображаются, однако можно увидеть сведения о SMTP server и электронной почте в списке типов медиа (см. столбец *Details*).

Также обратите внимание, что:

- Пароль не требуется для вариантов relay.
- Для Office365 relay доменное имя указанного адреса электронной почты будет использоваться для динамического заполнения SMTP server (то есть вместо "example.com" в `example-com.mail.protection.outlook.com` будет подставлено реальное значение).

OAuth tokens

Получение OAuth tokens имеет автоматизированные функции для почтовых провайдеров *Gmail*, *Gmail relay* и *Office365*.

Для получения OAuth access token и refresh token необходимо указать только ограниченный набор параметров - *Redirection endpoint*, *Client ID* и *Client secret*. Zabbix автоматически заполнит остальные требуемые значения (см. OAuth URL defaults by provider).

Обратите внимание, что для настройки OAuth authorization для этих провайдеров также можно использовать форму Generic SMTP.

**Note:**
SmtpClientAuthentication должно быть включено либо для каждого пользователя/общего почтового ящика, либо для всего tenant в Office365 (по умолчанию в настройках не включено).

Значения OAuth URL по умолчанию в зависимости от провайдера

В следующей таблице перечислены значения OAuth URL по умолчанию и параметры для каждого провайдера, используемые Zabbix.

| Параметр | Gmail | Office365 | Generic SMTP |
|---|---|---|---|
| *Authorization end-point* | `https://accounts.google.com/o/oauth2/auth` | `https://login.microsoftonline.com/common/oauth2/v2.0/authori` | Нет значения по умолчанию |
| *scope* | `https://mail.google.com` | `https://outlook.office.com/SMTP.Send, offline_access` | Нет значения по умолчанию |
| *access_type* | `offline` | Не используется | Нет значения по умолчанию |
| *prompt* | `consent` | Не используется | Нет значения по умолчанию |
| *redirect_uri* | Значение поля *Redirection endpoint* автоматически добавляется кодом. Не указано в параметрах *Authorization endpoint*. | | |
| *state* | Уникальный хэш для идентификации медиа, обновляемого с помощью OAuth token, автоматически добавляется кодом. Не указано в параметрах *Authorization endpoint*. | | |
| *Token end-point* | `https://oauth2.googleapis.com/token` | `https://login.microsoftonline.com/common/oauth2/v2.0/token` | Нет значения по умолчанию |
| *grant_type* | `authorization_code` | `authorization_code` | Нет значения по умолчанию |
| *redirect_uri* | Значение поля *Redirection endpoint* автоматически добавляется кодом. Не указано в параметрах *Token endpoint*. | | |
| *state* | Уникальный хэш для идентификации медиа, обновляемого с помощью OAuth token, автоматически добавляется кодом. Не указано в параметрах *Token endpoint*. | | |

Обходное решение для личных учетных записей Office365

Личные учетные записи Office365 не поддерживают строки запроса в URL перенаправления.

Чтобы обойти эту проблему, администратор веб-сервера Apache может добавить следующее правило переписывания в файл `.htaccess`:

```
RewriteEngine On
RewriteRule ^/oauth\.authorize$ /zabbix.php?action=oauth.authorize [QSA,L,PT]
```

Это позволяет использовать упрощенный URL перенаправления, например:

```
http://server-name/zabbix/oauth.authorize
```

вместо URL с параметрами запроса, обеспечивая совместимость с личными учетными записями Office365.

### 2. SMS

Обзор

Zabbix поддерживает отправку SMS сообщений с использованием GSM модемов, подключённых к последовательному порту Zabbix сервера.

Убедитесь, что:

- Скорость последовательного устройства (обычно /dev/ttyS0 под Linux) совпадает со скоростью GSM-модема. Zabbix не устанавливает скорость для последовательного соединения. Он использует настройки по умолчанию.
- Пользователь «zabbix» имеет доступ на чтение/запись к последовательному устройству. Выполните команду ls -l /dev/ttyS0 для просмотра текущих прав доступа к последовательному устройству.
- На GSM модеме введён ПИН-код, и он сохраняется даже после перезагрузки. В качестве альтернативного варианта, можно отключить запрос ПИН-кода на самой SIM карте. ПИН-код можно ввести путём использования команды AT+CPIN="NNNN" (где NNNN — это ваш ПИН-код, кавычки должны присутствовать) в терминальном приложении, таком как Unix minicom или Windows HyperTerminal.

Zabbix был протестирован на следующих GSM модемах:

- Siemens MC35
- Teltonika ModemCOM/G10

Для настройки SMS как канала отправки сообщений, вам также необходимо настроить SMS в качестве способа оповещений и ввести соответствующие номера телефонов для пользователей.

Конфигурация

Чтобы настроить SMS как тип оповещения:

- Перейдите в *Оповещения → Типы оповещений*
- Нажмите *Создать тип оповещения* (или нажмите *SMS* в списке предопределенных типов оповещений).

Следующие параметры специфичны для типа оповещения SMS:

| Parameter | Description |
|---|---|
| *GSM modem* | Укажите имя последовательного устройства GSM-модема.<br>Путь, введенный здесь, будет проверен по параметру сервера SMSDevices (если он указан). |

См. общие параметры типа оповещения для подробностей о настройке сообщений по умолчанию и параметров обработки оповещений. Обратите внимание, что параллельная обработка отправки SMS-уведомлений невозможна.

Оповещение пользователей

После того, как SMS способ оповещения настроен, перейдите в раздел *Пользователи → Пользователи (Users →Users)* и измените профиль пользователя, назначьте оповещение по SMS каналу этому пользователю. Шаги по настройке оповещений пользователей, общие для всех способов оповещений, описаны на странице Способов оповещений.

### 3. Пользовательские скрипты оповещений

Обзор

Если вы не удовлетворены существующими способами оповещений для отправки оповещений, то вы можете попробовать альтернативный способ отправки оповещений. Вы можете написать скрипт, который будет обрабатывать оповещение вашим способом.

Пользовательские скрипты оповещений выполняются на стороне Zabbix сервера. Эти скрипты должны быть размещены в директории, которая указывается в параметре AlertScriptsPath в файле конфигурации сервера.

Пример пользовательского скрипта оповещения:

```
#####!/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 проверяет код завершения выполненных команд и скриптов. Любой код, отличный от **0**, рассматривается как ошибка выполнения команды. В этом случае Zabbix попытается снова выполнить команду, которая завершилась с ошибкой.

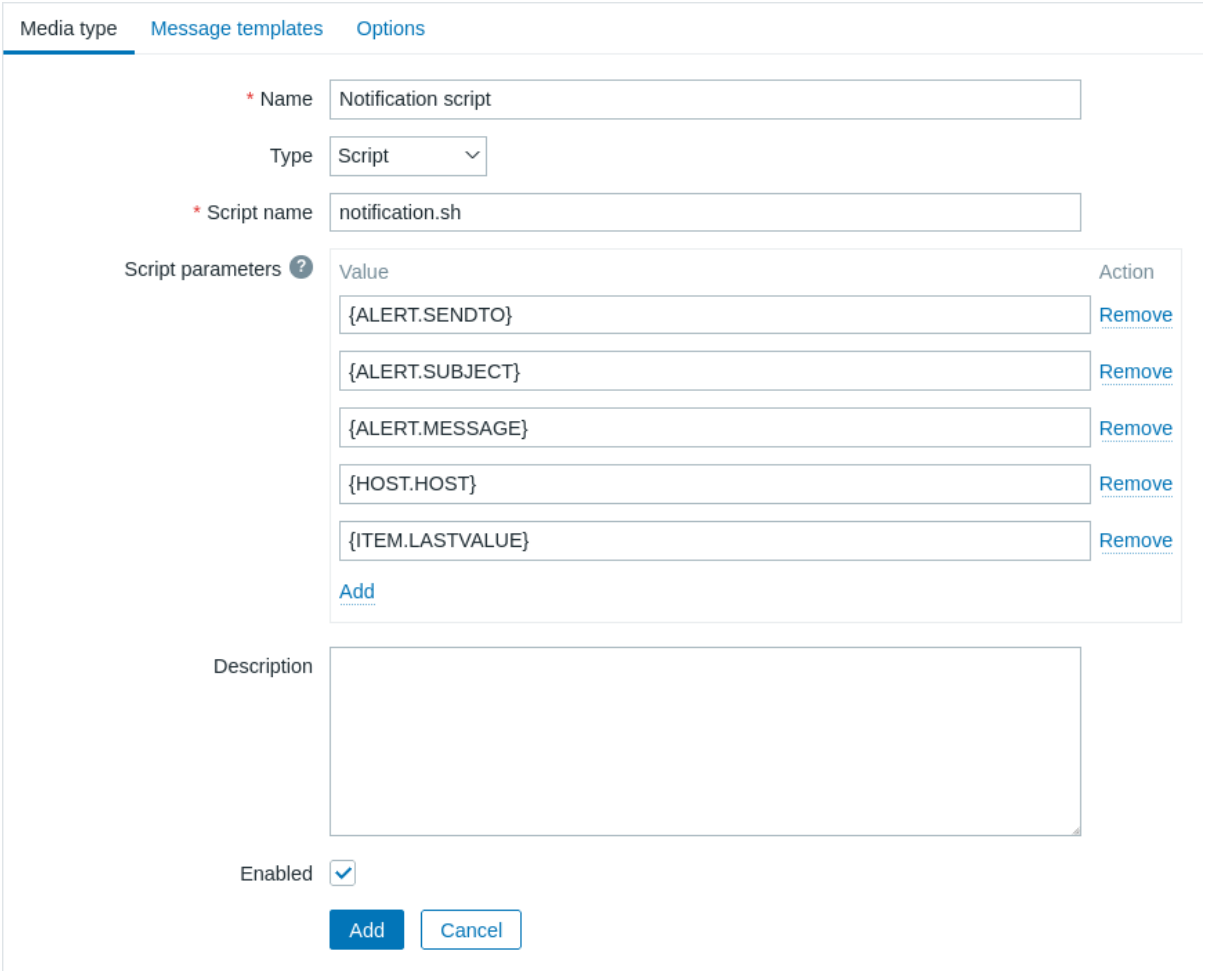
Переменные среды не сохраняются и не создаются для скрипта, поэтому их необходимо обрабатывать в явном виде.

Конфигурация

Чтобы настроить пользовательские скрипты оповещений как тип медиа:

1. Перейдите в *Оповещения → Типы медиа*.
2. Нажмите *Создать тип медиа*.

Вкладка **Тип медиа** содержит общие атрибуты типа медиа:




Все обязательные поля ввода отмечены красной звездочкой.

Следующие параметры специфичны для типа медиа "скрипт":

| Parameter | Description |
|---|---|
| *Script name* | Введите имя файла скрипта (например, notification.sh), который находится в каталоге, указанном в параметре конфигурации сервера AlertScriptsPath. |

| Parameter | Description |
|---|---|
| *Script parameters* | Добавьте необязательные параметры скрипта, которые будут переданы скрипту как аргументы командной строки в порядке их определения.<br><br>Параметры скрипта поддерживают макросы {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} и все макросы, поддерживаемые в уведомлениях, а также пользовательские макросы. |

См. общие параметры типа медиа для получения сведений о том, как настроить сообщения по умолчанию и параметры обработки оповещений.

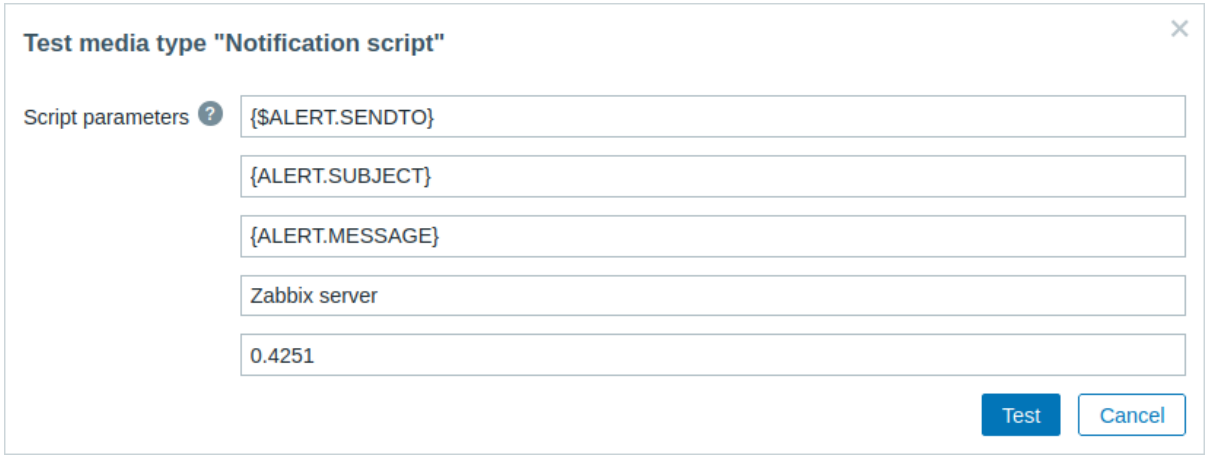
**Warning:**
Даже если скрипт оповещения не использует сообщения по умолчанию, шаблоны сообщений для типов операций, используемых этим типом медиа, все равно должны быть определены. В противном случае уведомление не будет отправлено.

**Attention:**
Если настроено более одного типа медиа "скрипт", эти скрипты могут обрабатываться параллельно процессами оповещателя. Общее количество процессов оповещателя ограничено параметром `StartAlerters` в файле конфигурации сервера.

Тестирование

Чтобы протестировать настроенный тип медиа-элемента script:

1. Найдите нужный script в списке типов медиа-элементов.
2. Нажмите *Test* в последнем столбце списка; в всплывающем окне откроется форма тестирования. Форма тестирования будет содержать такое же количество параметров, которое настроено для типа медиа-элемента script.
3. При необходимости измените значения параметров script. Изменение влияет только на процедуру тестирования; фактические значения изменены не будут.
4. Нажмите *Test*.




**Note:**
При тестировании настроенного типа медиа-элемента script {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} и пользовательские макросы будут подставлены в свои значения, но макросы, связанные с событиями (например, {HOST.HOST}, {ITEM.LASTVALUE} и т. д.), подставлены не будут, так как во время тестирования нет связанного события, из которого можно получить сведения. Обратите внимание, что макросы внутри макросов {ALERT.SUBJECT} и {ALERT.MESSAGE} также не будут подставлены. Например, если значение {ALERT.SUBJECT} состоит из "Problem: {EVENT.NAME}", то макрос {EVENT.NAME} не будет подставлен.

Оповещение пользователей

После того, как способ оповещения настроен, перейдите в раздел *Пользователи → Пользователи (Users → Users)* и измените профиль пользователя, назначьте ему адресата по этому способу оповещения. Шаги по настройке оповещений пользователей, общие для всех способов оповещений, описаны на странице Способов оповещений.

Обратите внимание, когда настраиваете оповещение пользователя, поле *Отправлять на (Send to)* не может быть пустым. Если это поле не используется в скрипте оповещений, введите любую комбинацию поддерживаемых символов для обхода требования наличия адресата.

#### 4. Вебхук

Обзор

Этот способ оповещений полезен для выполнения вызовов HTTP с использованием пользовательского кода JavaScript для прямой интеграции с внешними системами, такими как системы поддержки, чаты или мессенджеры. Вы можете выбрать — импортировать интеграцию, поставляемую Zabbix, или создать свою собственную интеграцию с нуля.

Интеграции

Доступны следующие интеграции, позволяющие использовать предопределенные типы медиа **вебхук** для отправки уведомлений Zabbix в:

- 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
- MS Teams Workflows
- LINE
- Opsgenie
- OTRS CE
- Pagerduty
- Pushover
- Redmine
- Rocket.Chat
- ServiceNow
- SIGNL4
- Slack
- SolarWinds
- SysAid
- Telegram
- TOPdesk
- VictorOps
- Zammad
- Zendesk

**Note:**
Помимо перечисленных здесь сервисов, Zabbix можно интегрировать с **Spiceworks** (вебхук не требуется). Чтобы преобразовать уведомления Zabbix в тикеты Spiceworks, создайте тип медиа электронной почты и укажите адрес электронной почты службы поддержки Spiceworks (например, help@zabbix.on.spiceworks.com) в настройках профиля назначенного пользователя Zabbix.

Конфигурация

Чтобы начать использовать интеграцию с вебхуком:

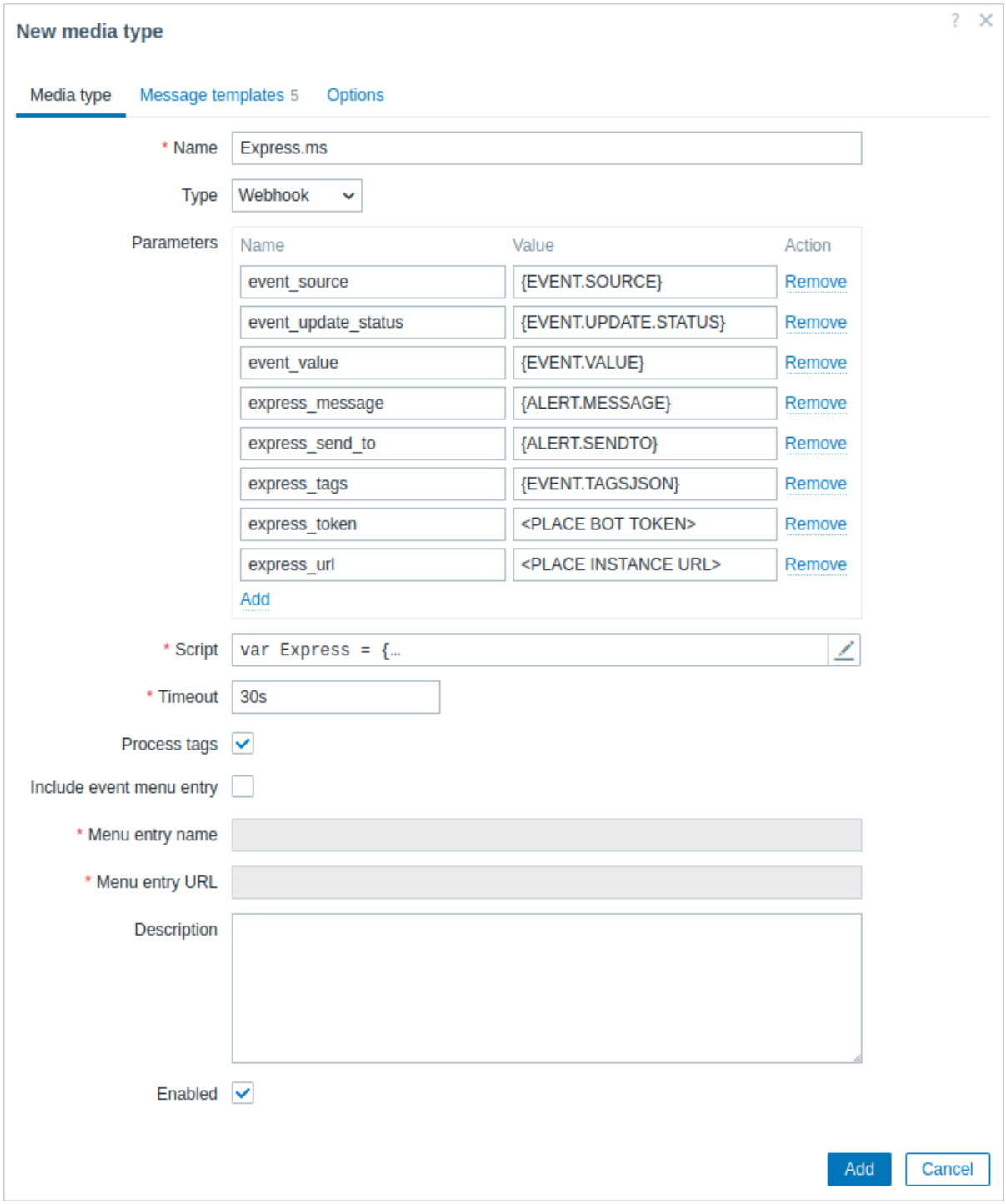
1. Найдите нужный файл .yaml в каталоге `templates/media` загруженной версии Zabbix или скачайте его из git-репозитория Zabbix.
2. Импортируйте файл в вашу установку Zabbix. Вебхук появится в списке типов медиа.

3. Настройте вебхук согласно инструкциям в файле *Readme.md* (вы можете щелкнуть по имени вебхука выше, чтобы быстро перейти к *Readme.md*).

Чтобы создать собственный вебхук с нуля:

1. Перейдите в *Оповещения* > *Типы медиа*.
2. Нажмите *Создать тип медиа*.

Вкладка **Тип медиа** содержит различные атрибуты, специфичные для этого типа медиа:




Все обязательные поля ввода отмечены красной звездочкой.

Следующие параметры специфичны для типа медиа вебхука:

| Параметр | Описание |
|---|---|
| *Parameters* | Укажите переменные вебхука в виде пар атрибутов и значений.<br>Для преднастроенных вебхуков список параметров зависит от сервиса. Смотрите описание параметров в файле *Readme.md* вебхука.<br>Для новых вебхуков по умолчанию включены несколько общих переменных (URL:<empty>, HTTPProxy:<empty>, To:{ALERT.SENDTO}, Subject:{ALERT.SUBJECT}, Message:{ALERT.MESSAGE}); при желании их можно оставить или удалить.<br><br>Параметры вебхука поддерживают пользовательские макросы, все макросы, поддерживаемые в уведомлениях о проблемах, а также макросы {ALERT.SENDTO}, {ALERT.SUBJECT} и {ALERT.MESSAGE}.<br><br>Если вы укажете HTTP-прокси, поле поддерживает ту же функциональность, что и поле HTTP proxy в настройке элемента данных. Перед строкой прокси можно указать префикс `[scheme]://`, чтобы задать тип используемого прокси (например, https, socks4, socks5; см. документацию). |
| *Script* | Введите код JavaScript в модальном редакторе, который открывается при щелчке в поле параметра или по значку карандаша рядом с ним. Этот код будет выполнять операцию вебхука.<br>Код представляет собой функцию, которая принимает пары параметр - значение. Значения следует преобразовать в объекты JSON с помощью метода JSON.parse(), например: `var params = JSON.parse(value);`.<br><br>Код имеет доступ ко всем параметрам, может выполнять HTTP-запросы GET, POST, PUT и DELETE, поддерживает дополнительные методы, такие как CONNECT, PATCH, HEAD, OPTIONS и TRACE, и управляет HTTP-заголовками и телом запроса.<br>Код должен содержать оператор return, иначе он не будет считаться корректным. Он может возвращать статус OK вместе с необязательным списком тегов и значений тегов (см. параметр *Process tags*) или строку ошибки.<br><br>Обратите внимание, что скрипт выполняется только после создания оповещения. Если в настройках скрипта указано возвращать и обрабатывать теги, эти теги не будут подставлены в макросы {EVENT.TAGS} и {EVENT.RECOVERY.TAGS} в исходном сообщении о проблеме и сообщениях о восстановлении, поскольку скрипт еще не успеет выполниться.<br>*Примечание*: Рекомендуется использовать локальные переменные (например, `var local = 1`) вместо глобальных (например, `global = 1`), чтобы каждый скрипт работал со своими данными и избежать конфликтов между одновременными вызовами (см. известные проблемы).<br><br>См. также: Руководство по разработке вебхуков, Примеры скриптов вебхуков, Дополнительные объекты JavaScript. |
| *Timeout* | Тайм-аут выполнения JavaScript (1-60 с, по умолчанию 30 с).<br>Поддерживаются суффиксы времени, например 30s, 1m. |
| *Process tags* | Установите флажок, чтобы обрабатывать возвращаемые значения свойств JSON как теги. Эти теги добавляются к любым существующим тегам проблемы.<br>Обратите внимание, что при использовании тегов вебхука вебхук должен возвращать объект JSON, содержащий как минимум пустой объект tags: `var result = {tags: {}};`<br>Примеры тегов, которые могут быть возвращены: *jira-id:prod-1234*, *responsible:John Smith*, *processed:<no value>* |
| *Include event menu entry* | Установите флажок, чтобы добавить в меню события пункт, ведущий к созданному внешнему тикету.<br>Пункт будет добавлен для каждого включенного вебхука, у которого установлен этот флажок. Обратите внимание, что если параметры *Menu entry name* и *Menu entry URL* содержат какие-либо макросы {EVENT.TAGS.<tag name>}, пункт будет добавлен только в том случае, если эти макросы можно подставить (то есть если у события определены такие теги).<br>Если флажок установлен, вебхук не следует использовать для отправки уведомлений разным пользователям (лучше создать отдельного пользователя) и не следует использовать его в нескольких действиях оповещения для одного события проблемы. |
| *Menu entry name* | Укажите имя пункта меню.<br>Поддерживается макрос {EVENT.TAGS.<tag name>}.<br>Это поле обязательно только если установлен флажок *Include event menu entry*. |

| Параметр | Описание |
|---|---|
| *Menu entry URL* | Укажите базовый URL пункта меню.<br>Поддерживается макрос {EVENT.TAGS.<tag name>}.<br>Это поле обязательно только если установлен флажок *Include event menu entry*. |

Подробнее о настройке сообщений по умолчанию и параметров обработки оповещений см. в разделе общие параметры типа медиа.

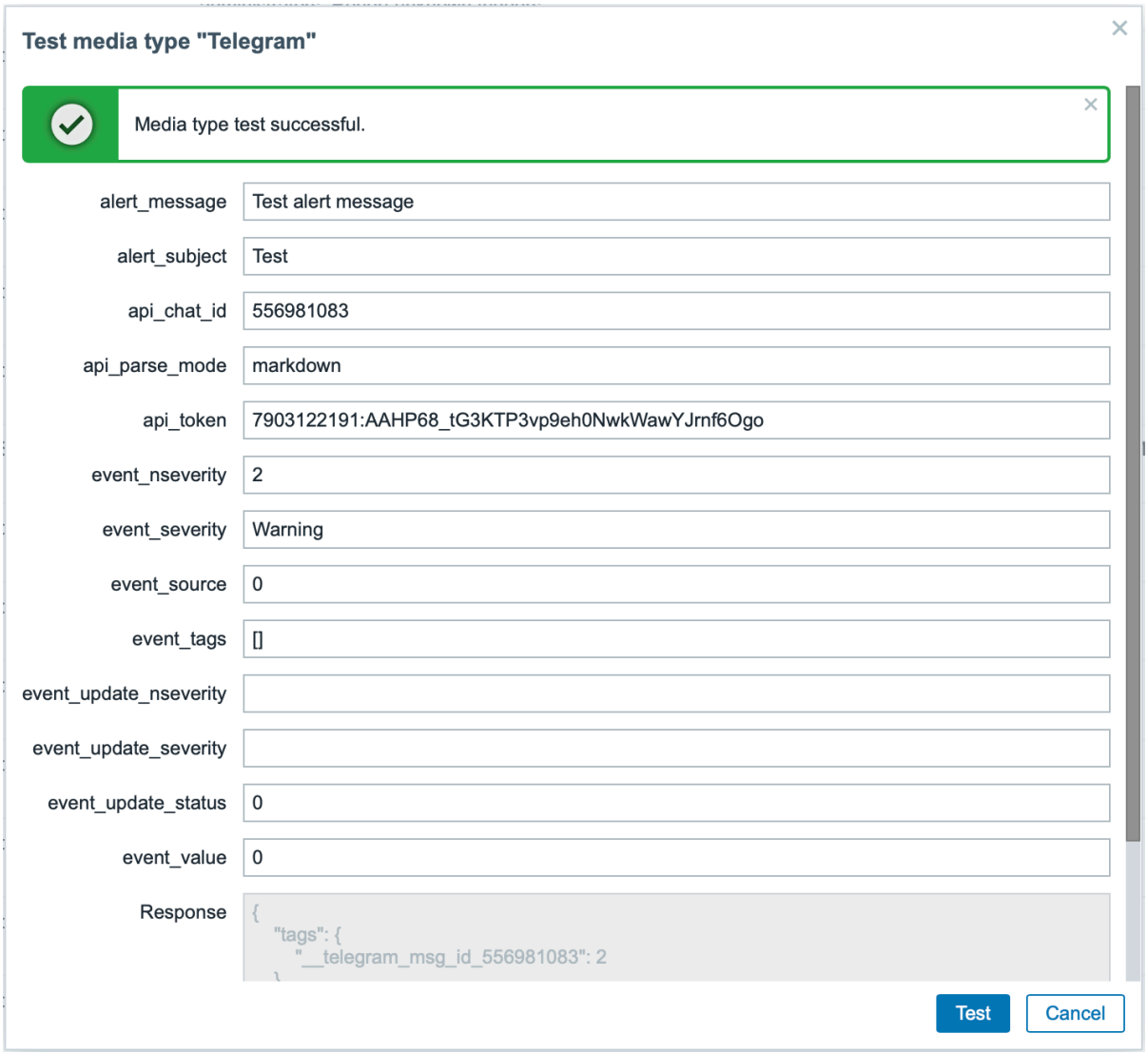
**Warning:**
Даже если вебхук не использует сообщения по умолчанию, шаблоны сообщений для типов операций, используемых этим вебхуком, все равно должны быть определены.

Тестирование

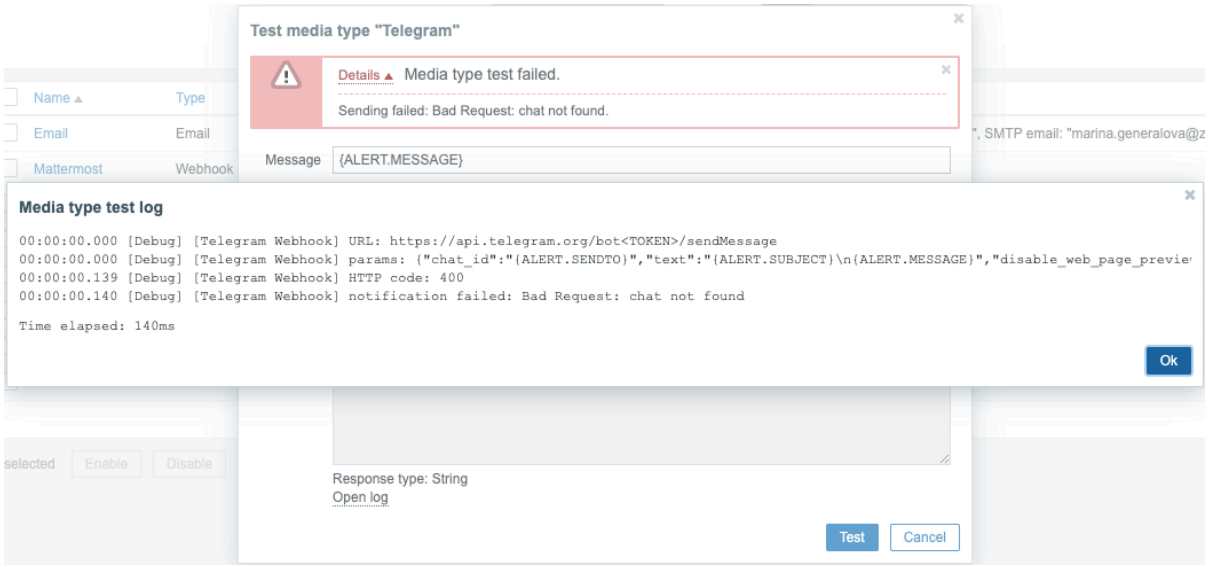
Чтобы протестировать настроенный тип медиа вебхука:

1. Найдите нужный вебхук в списке типов медиа.
2. Нажмите *Test* в последнем столбце списка (откроется окно тестирования).
3. При необходимости измените значения параметров вебхука. Замените макросы значениями-примерами; в противном случае макросы не будут разрешены, и тест завершится с ошибкой.
4. Нажмите *Test*.

Замена или удаление значений в окне тестирования влияет только на процедуру тестирования, фактические значения атрибутов вебхука останутся без изменений.




Чтобы просмотреть записи журнала тестирования типа медиа, не покидая окно тестирования, нажмите *Open log* (откроется новое всплывающее окно).




**Если проверка вебхука успешна:**

- Появится сообщение: *«Успешное тестирование способа оповещения (Media type test successful)»*.
- В сером поле *Ответ (Response)* отображается ответ сервера.
- Под полем *Ответ* указывается тип ответа («JSON» или «Строка (String)»).

**Если проверка вебхука неуспешна:**

- Отображается сообщение: *«Ошибка при тестировании способа оповещения (Media type test failed)»*, — с указанием дополнительных сведений.

Оповещение пользователей

После того, как способ оповещения настроен, перейдите в раздел *Пользователи → Пользователи (Users→ Users)* и измените профиль пользователя, назначив адресата вебхука по этому способу оповещения, или создайте нового пользователя для использования с этим вебхуком. Шаги по настройке оповещений пользователей, общие для всех способов оповещений, описаны на странице Способов оповещений.

Если вебхук использует теги для хранения идентификатора задачи (тикета) \ сообщения, избегайте назначение этого же вебхука в качестве способа оповещения разным пользователям, так как это может вызвать ошибки вебхука (применимо к большинству вебхуков, которые используют опцию *Добавить запись в меню события*). В этом случае лучше всего создать для вебхука выделенного пользователя:

1. После настройки способа оповещения вебхук, перейдите в раздел *Пользователи → Пользователи (Users→ Users)* и создайте выделенного пользователя Zabbix для вебхука — например, с именем пользователя *Slack* для вебхука Slack. Все настройки, за исключением способа оповещения, можно оставить по умолчанию, так как этот пользователь не будет входить в Zabbix.
2. В профиле пользователя перейдите на вкладку *Оповещения (Media)* и добавьте вебхук с необходимой контактной информацией. Если вебхук не использует поле *Отправлять на (Send to)*, введите любую комбинацию поддерживаемых символов для обхода требования наличия адресата.
3. Предоставьте этому пользователю по крайней мере права чтения ко всем узлам сети, по которым ему нужно отправлять оповещения.

При настройке действия на оповещение добавьте этого пользователя в поле *Отправка пользователям (Send to users)* в деталях Операции — это укажет Zabbix использовать вебхук для оповещений с этого действия.

Настройка действий на оповещения

Действия определяют, какие оповещения следует отправлять через вебхук. Шаги для настройки действий с использованием вебхуков абсолютно такие же, как и для всех остальных способов оповещения, за некоторыми исключениями:

- Если вебхук использует теги вебхуков [en] для хранения идентификатора задачи (тикета) \ сообщения и управляет операциями обновления\восстановления, этот вебхук не следует использовать в нескольких действиях на оповещения по одному и тому же событию на проблемы. Если {EVENT.TAGS.<имя>} существует и обновляется в вебхуке, то его окончательное значение будет не определено. Чтобы избежать этого, используйте в вебхуке новое имя тега для хранения обновлённых значений. Это относится к вебхукам Jira, Jira Service Desk, Mattermost, Opsgenie, OTRS, Redmine, ServiceNow, Slack, Zammad и Zendesk, которые поставляются Zabbix, и к большинству вебхуков, которые используют опцию *Добавить запись в меню события (Include event menu entry)*. Однако, обратите внимание, что один вебхук может быть использован в нескольких операциях или шагах эскалации того

же действия, а также в различных действиях, которые не будут срабатывать по тому же событию проблемы из-за отличающихся условий.

- При использовании вебхука в действиях на внутренние события убедитесь, что в настройках операций действия отмечена опция *Пользовательское сообщение (Custom message)* и указано пользовательское сообщение. В противном случае оповещение отправлено не будет.

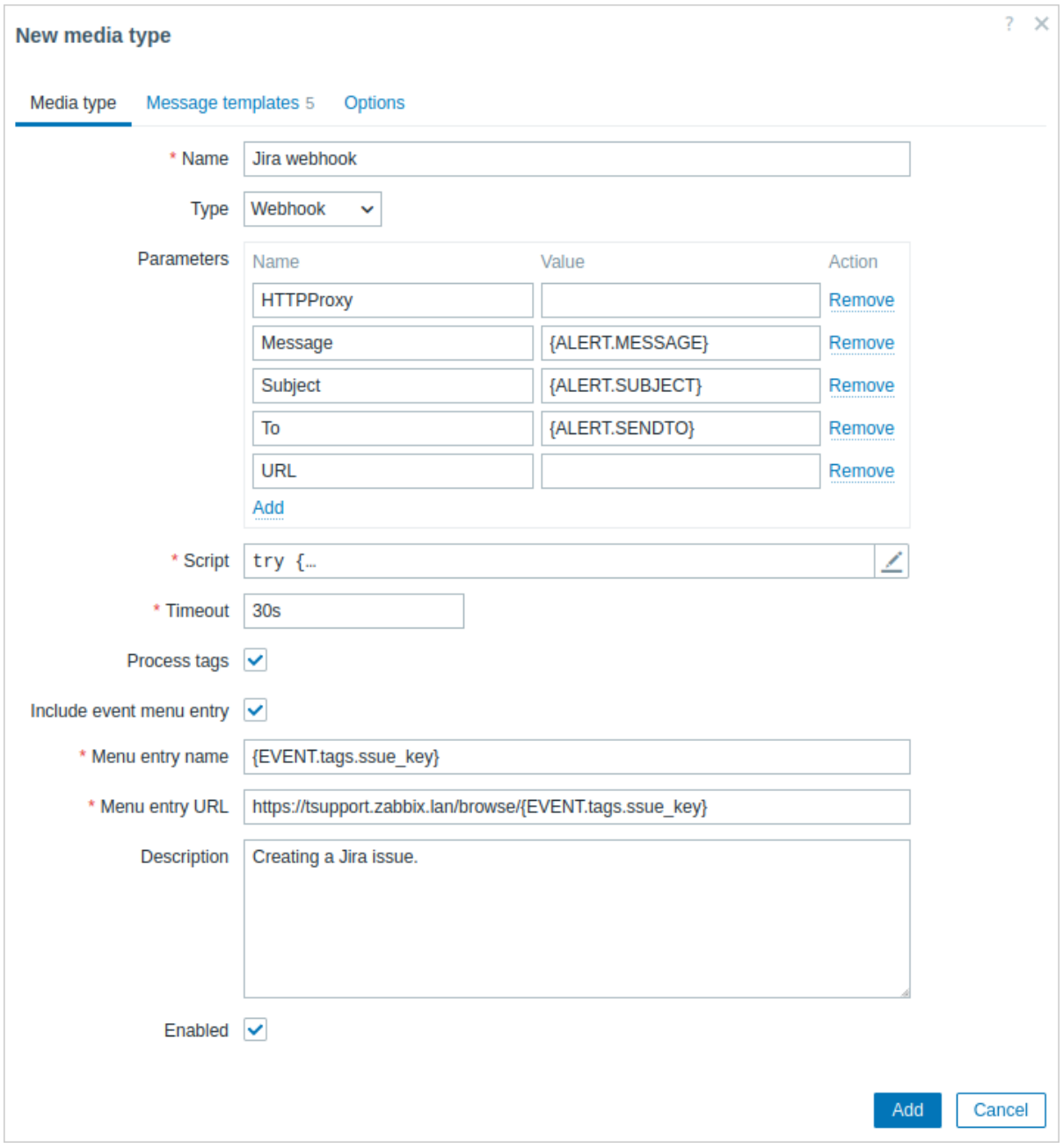
1. Примеры скриптов вебхуков

Обзор

Хоть Zabbix и предлагает большое количество интеграций вебхуков «из коробки», вы можете захотеть вместо них создать свои вебхуки. В этом разделе приведены примеры пользовательских скриптов вебхуков (используются в параметре *Скрипт*). Смотрите раздел вебхуков для получения более подробной информации по остальным параметрам вебхуков.

**Attention:**
Не используйте необъявленные присваивания в предварительной обработке JavaScript. Используйте `var` для объявления локальных переменных.

Jira вебхук (пользовательский)

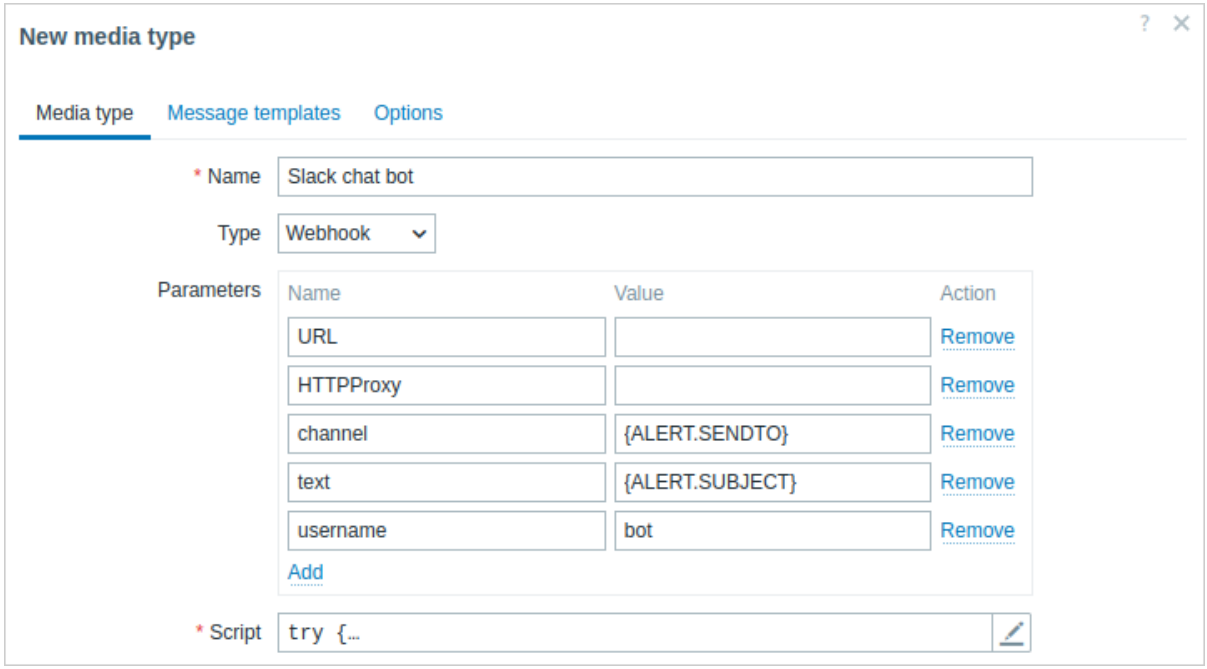



Этот скрипт создаст проблему (issue) в JIRA и вернёт некоторую информацию о созданной проблеме.

```
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 вебхук (пользовательский)

Этот вебхук будет пересылать оповещения из Zabbix в канал 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.hook_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;
}
```

## 2. Действия

Обзор

Если вы хотите, чтобы в результате событий выполнялись некоторые операции (например, отправка оповещений), вам нужно настроить действия.

Действия можно задать для событий со всех поддерживаемых источников:

- Действия на триггеры — на события при изменениях состояния триггеров с *OK* на *ПРОБЛЕМА* и обратно
- Действия на услуги — на события при изменениях состояния услуги с *OK* на *ПРОБЛЕМА* и обратно
- Действия на обнаружения — на события, когда производится сетевое обнаружение
- Действия на авторегистрацию — на события автоматической регистрации новых активных агентов (или при изменениях в метаданных узла сети, который уже был зарегистрирован)
- Внутренние действия — на события, когда элементы данных становятся неподдерживаемыми или триггеры переходят в неизвестное состояние

Ключевые отличия действий на услуги:

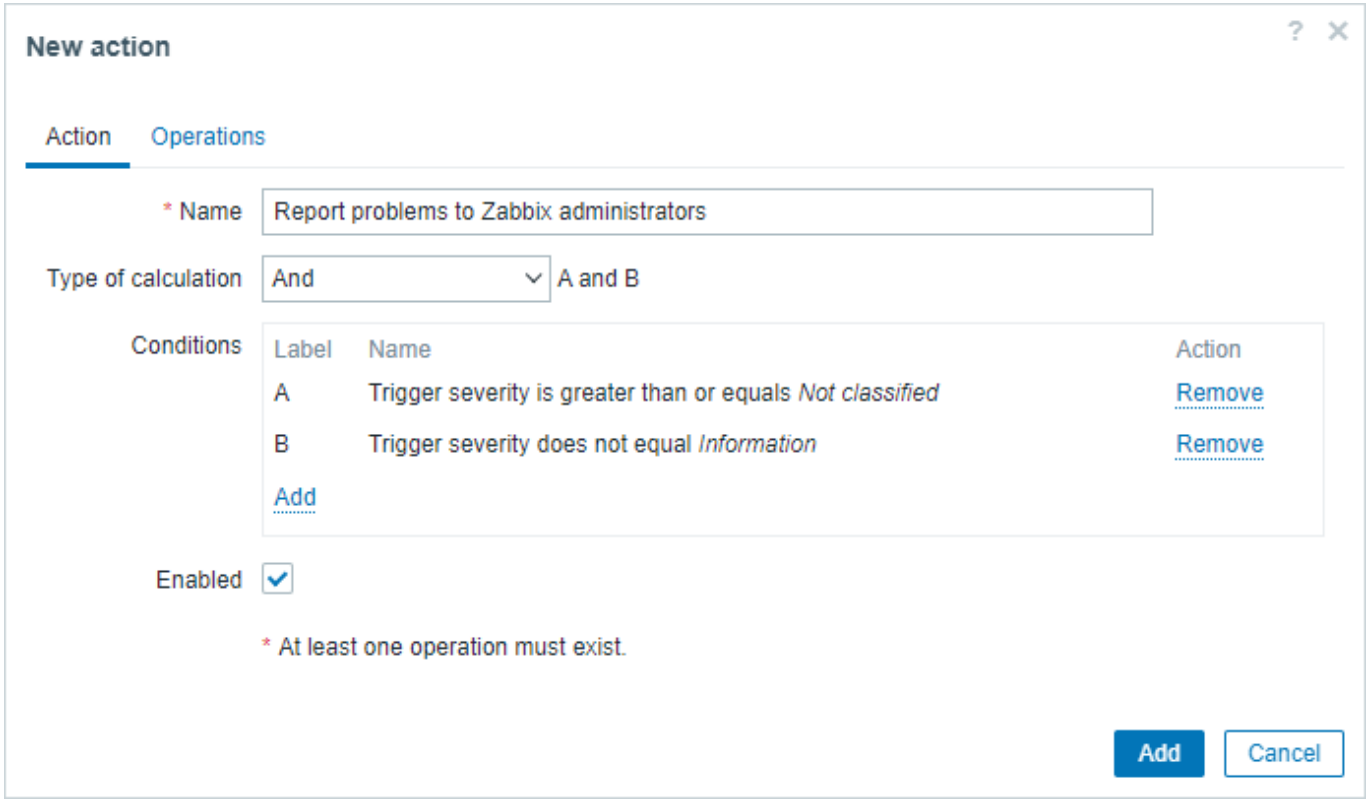
- Доступ пользователя к действиям на услуги зависит от прав доступа к услугам, предоставленных ролью пользователя
- Действия на услуги поддерживают другой набор условий

Настройка действия

Чтобы настроить действие, выполните следующее:

- Перейдите в *Оповещения* > *Действия* и выберите нужный тип действия в подменю (позже можно переключиться на другой тип, используя выпадающий список в заголовке).
- Нажмите *Создать действие*.
- Укажите имя действия.
- Выберите условия, при которых будут выполняться операции.
- Выберите операции, которые нужно выполнить.

Общие атрибуты действия:




Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Name* | Уникальное имя действия. |

| Parameter | Description |
|---|---|
| *Type of calculation* | Выберите вариант оценки условий действия (если условий больше одного):<br>**And** - должны быть выполнены все условия.<br>**Or** - достаточно, чтобы было выполнено одно условие.<br>**And/Or** - комбинация двух вариантов: AND для разных типов условий и OR для одинакового типа условий.<br>**Custom expression** - пользовательская формула вычисления для оценки условий действия. |
| *Conditions* | Список условий действия.<br>Нажмите *Add*, чтобы добавить новое условие.<br>Если условия не настроены, действие будет выполняться для каждого события, которое соответствует настраиваемому типу действия. |
| *Enabled* | Установите флажок, чтобы включить действие. В противном случае оно будет отключено. |

**1. Условия**

Обзор

Можно задать, что действие будет выполнено, только если событие удовлетворяет определённому набору условий. Условия задаются при настройке действия.

Сопоставление условий чувствительно к регистру.

Действия на триггеры

В действиях на события триггеров можно использовать следующие условия:

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Группа узлов сети (Host group)* | равно<br>не равно | Укажите группу узлов сети или исключение группы узлов сети.<br>**равно** — событие относится к указанной группе узлов сети.<br>**не равно** — событие не относится к указанной группе узлов сети.<br>При указании родительской группы узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Чтобы выбрать только родительскую группу, все вложенные группы необходимо дополнительно перечислить с оператором **не равно**. |
| *Шаблон (Template)* | равно<br>не равно | Укажите шаблон или исключение шаблона.<br>**равно** — событие относится к триггеру, унаследованному из указанного шаблона.<br>**не равно** — событие не относится к триггеру, унаследованному из указанного шаблона. |
| *Узел сети (Tag name)* | равно<br>не равно | Укажите узел сети или исключение узла сети.<br>**равно** — событие относится к указанному узлу сети.<br>**не равно** — событие не относится к указанному узлу сети. |
| *Имя тега (Tag name)* | равно<br>не равно<br>содержит<br>не содержит | Укажите тег события или исключение тега события.<br>**равно** — событие имеет указанный тег.<br>**не равно** — событие не имеет указанный тег.<br>**содержит** — событие имеет тег, который содержит указанную строку.<br>**не содержит** — событие не имеет тега, который содержит указанную строку. |
| *Значение тега (Tag value)* | равно<br>не равно<br>содержит<br>не содержит | Укажите комбинацию тега события и его значения или исключение комбинации тега и значения.<br>**равно** — событие имеет указанные тег и значение.<br>**не равно** — событие не имеет указанные тег и значение.<br>**содержит** — событие содержит указанные строки тега и значения.<br>**не содержит** — событие не содержит указанные строки тега и значения. |
| *Триггер (Trigger)* | равно<br>не равно | Укажите триггер или исключение триггера.<br>**равно** — событие сгенерировано указанным триггером.<br>**не равно** — событие сгенерировано любым триггером, отличным от указанного. |

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Имя события (Event name)* | содержит<br>не содержит | Укажите строку в имени события, сгенерированного триггером, или исключение строки. По умолчанию имя события совпадает с именем триггера, если в настройках триггера не было указано настраиваемого имени события.<br>**содержит** — событие содержит в имени указанную строку.<br>**не содержит** — указанная строка не содержится в имени события.<br>Примечание: Заданное значение будет сравниваться с именем события со всеми раскрытыми макросами. |
| *Важность триггера (Trigger severity)* | равно<br>не равно<br>больше чем или равно<br>меньше чем или равно | Укажите важность триггера.<br>**равно** — совпадает с указанной важностью триггера.<br>**не равно** — не совпадает с указанной важностью триггера.<br>**больше чем или равно** — выше или равно указанной важности триггера.<br>**меньше чем или равно** — ниже или равно указанной важности триггера. |
| *Период времени (Time period)* | в<br>не в | Укажите период времени или исключение периода времени.<br>**в** — время события в течение указанного периода времени.<br>**не в** — время события не входит в указанный период времени.<br>Смотрите страницу Спецификации периодов времени для получения более подробных сведений об этом формате.<br>Поддерживаются пользовательские макросы. |
| *Проблема подавлена (Problem is suppressed)* | нет<br>да | Укажите, если проблема подавлена (не отображается) по причине обслуживания узла сети.<br>**нет** — проблема не подавлена.<br>**да** — проблема подавлена. |

Действия на услуги

В действиях на события услуг можно использовать следующие условия:

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Услуга (Service)* | равно<br>не равно | Укажите услугу или исключение услуги.<br>**равно** — событие относится к указанной услуге.<br>**не равно** — событие не относится к указанной услуге.<br>При указании родительской услуги косвенным образом будут выбраны все дочерние услуги. Чтобы выбрать только родительскую услугу, все вложенные услуги необходимо дополнительно перечислить с оператором **не равно**. |
| *Имя услуги (Service name)* | содержит<br>не содержит | Укажите строку из имени услуги или исключение строки.<br>**содержит** — событие произошло от услуги, которая содержит в имени указанную строку.<br>**не содержит** — указанная строка не содержится в имени услуги. |
| *Имя тега услуги (Service tag name)* | равно<br>не равно<br>содержит<br>не содержит | Укажите тег события или исключение тега события. Теги на события услуг можно добавить в настройках услуги в разделе *Теги*.<br>**равно** — событие имеет указанный тег.<br>**не равно** — событие не имеет указанный тег.<br>**содержит** — событие имеет тег, который содержит указанную строку.<br>**не содержит** — событие не имеет тега, который содержит указанную строку. |

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Значение тега услуги (Service tag value)* | равно<br>не равно<br>содержит<br>не содержит | Укажите комбинацию тега события и его значение или исключение комбинации тега и значения. Теги на события услуг можно добавить в настройках услуги в разделе *Теги*.<br>**равно** — событие имеет указанные тег и значение.<br>**не равно** — событие не имеет указанные тег и значение.<br>**содержит** — событие содержит указанные строки тега и значения.<br>**не содержит** — событие не содержит указанные строки тега и значения. |

**Attention:**
Обязательно задайте шаблоны сообщений для действий на услуги в меню *Оповещения → Способы оповещений*. В противном случае, оповещения не будут отправлены.

Действия на обнаружение

В действиях на события обнаружения можно использовать следующие условия:

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *IP узла сети (Host IP)* | равно<br>не равно | Укажите диапазон IP адресов или исключение диапазона у обнаруженного узла сети.<br>**равно** — IP адрес узла сети в указанном диапазоне.<br>**не равно** — IP адрес узла сети не из указанного диапазона.<br>Доступны следующие форматы:<br>Одиночный IP: 192.168.1.33<br>Диапазон IP адресов: 192.168.1-10.1-254<br>маска IP: 192.168.4.0/24<br>Список диапазонов: 192.168.1.1-254, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24<br>В формате списка поддерживаются пробелы. |
| *Тип сервиса (Service type)* | равно<br>не равно | Укажите тип сервиса обнаруженного сервиса или исключение типа сервиса.<br>**равно** — совпадает с обнаруженным сервисом.<br>**не равно** — не совпадает с обнаруженным сервисом.<br>Доступные типы сервисов: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Zabbix агент, SNMPv1 агент, SNMPv2 агент, SNMPv3 агент, ICMP пинг, telnet. |
| *Порт сервиса (Service port)* | равно<br>не равно | Укажите диапазон TCP портов или исключение диапазона у обнаруженного узла сети.<br>**равно** — порт сервиса из указанного диапазона.<br>**не равно** — порт сервиса не из указанного диапазона. |
| *Правило обнаружения (Discovery rule)* | равно<br>не равно | Укажите правило обнаружения или исключение правила обнаружения.<br>**равно** — использование указанного правила обнаружения.<br>**не равно** — использование любого другого правила обнаружения, отличного от указанного. |
| *Проверка обнаружения (Discovery check)* | равно<br>не равно | Укажите проверку обнаружения или исключение проверки обнаружения.<br>**равно** — использование указанной проверки обнаружения.<br>**не равно** — использование любой другой проверки обнаружения, отличной от указанной. |
| *Обнаруженный объект (Discovery object)* | равно | Укажите обнаруженный объект.<br>**равно** — совпадает с обнаруженным объектом (устройством или сервисом). |

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Состояние обнаружения (Discovery status)* | равно | **Доступен (Up)** — совпадает с событиями «Узел сети доступен» и «Сервис доступен».<br>**Недоступен (Down)** — совпадает с событиями «Узел сети недоступен» и «Сервис недоступен».<br>**Обнаружен (Discovered)** — совпадает с событиями «Узел сети обнаружен» и «Сервис обнаружен».<br>**Потерян (Lost)** — совпадает с событиями «Узел сети потерян» и «Сервис потерян». |
| *Доступен/Недоступен (Uptime/Downtime)* | больше чем или равно<br>меньше чем или равно | Время доступности для событий «Узел сети доступен» и «Сервис доступен». Время недоступности для событий «Узел сети недоступен» и «Сервис недоступен».<br>**больше чем или равно** — больше или совпадает. Параметр задается в секундах.<br>**меньше чем или равно** — меньше или совпадает. Параметр задается в секундах. |
| *Полученное значение (Received value)* | равно<br>не равно<br>больше чем или равно<br>меньше чем или равно<br>содержит<br>не содержит | Укажите полученное значение от проверки агента (Zabbix, SNMP) в правиле обнаружения. Регистрозависимое сравнение строк. Если в правиле заданы несколько проверок Zabbix агента или SNMP, проверяется каждое полученное от них значения (каждая проверка генерирует новое событие, которое сопоставляется со всеми условиями).<br>**равно** — совпадает с указанным значением.<br>**не равно** — не совпадает с указанным значением.<br>**больше чем или равно** — больше или совпадает с указанным значением.<br>**меньше чем или равно** — меньше или совпадает с указанным значением.<br>**содержит** — содержит указанную подстроку. Параметр задается в виде строки.<br>**не содержит** — не содержит указанную подстроку. Параметр задается строкой. |
| *Прокси (Proxy)* | равно<br>не равно | Укажите прокси или исключение прокси.<br>**равно** — используется указанный прокси.<br>**не равно** — используется любой другой прокси отличный от указанного. |

**Note:**
Проверки сервисов в правиле обнаружения, которые приводят к событиям обнаружения, не выполняются одновременно. Поэтому, если в действии настроено **несколько** значений для условий , или , они будут сравниваться с одним событием обнаружения за раз, но **не** будут сравниваться с несколькими событиями одновременно. В результате действия с несколькими значениями по одним и тем же типам проверок могут быть выполнены некорректно.

Действия на авторегистрацию

В действиях на события авторегистрации активного агента можно использовать следующие условия:

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Метаданные узлов сети (Host metadata)* | содержит<br>не содержит<br>совпадает<br>не соответствует | Укажите метаданные узла сети или метаданные узла сети для исключения.<br>**содержит** — метаданные узла сети содержат указанную строку.<br>**не содержит** — метаданные узла сети не содержат указанную строку. Метаданные узла сети можно задать в файле конфигурации агента.<br>**совпадает** — метаданные узла сети соответствуют регулярному выражению.<br>**не соответствует** — метаданные узла сети не соответствуют регулярному выражению. |

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Имя узла сети (Host name)* | содержит<br>не содержит<br>совпадает<br>не соответствует | Укажите имя узла сети или имя узла сети для исключения.<br>**содержит** — имя узла сети содержит указанную строку.<br>**не содержит** — имя узла сети не содержит указанную строку.<br>**совпадает** — имя узла сети соответствует регулярному выражению.<br>**не соответствует** — имя узла сети не соответствует регулярному выражению. |
| *Прокси (Proxy)* | равно<br>не равно | Укажите прокси или прокси для исключения:<br>**равно** — используется указанный прокси.<br>**не равно** — используется любой другой прокси, отличный от указанного. |

Действия на внутренние события

В действиях на внутренние события можно использовать следующие условия:

| Тип условия | Поддерживаемые операторы | Описание |
|---|---|---|
| *Тип события (Event type)* | равно | **Элемент данных в состоянии "не поддерживается" (Item in "not supported" state)** — соответствует событию, когда элемент данных переходит из состояния «активировано» в состояние «не поддерживается».<br>**Правило низкоуровневого обнаружения в состоянии "не поддерживается" (Low-level discovery rule in "not supported" state)** — соответствует событию, когда правило низкоуровневого обнаружения переходит из состояния «активировано» в состояние «не поддерживается».<br>**Триггер в состоянии "неизвестно" (Trigger in "unknown" state)** — соответствует событию, когда триггер переходит из состояния «активировано» в состояние «неизвестно». |
| *Группа узлов сети (Host group)* | равно<br>не равно | Укажите группу узлов сети или группу узлов сети для исключения.<br>**равно** — событие относится к указанной группе узлов сети.<br>**не равно** — событие не относится к указанной группе узлов сети. |
| *Имя тега (Tag name)* | равно<br>не равно<br>содержит<br>не содержит | Укажите тег события или исключение тега события.<br>**равно** — событие имеет указанный тег.<br>**не равно** — событие не имеет указанный тег.<br>**содержит** — событие имеет тег, который содержит указанную строку.<br>**не содержит** — событие не имеет тега, который содержит указанную строку. |
| *Значение тега (Tag value)* | равно<br>не равно<br>содержит<br>не содержит | Укажите комбинацию тега события и его значения или исключение комбинации тега и значения.<br>**равно** — событие имеет указанные тег и значение.<br>**не равно** — событие не имеет указанные тег и значение.<br>**содержит** — событие содержит указанные строки тега и значения.<br>**не содержит** — событие не содержит указанные строки тега и значения. |
| *Шаблон (Template)* | равно<br>не равно | Укажите шаблон или шаблон для исключения.<br>**равно** — событие относится к элементу данных / триггеру / правилу низкоуровневого обнаружения, унаследованному из указанного шаблона.<br>**не равно** — событие не относится к элементу данных / триггеру / правилу низкоуровневого обнаружения, унаследованному из указанного шаблона. |
| *Узел сети (Host)* | равно<br>не равно | Укажите узел сети или узел сети для исключения.<br>**равно** — событие относится к указанному узлу сети.<br>**не равно** — событие не относится к указанному узлу сети. |

Тип вычисления

Для вычисления условий доступны следующие опции:

- **И (And)** — должны быть выполнены все условия

Обратите внимание, вычисление «И» запрещено использовать между несколькими триггерами, когда они добавлены с условием . Действия могут выполняться на основе события только одного триггера.

- **Или (Or)** — достаточно выполнения одного условия
- **И/Или (And/Or)** — комбинация из двух опций: И с различными типами условий и ИЛИ с одинаковым типом условий, например:

*Группа узлов сети* равно Oracle сервера<br> *Группа узлов сети* равно MySQL сервера<br> *Имя события* содержит «База данных не работает»<br> *Имя события* содержит «База данных недоступна»

вычисляется как

**(**Группа узлов сети равно Oracle сервера **или** Группа узлов сети равно MySQL сервера**) и (**Имя события содержит «База данных не работает» **или** Имя события содержит «База данных недоступна»**)**

- **Пользовательское выражение** — формула вычисления, введенная пользователем, для оценки условий действия. Она должна включать в себя все условия (представленные в виде заглавных букв A, B, C, ...) и может включать пробелы, символы табуляции, скобки ( ), **and** (с учётом регистра), **or** (с учётом регистра), **not** (с учётом регистра).

Тогда как предыдущий пример с / был бы представлен в виде `(A or B) and (C or D)`, в пользовательском выражении вы также можете использовать несколько других методов вычисления:

(A and B) and (C or D)<br> (A and B) or (C and D)<br> ((A or B) and C) or D<br> (not (A or B) and C) or not D<br> и так далее.

Удаление объектов, используемых в действиях

При попытке удалить объекты, которые используются в условии/операции действия, пользователь получит соответствующее сообщение об ошибке.

Действия не отключаются, если условия или операции действия:

- содержат удаленные объекты;
- содержат объекты, которые больше не обнаруживаются низкоуровневым обнаружением и были удалены (по истечении периода *Удаление потерянных ресурсов*).

Условия/операции действия не удаляются, если в них отсутствуют объекты. Вместо этого удаленные объекты отображаются как *Удаленная группа узлов сети*, *Удаленный узел сети*, *Удаленный триггер* и т. д. В столбце ”Info” списка действий отображается значок предупреждения с сообщением ”This action has conditions or operations referencing deleted object(s).”

**2. Операции**

Обзор

Вы можете указать следующие операции по всем событиям:

- Отправка сообщений
- Выполнение удалённых команд

> **Attention:**
> Zabbix сервер не создаёт оповещения, если доступ к узлу сети явным образом «запрещён» пользователю, который задан в операции в качестве получателя, или если пользователь вовсе не имеет прав доступа, заданных к этому узлу сети.

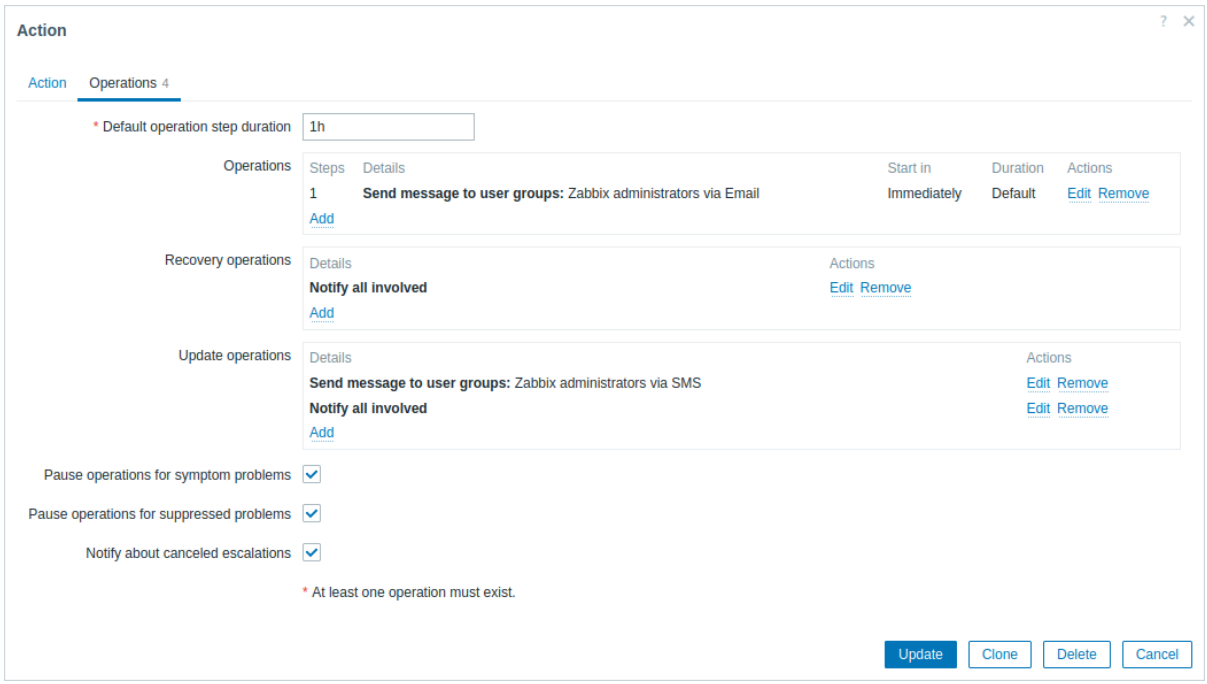
Для событий обнаружения и авторегистрации доступны дополнительные операции:

- Добавление узла сети
- Удаление узла сети
- Активация узла сети
- Деактивация узла сети
- Добавление в группу узлов сети
- Удаление из группы узлов сети
- Добавление тегов уровня узла сети
- Удаление тегов уровня узла сети
- Присоединение шаблона
- Отсоединение шаблона

- Установка режима инвентарных данных узлам сети

Настройка операции

Чтобы настроить операцию, перейдите на вкладку *Operations* в конфигурации action.




Общие атрибуты операции:

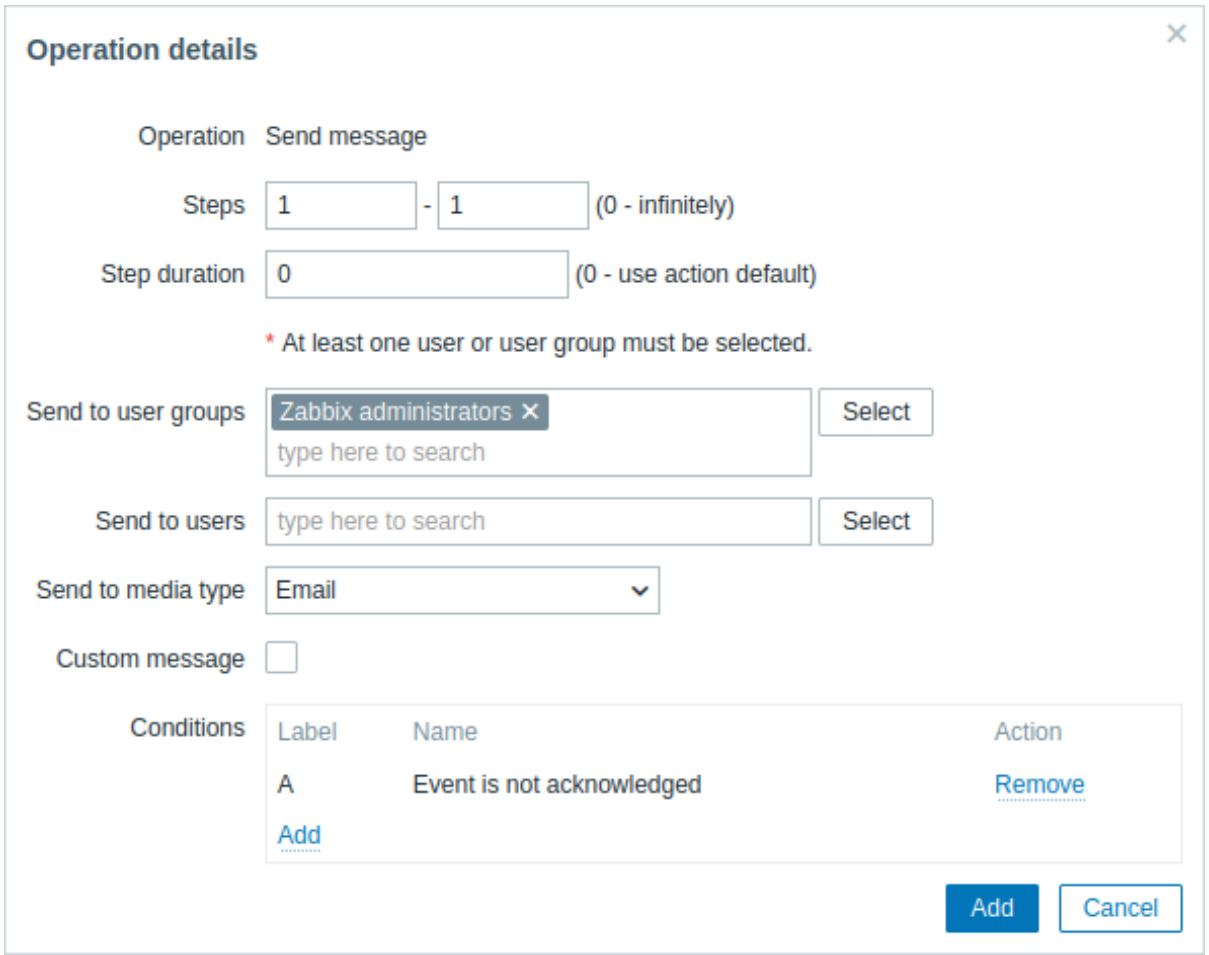
| Parameter | Description |
|---|---|
| *Default operation step duration* | Длительность одного шага операции по умолчанию (от 60 секунд до 1 недели).<br>Например, если длительность шага составляет один час, то при выполнении операции до следующего шага пройдет один час.<br>Поддерживаются суффиксы времени, например 60s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы. |
| *Operations* | Отображаются операции действия, если они есть, со следующими сведениями:<br>**Steps** - шаг(и) эскалации, к которым назначена операция.<br>**Details** - тип операции и ее получатель/цель.<br>В списке операций также отображаются тип медиа (email, SMS или script), а также имя и фамилия (в скобках после имени пользователя) получателя уведомления.<br>**Start in** - через какое время после события выполняется операция.<br>**Duration (sec)** - отображается длительность шага. Если используется длительность по умолчанию, отображается *Default*; если используется пользовательская длительность, отображается время.<br>**Actions** - отображаются ссылки для редактирования и удаления операции. |
| *Recovery operations* | Отображаются операции действия, если они есть, со следующими сведениями:<br>**Details** - тип операции и ее получатель/цель.<br>В списке операций также отображаются тип медиа (email, SMS или script), а также имя и фамилия (в скобках после имени пользователя) получателя уведомления.<br>**Actions** - отображаются ссылки для редактирования и удаления операции. |
| *Update operations* | Отображаются операции действия, если они есть, со следующими сведениями:<br>**Details** - тип операции и ее получатель/цель.<br>В списке операций также отображаются тип медиа (email, SMS или script), а также имя и фамилия (в скобках после имени пользователя) получателя уведомления.<br>**Actions** - отображаются ссылки для редактирования и удаления операции. |
| *Pause operations for symptom problems* | Установите этот флажок, чтобы приостановить операции (после первой операции) для проблем-симптомов.<br>Обратите внимание, что эта настройка влияет только на эскалации проблем; операции восстановления и обновления затронуты не будут.<br>Этот параметр доступен только для *Trigger actions*. |

| Parameter | Description |
|---|---|
| *Pause operations for suppressed problems* | Установите этот флажок, чтобы отложить начало операций на время периода обслуживания.<br>Когда операции будут запущены после обслуживания, будут выполнены все операции, включая те, что относятся к событиям, произошедшим во время обслуживания.<br>Обратите внимание, что эта настройка влияет только на эскалации проблем; операции восстановления и обновления затронуты не будут.<br>Если снять этот флажок, операции будут выполняться без задержки даже во время периода обслуживания.<br>Этот параметр недоступен для *Service actions*. |
| *Notify about canceled escalations* | Снимите этот флажок, чтобы отключить уведомления об отмененных эскалациях (когда узел сети, элемент данных, триггер или действие отключены). |

Все обязательные поля ввода отмечены красной звездочкой.

Чтобы настроить сведения о новой операции, нажмите Add в блоке *Operations*. Чтобы изменить существующую операцию, нажмите Edit рядом с операцией. Откроется всплывающее окно, в котором можно изменить сведения о шаге операции.

Детали операции




| Parameter | Description |
|---|---|
| *Operation* | Выберите операцию:<br>**Send message** - отправить сообщение пользователю.<br>**<remote command name>** - выполнить удаленную команду. Команды доступны для выполнения, если они ранее определены в global scripts и для них в качестве области действия выбрано *Action operation*.<br>Дополнительные операции доступны для событий обнаружения и автосоздания на основе событий (см. выше). |

| Parameter | Description |
|---|---|
| *Steps* | Выберите шаг(и), к которым нужно назначить операцию, в расписании эскалации:<br>**From** - выполнять, начиная с этого шага.<br>**To** - выполнять до этого шага (0=бесконечность, выполнение не будет ограничено). |
| *Step duration* | Пользовательская длительность для этих шагов (0=использовать длительность шага по умолчанию).<br>Поддерживаются суффиксы времени, например 60s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы.<br>Несколько операций можно назначить одному и тому же шагу. Если для этих операций задана разная длительность шага, учитывается и применяется к шагу наименьшая из них. |
| Operation type: send message | |
| *Send to user groups* | Выберите группы пользователей, которым нужно отправить сообщение.<br>Группа пользователей должна иметь как минимум права ”read” на узел сети, чтобы получать уведомления. |
| *Send to users* | Выберите пользователей, которым нужно отправить сообщение.<br>Пользователь должен иметь как минимум права ”read” на узел сети, чтобы получать уведомления. |
| *Send to media type* | Отправить сообщение всем доступным (настроенным и включенным) типам медиа или только определенному. |
| *Custom message* | Если выбрано, можно настроить пользовательское сообщение.<br>Для уведомлений о внутренних событиях через webhooks пользовательское сообщение обязательно. |
| *Subject* | Тема пользовательского сообщения. Тема может содержать макросы.<br>Ограничение - 255 символов. |
| *Message* | Пользовательское сообщение. Сообщение может содержать макросы.<br>Ограничение по количеству символов зависит от типа базы данных (подробнее см. в разделе Sending message). |
| Operation type: remote command | |
| *Target list* | Выберите цели, на которых нужно выполнить команду:<br>**Current host** - команда выполняется на узле сети триггера, вызвавшего событие проблемы. Этот вариант не будет работать, если в триггере несколько узлов сети.<br>**Host** - выберите узел(ы) сети, на которых нужно выполнить команду.<br>**Host group** - выберите группу(ы) узлов сети, на которых нужно выполнить команду. Указание родительской группы узлов сети неявно выбирает все вложенные группы. Таким образом, удаленная команда также будет выполнена на узлах сети из вложенных групп.<br>Команда на узле сети выполняется только один раз, даже если узел сети соответствует нескольким условиям (например, входит в несколько групп узлов сети; выбран отдельно и через группу узлов сети).<br>Список целей не имеет смысла, если пользовательский скрипт выполняется на сервере Zabbix. В этом случае выбор большего числа целей лишь приводит к тому, что скрипт выполняется на сервере большее число раз.<br>Обратите внимание, что для глобальных скриптов выбор цели также зависит от настройки *Host group* в конфигурации глобального скрипта.<br>Параметр *Target list* недоступен для *Service actions*, поскольку в этом случае удаленные команды всегда выполняются на сервере Zabbix. |

| Parameter | Description |
|---|---|
| *Conditions* | Условие для выполнения операции:<br>**Event is not acknowledged** - только если событие не подтверждено.<br>**Event is acknowledged** - только если событие подтверждено.<br>Параметр *Conditions* доступен только для *Trigger actions*. |

Когда закончите, нажмите *Add*, чтобы добавить операцию в список *Operations*.

1. Отправка сообщений

Обзор

Отправка сообщений является одним из наилучших способов оповестить людей о наличии проблем. Вот по какой причине это одно из основных действий, предлагаемых Zabbix.

Настройка

Чтобы иметь возможность отправлять и получать оповещения от Zabbix, вам потребуется:

- задать способы оповещений для отправки сообщения

Если операция происходит вне пределов периода Когда активен, определённого для выбранного способа оповещений в настройках пользователя, то сообщение не будет отправлено.

Важность триггера по умолчанию («Не классифицировано») **должна быть** выбрана в настройках оповещения пользователя, если вы хотите получать сообщения о событиях, не связанных с триггерами, такими как обнаружение, автоматическая регистрация активных агентов или внутренние события.

- настроить операцию действия, которая отправит сообщения по одному из заданных способов оповещений

**Attention:**
Zabbix отправит оповещения только тем пользователям, которые имеют по крайней мере права на «чтение» узла сети, который сформировал это событие. По крайней мере один узел сети из выражения триггера должен быть доступен.

Вы можете настроить пользовательские сценарии для отправки сообщений с использованием эскалаций.

Для успешного получения и чтения email от Zabbix, email сервера/клиенты должны поддерживать стандарт «SMTP/MIME e-mail», поскольку Zabbix отправляет данные в кодировке UTF-8 (если тема содержит только ASCII символы, она не будет перекодирована в UTF-8). Тема и тело сообщения кодируются в base64, чтобы они соответствовали стандарту формата «SMTP/MIME email».

Ограничение длины сообщения после раскрытия макросов такое же, как и ограничение длины сообщения для Удалённых команд.

Отслеживание сообщений

Вы можете просматривать статус сообщений, отправленных в *Monitoring* → *Problems*.

В столбце *Actions* вы можете видеть сводную информацию о выполненных действиях. Зеленые числа обозначают отправленные сообщения, красные — неудачные сообщения. *In progress* означает, что действие инициировано. *Failed* сообщает, что ни одно действие не было выполнено успешно.

Если вы щелкнете по времени события, чтобы просмотреть его подробности, вы сможете увидеть сведения о сообщениях, отправленных (или не отправленных) из-за этого события, в блоке *Actions*.

В *Reports* → *Action log* вы увидите подробности всех выполненных действий для тех событий, для которых настроено действие.

2. Удалённые команды

Обзор

При помощи удалённых команд вы можете указать, что при некоторых условиях на наблюдаемом узле сети будет автоматически выполнена определённая предварительно настроенная команда.

Поэтому удалённые команды являются мощным механизмом для умного проактивного мониторинга.

Из наиболее очевидных применений этой возможности, которые вы можете попробовать:

- Автоматически перезапустить какое-либо приложение (веб-сервер, промежуточное ПО, CRM), если оно не отвечает
- Использовать команду IPMI «reboot» для перезагрузки некоторого удалённого сервера, если он не отвечает на запросы
- Автоматически почистить диск (удаляя старые файлы, очищая /tmp), если заканчивается место на диске
- Мигрировать виртуальную машину с одного физического сервера на другой, в зависимости от загрузки CPU
- Добавить новые узлы в облачную среду при недостаточных ресурсах CPU (диска, памяти, чего-либо)

Настройка действий для удалённых команд аналогична настройке действий для отправки сообщений, с той лишь разницей, что вместо отправки сообщения Zabbix будет выполнять команду.

Удалённые команды могут выполняться на стороне Zabbix сервера, прокси или агента. Удалённые команды на Zabbix агенте могут выполняться напрямую Zabbix сервером или через Zabbix прокси. По умолчанию удалённые команды отключены как у Zabbix агента, так и у Zabbix прокси. Их можно включить:

- добавив параметр `AllowKey=system.run[*]` в конфигурацию агента;
- изменив параметр EnableRemoteCommands на значение «1» в конфигурации прокси.

Удалённые команды, выполняемые Zabbix сервером, запускаются, как описано в разделе Выполнение команд, включая проверку кода завершения.

Удалённые команды будут выполнены, даже если целевой узел сети находится в обслуживании.

Ограничение удаленной команды

Ограничение удаленной команды после разрешения всех макросов зависит от типа базы данных и набора символов (не-ASCII символы требуют более одного байта для хранения):

| Database | Limit in characters | Limit in bytes |
|---|---|---|
| MySQL | 65535 | 65535 |
| PostgreSQL | 65535 | not limited |
| SQLite (only Zabbix proxy) | 65535 | not limited |

Вывод выполнения удаленной команды (возвращаемое значение) ограничен 16 МБ (включая завершающие пробельные символы, которые усекаются). Ограничение для удаленной команды IPMI основано на установленной библиотеке IPMI. Обратите внимание, что ограничения базы данных применяются ко всем удаленным командам.

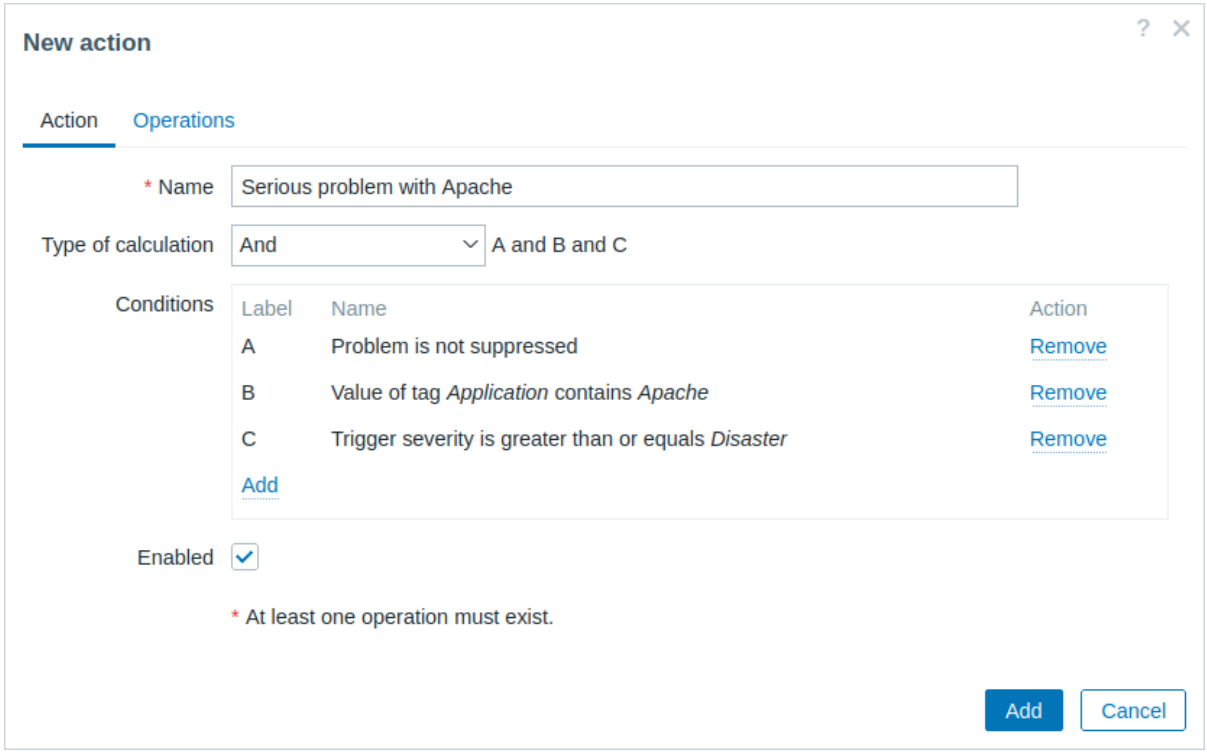
Конфигурация

Те удаленные команды, которые выполняются на агенте Zabbix (пользовательские скрипты), сначала должны быть включены в конфигурации агента.

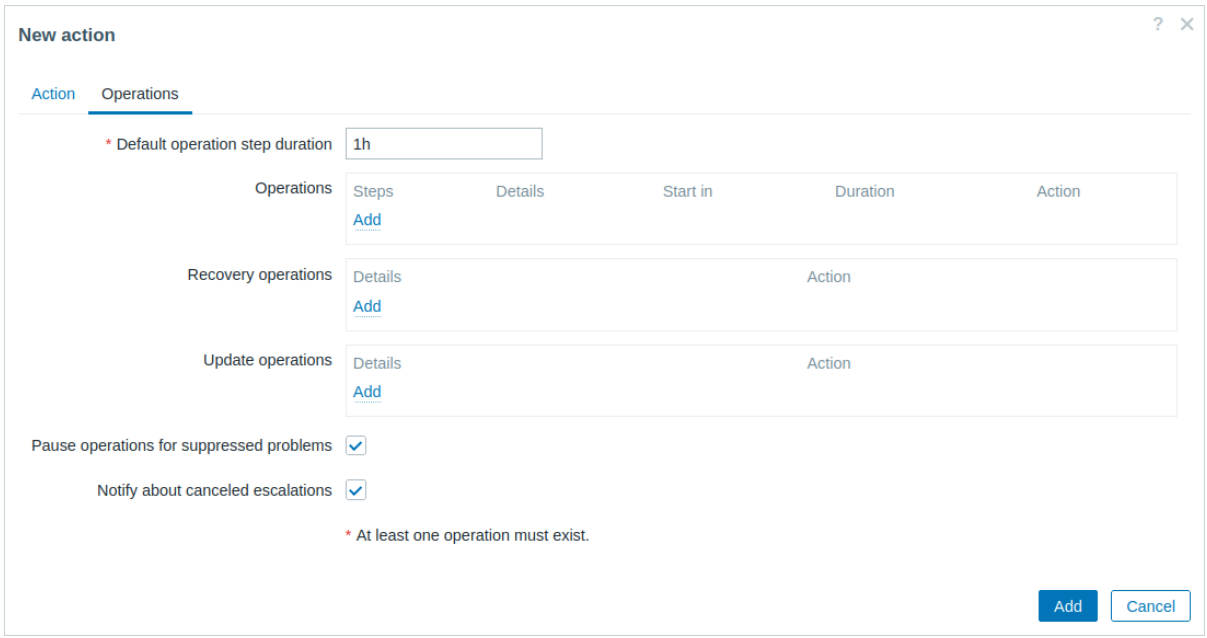
Убедитесь, что параметр AllowKey=system.run[<command>,*] добавлен для каждой разрешенной команды в конфигурации агента, чтобы разрешить выполнение конкретной команды в режиме nowait. Перезапустите демон агента, если изменяете этот параметр.

Затем при настройке нового действия в *Уведомления* → *Действия* → *Действия триггеров*:

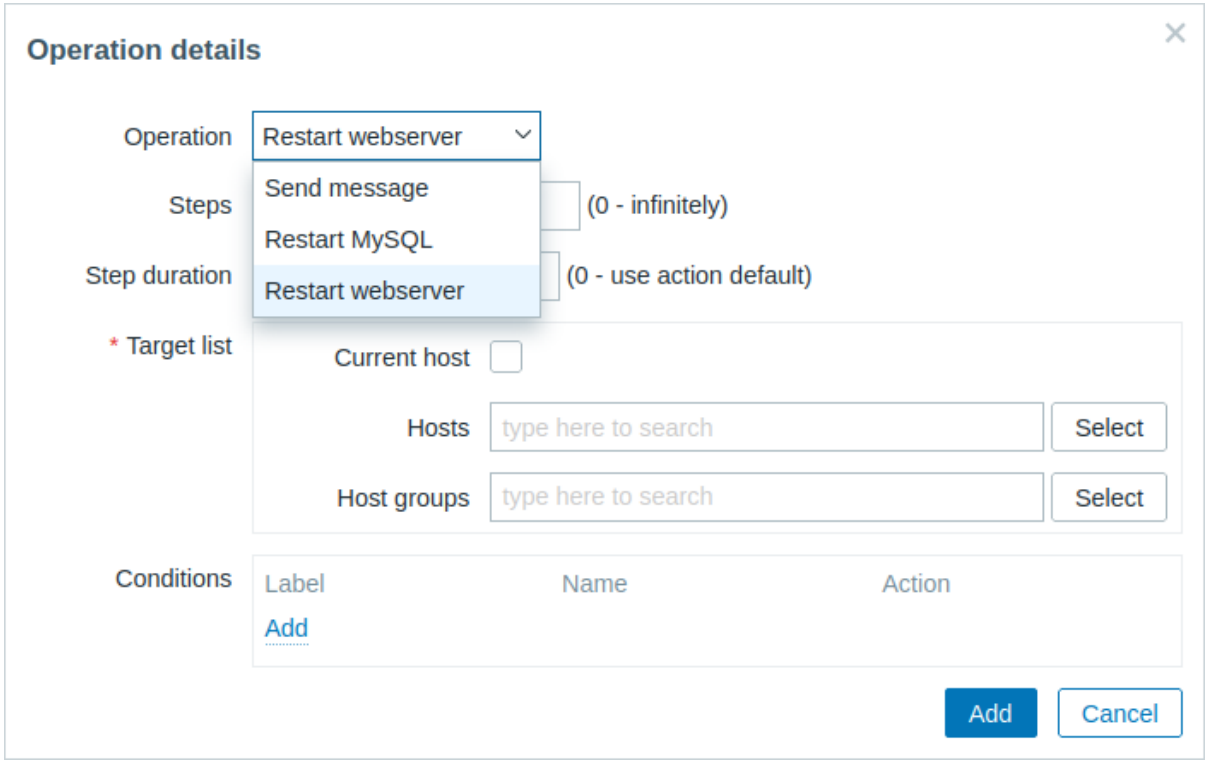
1. Определите соответствующие условия, например, задайте, чтобы действие активировалось при любых проблемах уровня disaster с одним из приложений Apache.




2. На вкладке *Операции* нажмите *Добавить* в блоке *Операции*, *Операции восстановления* или *Операции обновления*.




3. Выберите один из предопределенных скриптов в раскрывающемся списке *Операция* и задайте *Список целей* для скрипта.




Предварительно настроенные скрипты

Скрипты, которые доступны для операций действий (вебхук, скрипт, SSH, Telnet, IPMI), задаются в глобальных скриптах.

Например:

```
sudo /etc/init.d/apache restart
```

В даном случае Zabbix попытается перезапустить процесс Apache. При использовании этой команды убедитесь, что она будет выполнена Zabbix агентом (нажмите на *Zabbix агент* напротив *Выполнять на (Execute on)*).

**Attention:**
Обратите внимание на использование **sudo** — по умолчанию пользователь Zabbix не имеет прав на перезапуск системных служб. Смотрите ниже подсказки о том, как настроить **sudo**.

**Note:**
Начиная с Zabbix агента 7.0, удалённые команды также могут быть выполнены на агенте, который работает в активном режиме. Zabbix агент — активный или пассивный — должен работать на удалённом узле сети; он выполняет команды в фоновом режиме.

Удалённые команды на стороне Zabbix агента выполняются без учёта времени ожидания при помощи ключа system.run[,nowait], таким образом, результаты выполнения не проверяются. На стороне Zabbix сервера и Zabbix прокси команды выполняются с учётом времени ожидания, указанного в параметре TrapperTimeout в файле zabbix_server.conf или в zabbix_proxy.conf, и результаты выполнения проверяются. Для получения дополнительной информации смотрите *Тайм-аут скрипта*.

Права доступа

Убедитесь, что пользователь «zabbix» имеет права на выполнение сконфигурированных команд. Может оказаться интересным вариант с использованием **sudo** для предоставления доступа к привилегированным командам. Для настройки доступа выполните из-под root:

```
visudo
```

Примеры строк, которые можно использовать в файле *sudoers*:

```
#              'zabbix'                    .
zabbix ALL=NOPASSWD: ALL

#              'zabbix'        apache      .
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/apache restart
```

**Note:**
В некоторых системах настройки файла *sudoers* будут препятствовать не-локальным пользователям выполнять команды. Чтобы это изменить, закомментируйте опцию **requiretty** в */etc/sudoers*.

Удалённые команды при наличии нескольких интерфейсов

Если целевая система имеет несколько интерфейсов выбранного типа (Zabbix агент или IPMI), удалённые команды будут выполнены с использованием интерфейса по умолчанию.

Имеется возможность выполнения удалённых команд через SSH и Telnet с использованием другого интерфейса, отличного от интерфейса Zabbix агента. Доступный интерфейс, который будет использоваться, выбирается в следующем порядке:

- Zabbix агент интерфейс по умолчанию
- SNMP интерфейс по умолчанию
- JMX интерфейс по умолчанию
- IPMI интерфейс по умолчанию

Удалённые команды через IPMI

В случае использования удалённых команд через IPMI нужно использовать следующий синтаксис:

<  > [<  >]

где

- <команда> — одна из IPMI команд без пробелов
- <значение> — «on», «off» или любое неотрицательное целое число. <значение> — опциональный параметр.

Примеры

Примеры глобальных скриптов, которые можно использовать как удалённые команды в операциях к действиям.

**Пример 1**

Перезагрузка Windows при определённом условии.

Для того, чтобы автоматически перезагрузить Windows, когда Zabbix обнаруживает проблемы, задайте следующий скрипт:

| Параметр скрипта | Значение |
|---|---|
| *Область* | «Операция действия» |
| *Тип* | «Скрипт» |
| *Команда* | c:\windows\system32\shutdown.exe -r -f |

**Пример 2**

Перезагрузка узла сети с использованием управления по IPMI.

| Параметр скрипта | Значение |
|---|---|
| *Область* | «Операция действия» |
| *Тип* | «IPMI» |
| *Команда* | reset |

**Пример 3**

Выключение узла сети с использованием управления по IPMI.

| Параметр скрипта | Значение |
|---|---|
| *Область* | «Операция действия» |
| *Тип* | «IPMI» |
| *Команда* | power off |

3. Дополнительные операции

Обзор

В этом разделе вы можете найти некоторые сведения о дополнительных операциях на события обнаружения / автоматической регистрации.

Добавление узла сети

Узлы сети добавляются в процессе обнаружения, сразу после того, как узел сети обнаружен, а не в конце процесса обнаружения.

> **Note:**
> Поскольку обнаружение сети может занять некоторое время из-за большого количества недоступных узлов сети/служб, рекомендуется проявлять терпение и использовать разумные диапазоны IP-адресов.

При добавлении узла сети его имя определяется стандартной функцией **gethostbyname**. Если узел сети удается разрешить, используется разрешенное имя. Если нет, используется IP-адрес. Кроме того, если для имени узла сети необходимо использовать IPv6-адрес, то все символы ":" (двоеточия) заменяются на "_" (подчеркивания), поскольку двоеточия не допускаются в именах узлов сети.

> **Attention:**
> Если обнаружение выполняется через прокси, в настоящее время поиск имени хоста по-прежнему выполняется на сервере Zabbix.

> **Attention:**
> Если узел сети уже существует в конфигурации Zabbix с тем же именем, что и только что обнаруженный, Zabbix добавит **_N** к имени узла сети, где **N** — возрастающее число, начиная с 2.

4. Использование макросов в сообщениях

Обзор

Для более эффективного оповещения о проблемах можно в теме и в тексте сообщений использовать макросы.

В дополнение к встроенным макросам, также поддерживаются пользовательские макросы и макросы выражений. Доступен полный список макросов, поддерживаемых Zabbix.

Примеры

Приведённые здесь примеры иллюстрируют то, как вы можете использовать макросы в сообщениях.

Пример 1

Тема сообщения:

```
: {TRIGGER.NAME}
```

При получении сообщения тема этого сообщения будет заменена на нечто подобное:

```
:                          Zabbix
```

Пример 2

Сообщение:

```
          : {?last(/zabbix.zabbix.com/system.cpu.load[,avg1])}
```

При получении сообщения само сообщение будет заменено на нечто подобное:

```
          : 1.45
```

Пример 3

Сообщение:

```
         : {?last(/{HOST.HOST}/{ITEM.KEY})}
 15       : {?max(/{HOST.HOST}/{ITEM.KEY},15m)}
15       : {?min(/{HOST.HOST}/{ITEM.KEY},15m)}
```

При получении сообщения само сообщение будет заменено на нечто подобное:

```
          : 1.45
     15    : 2.33
     15    : 1.01
```

Пример 4

Сообщение:

```
http://<server_ip_or_name>/zabbix/tr_events.php?triggerid={TRIGGER.ID}&eventid={EVENT.ID}
```

Когда вы получите сообщение, оно будет содержать ссылку на страницу *Детали события*, которая предоставляет информацию о событии, его триггере и список последних событий, сгенерированных тем же триггером.

Пример 5

Информирование о значениях с нескольких узлов сети из выражения триггера.

Сообщение:

```
       : {TRIGGER.NAME}
          : {TRIGGER.EXPRESSION}

1.                  {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1})
2.                  {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})
```

При получении сообщения само сообщение будет заменено на нечто подобное:

```
       :
          : last(/Myhost/system.cpu.load[percpu,avg1])>5 or last(/Myotherhost/system.cpu.load[percpu,avg1]

1.                  Myhost: 0.83 (            (         1         ))
2.                  Myotherhost: 5.125 (            (         1         ))
```

Пример 6

Получение деталей по событию о проблеме и событию о восстановлении в сообщении о восстановлении:

Сообщение:

```
     :

ID     : {EVENT.ID}
          : {EVENT.VALUE}
           : {EVENT.STATUS}
         : {EVENT.TIME}
        : {EVENT.DATE}
          : {EVENT.AGE}
             : {EVENT.ACK.STATUS}
                : {EVENT.UPDATE.HISTORY}

         :

ID     : {EVENT.RECOVERY.ID}
          : {EVENT.RECOVERY.VALUE}
           : {EVENT.RECOVERY.STATUS}
         : {EVENT.RECOVERY.TIME}
        : {EVENT.RECOVERY.DATE}
           : {EVENT.OPDATA}
```

При получении сообщения эти макросы будет заменены на нечто подобное:

```
     :

ID     : 21874
          : 1
           : PROBLEM
         : 13:04:30
        : 2018.01.02
          : 5m 0s
             : Yes
                : 2018.01.02 13:05:51 "John Smith (Admin)"
```

```
    :        .

        :

ID     : 21896
         : 0
          : OK
        : 13:10:07
       : 2014.01.02
          :          : 0.83
```

**3. Операции восстановления**

Обзор

Операции восстановления позволяют вам получить оповещения при решении проблем.

В операциях восстановления поддерживаются как сообщения, так и удалённые команды. Хоть можно добавить несколько операций, эскалации не поддерживаются — все операции назначаются на единственный шаг, поэтому будут выполняться одновременно.

Сценарии применения

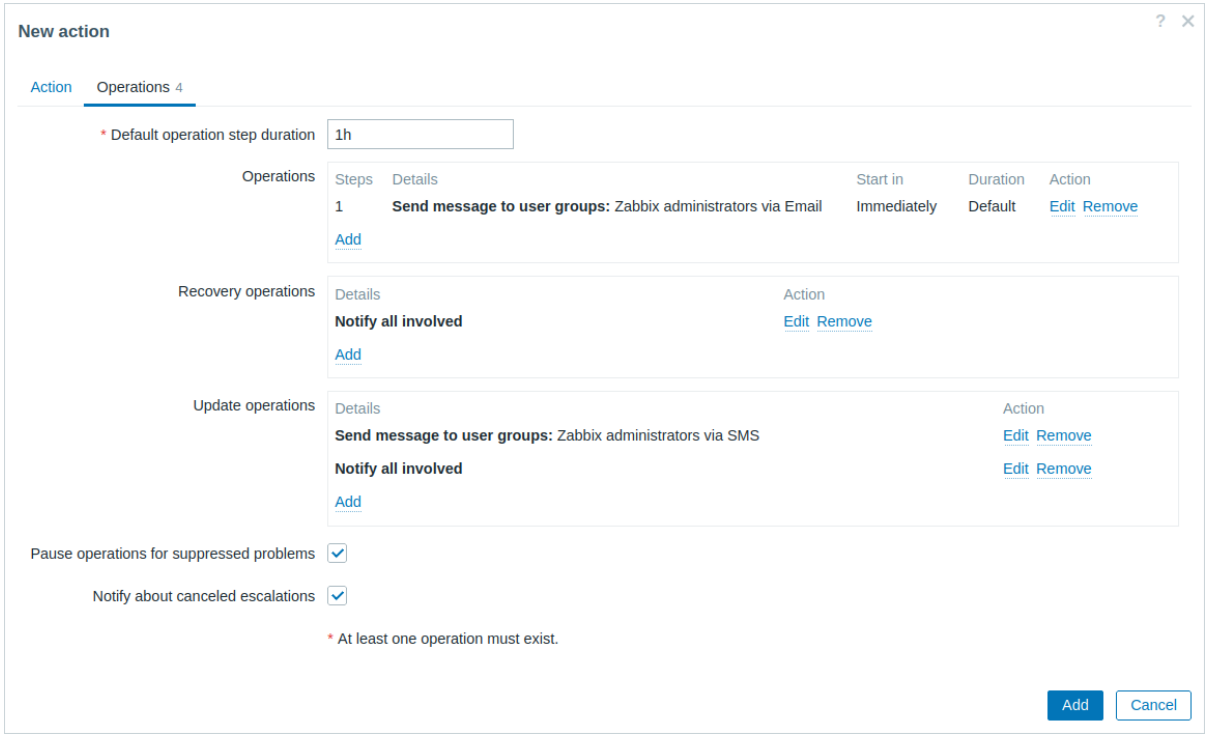
Некоторые сценарии использования операций восстановления заключаются в следующем:

1. Оповещение о восстановлении всех пользователей, которые были ранее оповещены о проблеме:
   - Выберите в качестве типа операции *Оповещать всех участников (Notify all involved)*
2. Наличие нескольких операций по восстановлению: отправка оповещений и выполнение удалённой команды:
   - Добавьте типы операций для отправки сообщения и выполнения команды
3. Открытие задачи во внешней системе поддержки клиентов / системе задач и закрытие этих задач при решении проблемы:
   - Создайте внешний скрипт, который будет взаимодействовать с системой поддержки клиентов.
   - Создайте действие, имеющее операцию, которая выполняет этот скрипт и таким образом открывает задачу.
   - Добавьте операцию восстановления, которая выполнит этот скрипт с другими параметрами и закроет задачу.
   - Используйте макрос {EVENT.ID} в качестве ссылки на исходную проблему.

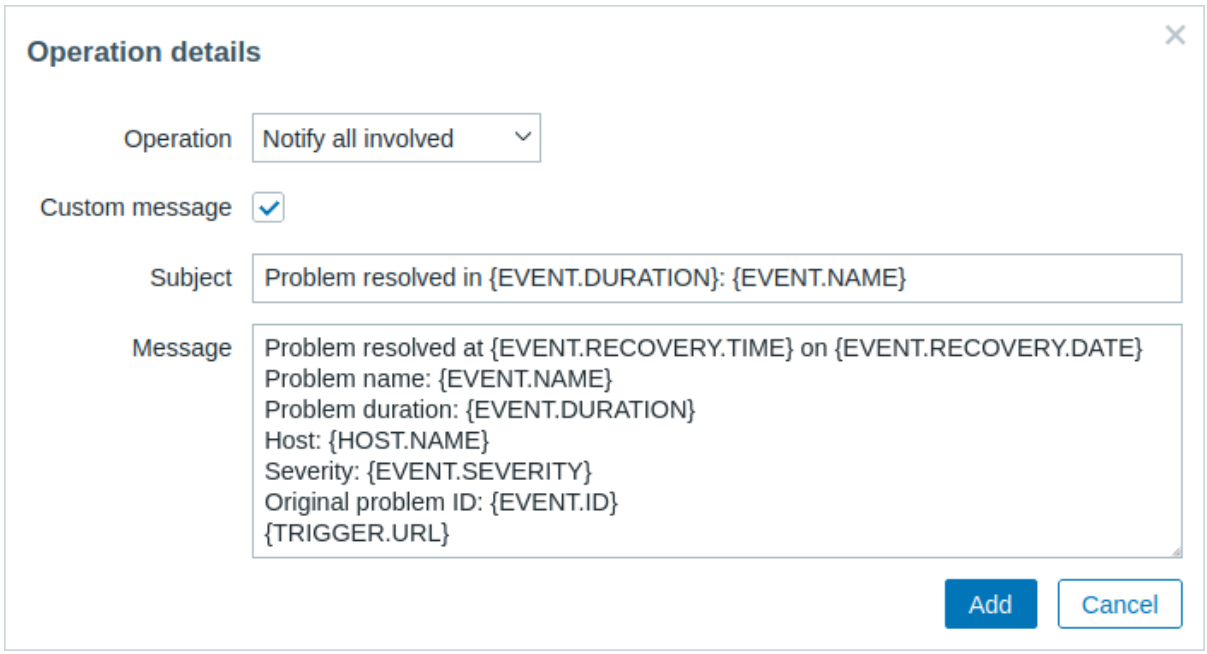
Настройка операции восстановления

Для настройки операции восстановления используйте следующие шаги:

1. Перейдите на вкладку *Операции (Operations)* в настройках действия.
2. Чтобы настроить новую операцию восстановления, нажмите *Добавить (Add)* в блоке *Операции восстановления (Recovery operations)*. Чтобы изменить существующую операцию, нажмите *Изменить (Edit)* сразу после операции.
3. Настройте подробности операции.




Детали операции восстановления




Для событий восстановления доступны три типа операций:

- **Отправить сообщение** - отправить сообщение о восстановлении указанному пользователю;
- **Уведомить всех причастных** - отправить сообщение о восстановлении всем пользователям, которые были уведомлены о событии проблемы;
- **<remote command name>** - выполнить удаленную команду. Команды доступны для выполнения, если они ранее определены в глобальных скриптах с выбранной областью действия *Action operation*.

Параметры для каждого типа операции описаны ниже. Все обязательные поля ввода отмечены красной звездочкой. После завершения нажмите *Add*, чтобы добавить операцию в список *Recovery operations*.

**Note:**
Обратите внимание: если один и тот же получатель указан в нескольких типах операций без заданного *Custom message*, дублирующиеся уведомления не отправляются.

Тип операции: отправить сообщение

| Параметр | Описание |
|---|---|
| *Отправить группам пользователей* | Выберите группы пользователей, которым нужно отправить сообщение о восстановлении.<br>Группа пользователей должна иметь как минимум права "read" permissions на узел сети, чтобы получить уведомление. |
| *Отправить пользователям* | Выберите пользователей, которым нужно отправить сообщение о восстановлении.<br>Пользователь должен иметь как минимум права "read" permissions на узел сети, чтобы получить уведомление. |
| *Отправить через тип медиа* | Отправить стандартное сообщение о восстановлении всем доступным (настроенным и включенным) типам медиа или только определенному. |
| *Пользовательское сообщение* | Если выбрано, можно определить пользовательское сообщение. |
| *Тема* | Тема пользовательского сообщения. Тема может содержать макросы. |
| *Сообщение* | Пользовательское сообщение. Сообщение может содержать макросы. |

Тип операции: удалённая команда

| Параметр | Описание |
|---|---|
| *Список целей (Target list)* | Выберите цели для выполнения команды:<br>**Текущий узел сети (Current host)** — команда выполнится на узле сети триггера, который вызвал событие о проблеме. Эта опция не будет работать, если в выражении триггера имеется несколько узлов сети.<br>**Узел сети (Host)** — выберите узел(ы) сети для выполнения команды.<br>**Группа узлов сети (Host group)** — выберите группу(ы) узлов сети для выполнения команды. При указании родительской группы узлов сети, неявным образом будут выбраны все вложенные группы узлов сети. Таким образом, команда будет выполнена на узлах сети из вложенных групп.<br>Команда на узле сети выполняется только один раз, даже если узел сети соответствует более одного раза (например, из нескольких групп узлов сети; отдельно и из группы узлов сети).<br>Список целей не имеет смысла, если команда выполняется на стороне Zabbix сервера. В этом случае выбор нескольких целей приведёт только к тому, что команда выполнится на сервере несколько раз.<br>Обратите внимание, что в случае глобальных скриптов выбор цели также зависит от настройки *Группа узлов сети (Host group)* в конфигурации глобального скрипта. |

Тип операции: оповещать всех участников

| Параметр | Описание |
|---|---|
| *Пользовательское сообщение (Custom message)* | Если выбрано, будет использовано пользовательское сообщение. |
| *Тема (Subject)* | Тема пользовательского сообщения. Тема может содержать макросы. |
| *Сообщение (Message)* | Пользовательское сообщение. Сообщение может содержать макросы. |

#### 4. Операции обновления

Обзор

Операции обновления доступны в действиях со следующими источниками событий:

- *Триггеры* — когда проблемы обновляются другими пользователями, то есть при добавлении комментария, подтверждения, изменения важности, закрытия (вручную);
- *Услуги* — когда важность услуги меняется, но сама услуга еще не восстановилась.

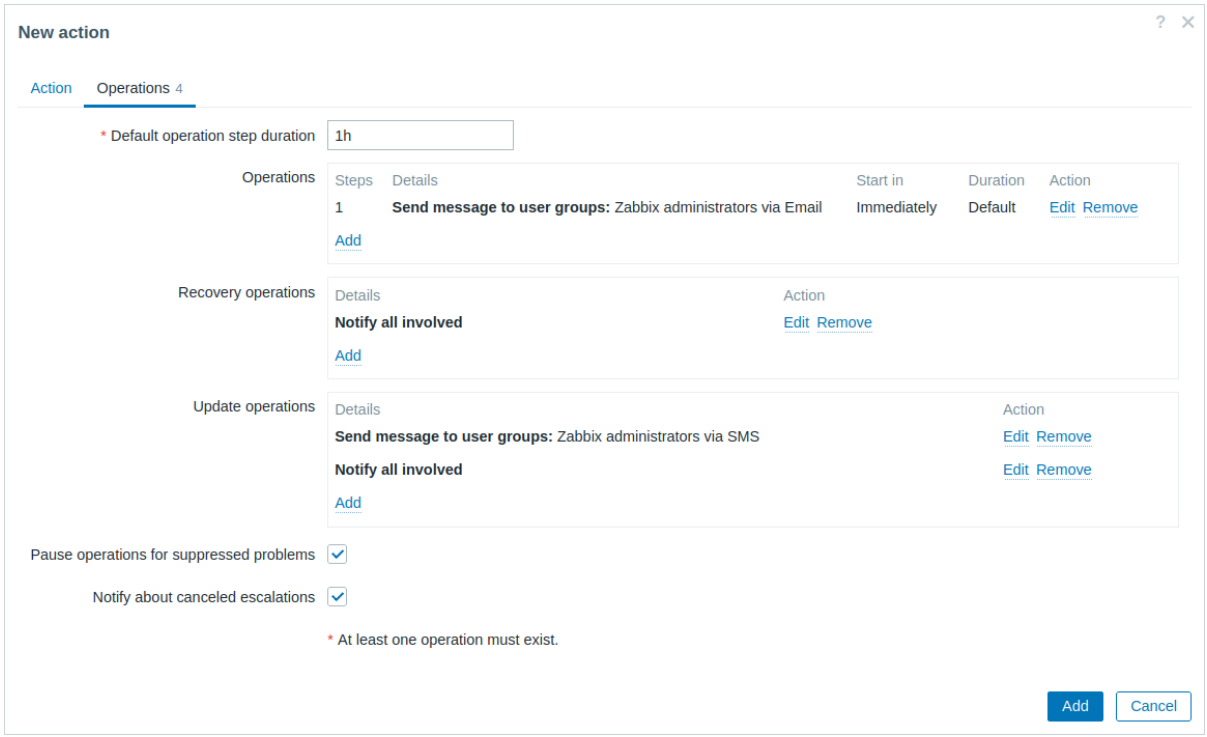
Обратите внимание, что пользователи не получают уведомления о своих собственных обновлениях.

В операциях обновления поддерживаются как сообщения, так и удалённые команды. Хоть и можно добавить несколько операций, эскалации не поддерживаются — все операции назначаются единственному шагу и поэтому будут выполнены одновременно.

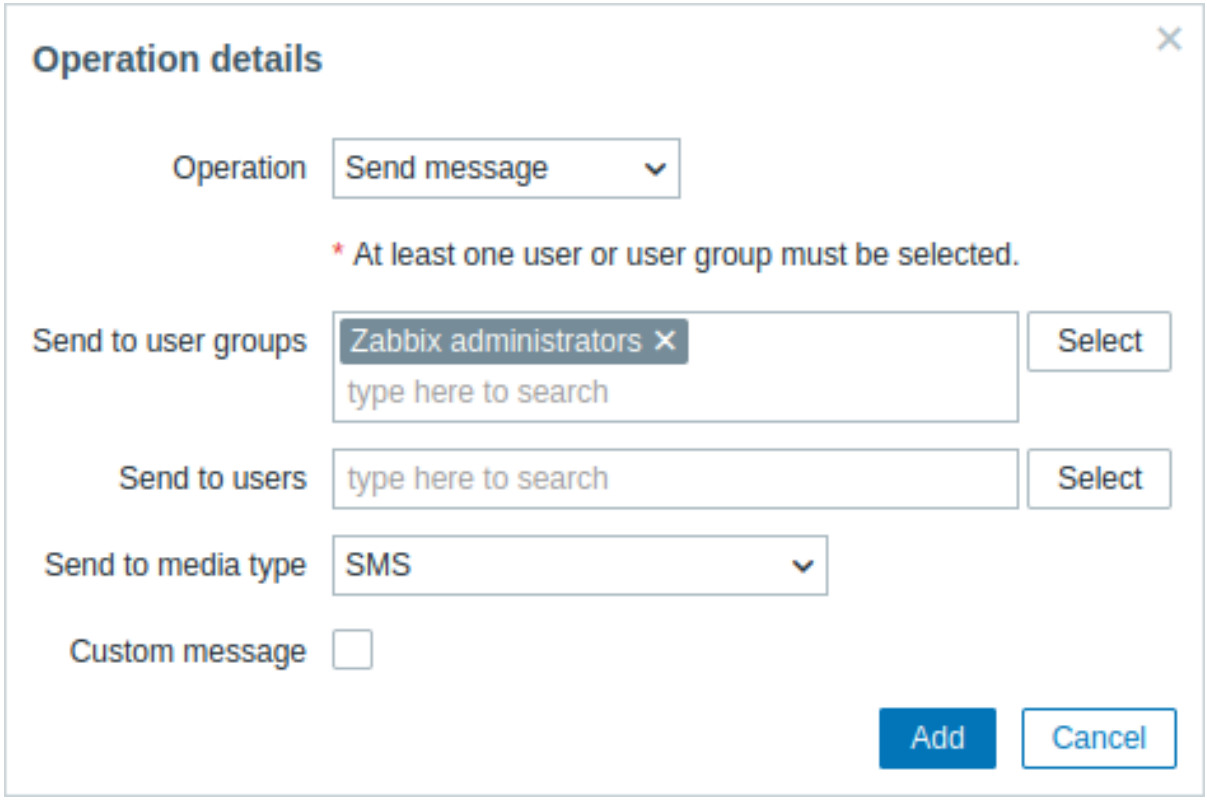
Настройка операции обновления

Для настройки операции обновления используйте следующие шаги:

1. Перейдите на вкладку *Операции (Operations)* в настройках действия.
2. Чтобы настроить новую операцию обновления, нажмите *Добавить (Add)* в блоке *Операции обновления (Update operations)*. Чтобы изменить существующую операцию, нажмите *Изменить (Edit)* сразу после операции.
3. Настройте подробности операции.




Подробности операции обновления




Операции обновления предлагают тот же набор параметров, что и операции восстановления.

**5. Эскалации**

Обзор

Используя эскалации, вы можете создавать пользовательские сценарии для отправки оповещений или выполнения удалённых команд.

С практической точки зрения эскалации означают, что:

- Пользователей можно информировать о новых проблемах незамедлительно.
- Оповещения могут повторяться до решения проблемы.
- Отправка оповещения может быть выполнена с задержкой.
- Оповещения могут эскалироваться другой «более высокой» группе пользователей.
- Удалённые команды могут быть выполнены незамедлительно или когда проблема не будет решена за длительный период времени.

Действия эскалируются на основании **шага эскалации**. Каждый шаг имеет длительность по времени.

Вы можете задать и длительность по умолчанию, и пользовательскую длительность для каждого отдельного шага. Минимальная длительность одного шага эскалации 60 секунд.

Вы можете начать действия, такие как отправка оповещения или выполнение команд, с любого шага. Шаг первый используется для немедленных действий. Если вы хотите отложить действие, вы можете назначить его на следующий шаг. Для каждого шага можно назначать несколько действий.

Количество шагов эскалаций не ограничено.

Эскалации задаются при настройке операции. Эскалации поддерживаются только для операций на проблемы, но не восстановления.

Различные аспекты поведения эскалаций

Давайте рассмотрим, что произойдёт при разных обстоятельствах, если действие содержит несколько шагов эскалаций.

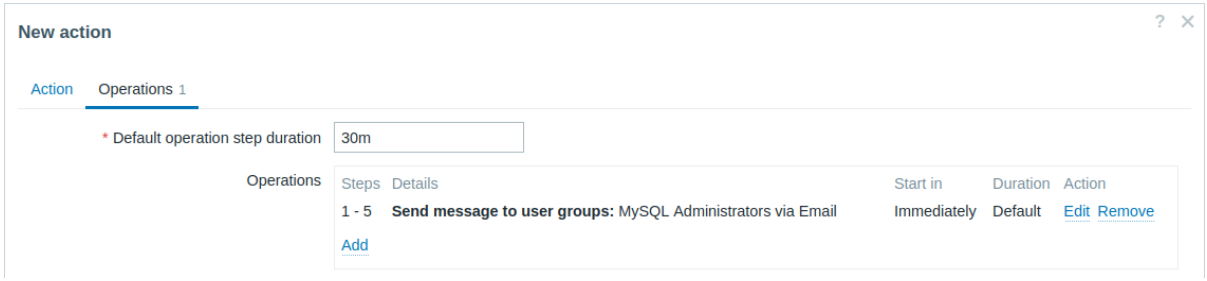
| Ситуация | Поведение |
|---|---|
| *Узел сети, о котором идёт речь, переходит в состояние обслуживания после отправки первоначального оповещения о проблеме* | В зависимости от опции *Приостановить операции для подавленных проблем* в настройках действия, все оставшиеся шаги эскалаций выполнятся либо с задержкой, вызванной периодом обслуживания, либо без задержки. Период обслуживания не отменяет операции. |
| *Период времени, указанный в условии **Период времени** действия, завершается после первоначальной отправки оповещения* | Выполнятся все оставшиеся шаги эскалации. Условие *Период времени* не может прервать операции; это условие влияет на то, когда начинается или не начинается действие, а не операция. |
| *Проблема началась в течение периода обслуживания и продолжается (не исправлена) после окончания обслуживания* | В зависимости от опции *Приостановить операции для подавленных проблем* в настройках действия, все шаги выполнятся либо с момента завершения обслуживания, либо сразу же. |
| *Проблема началась в течение периода обслуживания без сбора данных и продолжается (не исправлена) после окончания обслуживания* | Прежде чем выполнятся все шаги эскалаций, нужно дождаться срабатывания триггера. |
| *Разные эскалации следуют в тесном порядке и перекрываются* | Выполнение каждой новой эскалации заменяет предыдущую эскалацию, но по крайней мере для одного шага эскалации, который всегда обязательно выполняется при предыдущей эскалации. Такое поведение уместно для действий на события, которые создаются на КАЖДОЕ вычисление проблемы триггера. |
| *В процессе выполнения эскалации (например, во время отправки сообщения), основанном на любом типе событий:<br>- действие деактивировано<br>На основе события на триггер:<br>- триггер деактивирован<br>- узел сети или элемент данных деактивирован<br>На основе внутреннего события о триггере:<br>- триггер деактивирован<br>На основе внутреннего события об элементах данных/правилах низкоуровневого обнаружения:<br>- элемент данных деактивирован<br>- узел сети деактивирован* | Будет отправлено сообщение, которое уже в процессе отправки, затем будет отправлено ещё одно сообщение на эскалацию. Это последующее сообщение будет иметь следующий текст в начале тела сообщения (*NOTE: Escalation cancelled*) с указанием причины (например, *NOTE: Escalation cancelled: action '<Имя действия>' disabled*). Таким образом получатель будет проинформирован о том, что эскалация отменена и дальнейшие шаги не будут выполнены. Это сообщение будет отправлено всем, кто получал оповещения ранее. Причина отмены также будет записана в файл журнала сервера (начиная с уровня отладки 3=Предупреждения).<br><br>Обратите внимание, что сообщение *Escalation canceled* также будет отправлено, если все операции завершены, а операции восстановления настроены, но ещё не выполнялись. |
| *В процессе выполнения эскалации (например, во время отправки сообщения) удалено действие* | Сообщения более не отсылаются. Информация будет записана в файл журнала сервера (начиная с уровня отладки 3=Предупреждения), например: `escalation cancelled: action id:334 deleted` |

Примеры эскалаций

Пример 1

Отправка повторяющихся оповещений каждые 30 минут (в общей сложности 5 раз) группе «MySQL администраторы». Для настройки:

- На вкладке *Операции (Operations)* задайте *Длительность шага операции по умолчанию (Default operation step duration)* значением «30m» (30 минут)
- Укажите шаги эскалаций *С (From)* «1» *До (To)* «5»
- Выберите группу «MySQL администраторы» получателями сообщений




Оповещения будут отправлены через 0:00, 0:30, 1:00, 1:30, 2:00 часов после начала проблемы (если, конечно, проблема не будет решена раньше).

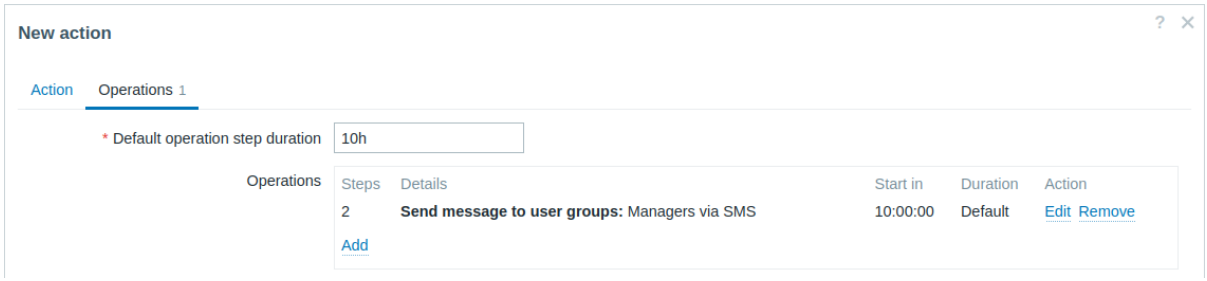
Если проблема решена и сообщение о восстановлении настроено, оно будет отправлено всем тем, кто получил хотя бы одно сообщение в этом сценарии эскалаций.

**Note:**
Если триггер, который вызвал активную эскалацию, был деактивирован, Zabbix отправит информационное сообщение об этом всем, кто уже получил оповещения.

Пример 2

Отправка оповещения с задержкой о давней проблеме. Для настройки:

- На вкладке *Операции (Operations)* задайте *Длительность шага операции по умолчанию (Default operation step duration)* значением «10h» (10 часов)
- Укажите шаги эскалации *С (From)* «2» *До (To)* «2»



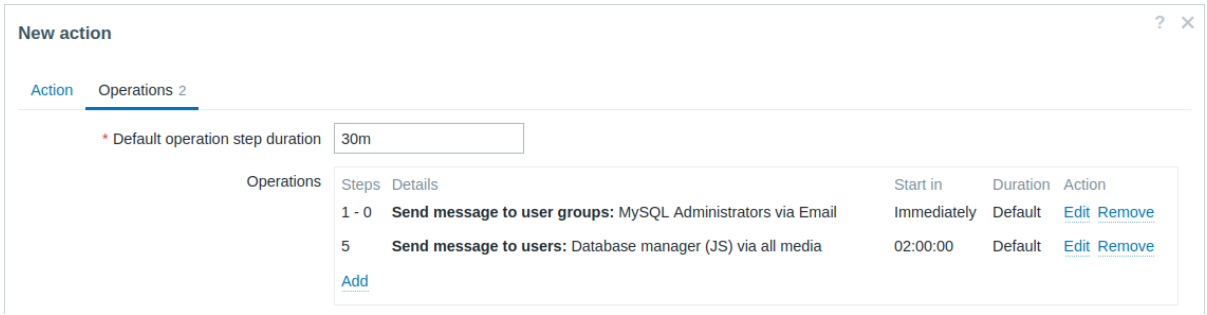

Оповещение будет отправлено только на Шаге 2 сценария эскалации, или через 10 часов после начала проблемы.

Вы можете изменить текст сообщения на что-то вроде «Проблема не решена уже более чем 10 часов».

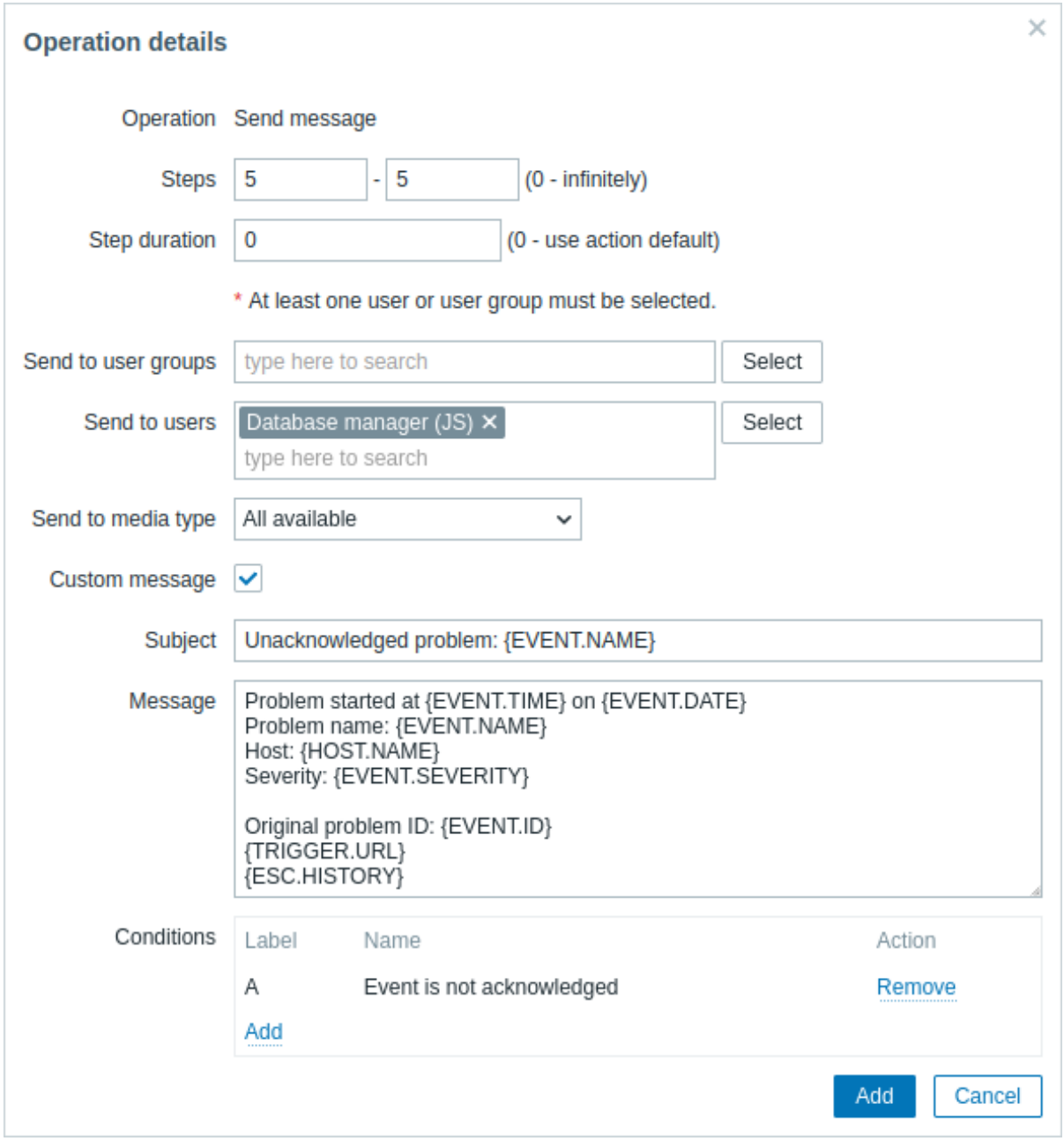
Пример 3

Эскалирование проблемы Руководству.

В первом примере мы настраивали периодическую отправку сообщений Администраторам MySQL. В данном же случае администраторы получат четыре сообщения до того, как проблема будет эскалирована Менеджеру баз данных. Обратите внимание, что менеджер получит сообщение только в случае, если проблема ещё не подтверждена, предположительно, никто не работает над ней.




Детали Операции 2:



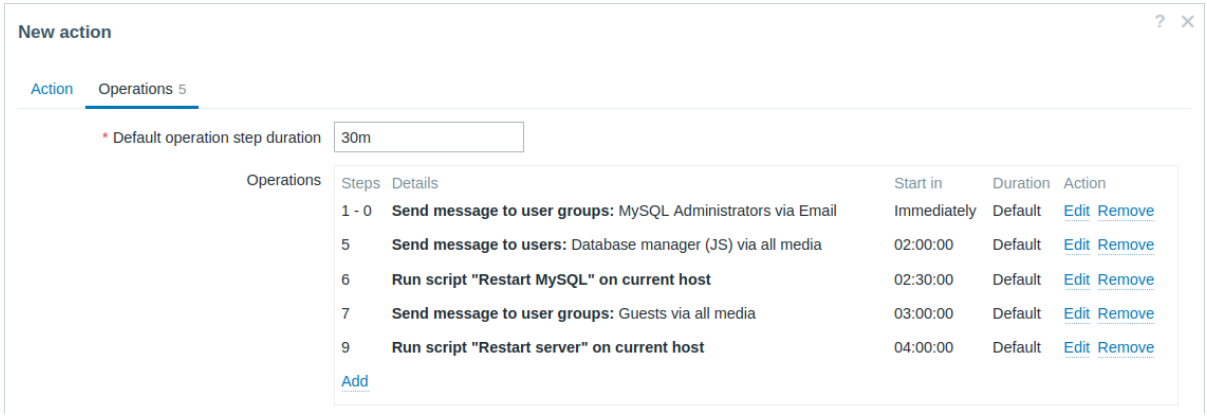

Обратите внимание на использование макроса {ESC.HISTORY} в сообщении. Этот макрос будет содержать информацию обо всех ранее выполненных шагах этой эскалации, таких как отправленные оповещения и выполненные команды.

Пример 4

Более сложный сценарий. После нескольких сообщений Администраторам MySQL и эскалации менеджеру, Zabbix попытается перезапустить базу данных MySQL. Это произойдёт, если через 2:30 часа проблема присутствует и не была подтверждена.

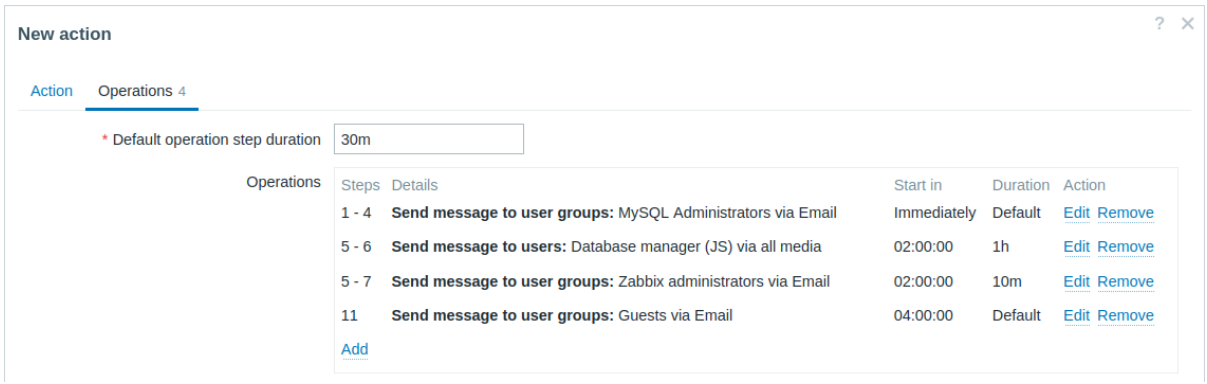
Если проблема всё ещё существует, спустя ещё 30 минут Zabbix отправит сообщение всем гостевым пользователям.

Если и это не поможет, спустя ещё час Zabbix перезагрузит сервер с этой базой данных MySQL (вторая удалённая команда), используя IPMI команды.




Пример 5

Эскалация с несколькими операциями, которые имеют перекрывающиеся диапазоны шагов и пользовательские интервалы. Длительность шага операции по умолчанию равна 30 минутам.




Оповещения будут отправлены в следующем порядке:

- Администраторам MySQL через 0:00, 0:30, 1:00, 1:30 после начала проблемы.
- Менеджеру баз данных через 2:00 и 2:10 (более короткая пользовательская длительность шага в 10 минут, заданная в последующей операции, перекрывает более длительную длительность шага в 1 час, настроенную здесь, как указано в разделе Детали операций для параметра *Длительность шага (Step duration)*, если шаги перекрываются).
- Zabbix администраторам через 2:00, 2:10 и 2:20 после начала проблемы (применяется пользовательская длительность шага, равная 10 минутам).
- Гостевым пользователям через 4:00 часа после начала проблемы (между шагами 8 и 11 имеет силу интервал по умолчанию, равный 30 минутам).

### 3. Получение оповещений о неподдерживаемых элементах данных

Обзор

В Zabbix есть возможность получать оповещения о неподдерживаемых элементах данных.

Это часть нового концепта внутренних событий в Zabbix, позволяющая пользователям получать оповещения о таких случаях. Внутренние события отражают изменение состояния:

- при переходе элемента данных из состояния «активировано» в состояние «не поддерживается» (и обратно);
- при переходе триггера из состояния «активировано» в состояние «неизвестно» (и обратно);
- при переходе правила низкоуровневого обнаружения из состояния «активировано» в состояние «не поддерживается» (и обратно).

Этот раздел демонстрирует, как **получать оповещения**, когда элемент данных становится неподдерживаемым.

Настройка

В целом, процесс настройки оповещений должен быть знаком тем, кто раньше настраивал оповещения в Zabbix.

Шаг 1

Настройте какой-либо способ оповещения, такой как email, SMS или скрипт, для использования в оповещениях. Для выполнения этой задачи обратитесь к соответствующим разделам документации.

**Attention:**
При оповещении о внутренних событиях используется важность по умолчанию («Не классифицировано»), поэтому оставьте эту важность выбранной при настройке оповещения пользователя, если вы хотите получать оповещения о внутренних событиях.

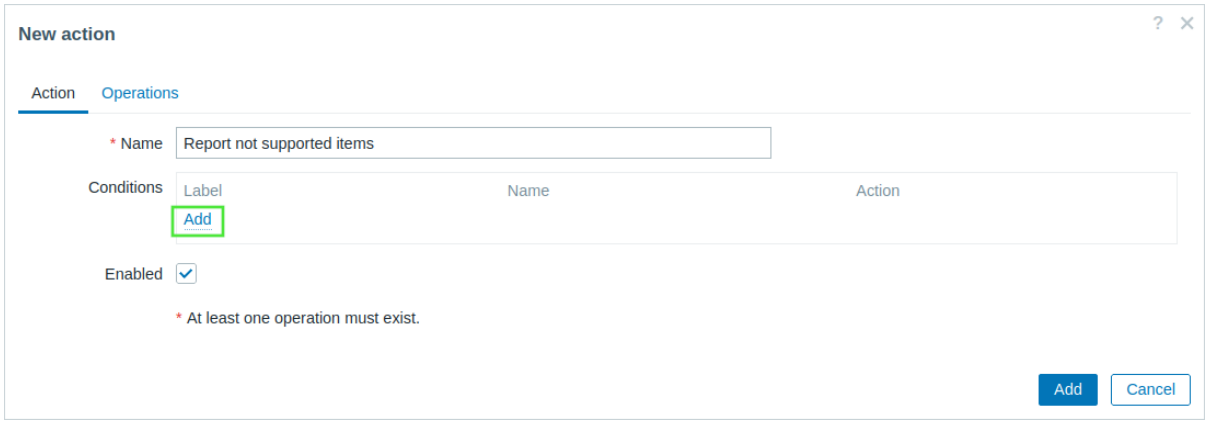
Шаг 2

Перейдите в *Alerts* → *Actions* → *Internal actions*.

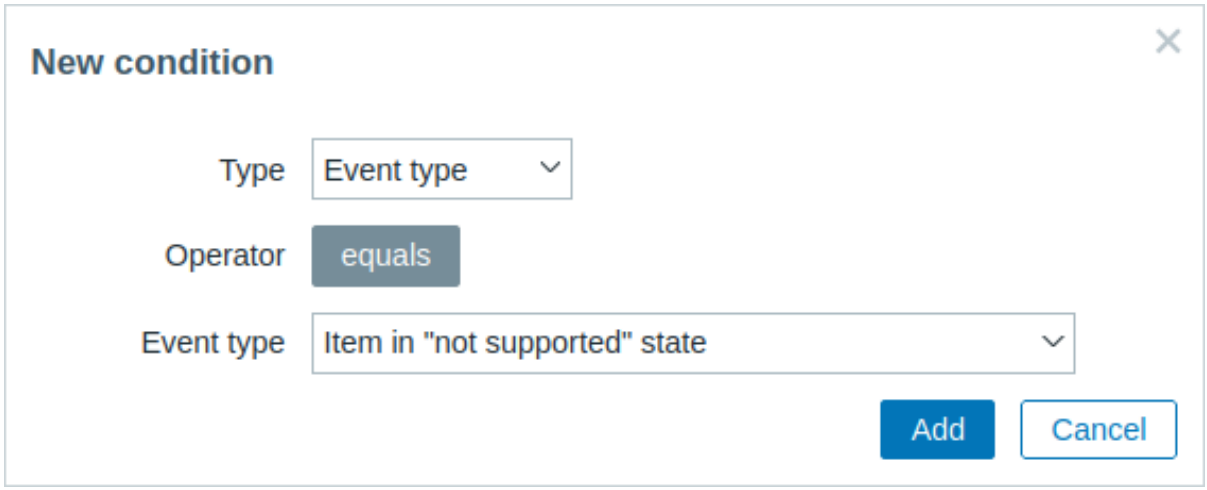
Нажмите *Create action* в правом верхнем углу страницы, чтобы открыть форму настройки действия.

Шаг 3

На вкладке *Action* введите имя для действия. Затем нажмите *Add* в блоке *Conditions*, чтобы добавить новое условие.



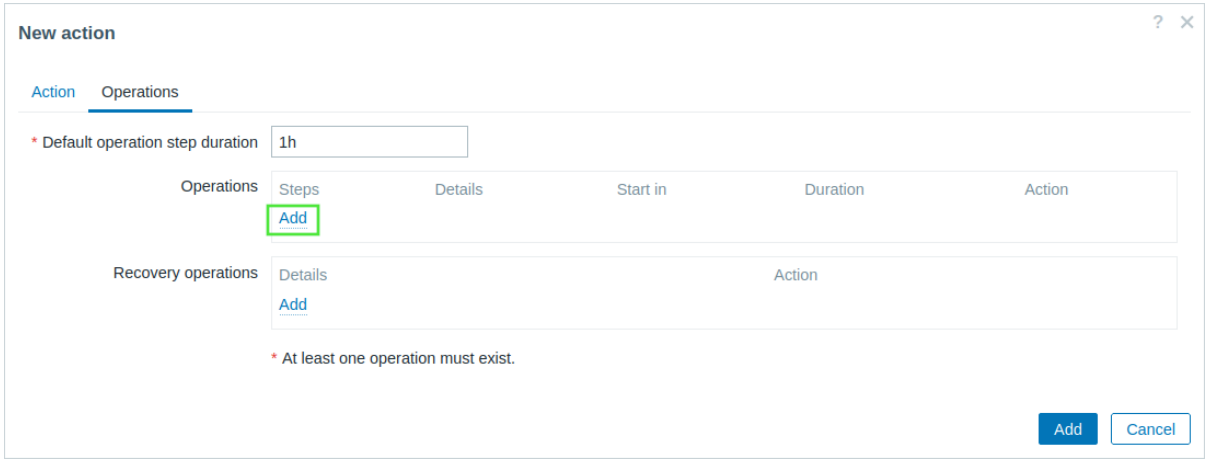

Во всплывающем окне *New condition* выберите ”Event type” в качестве типа условия, а затем выберите ”Item in ’not supported’ state” в качестве типа события.



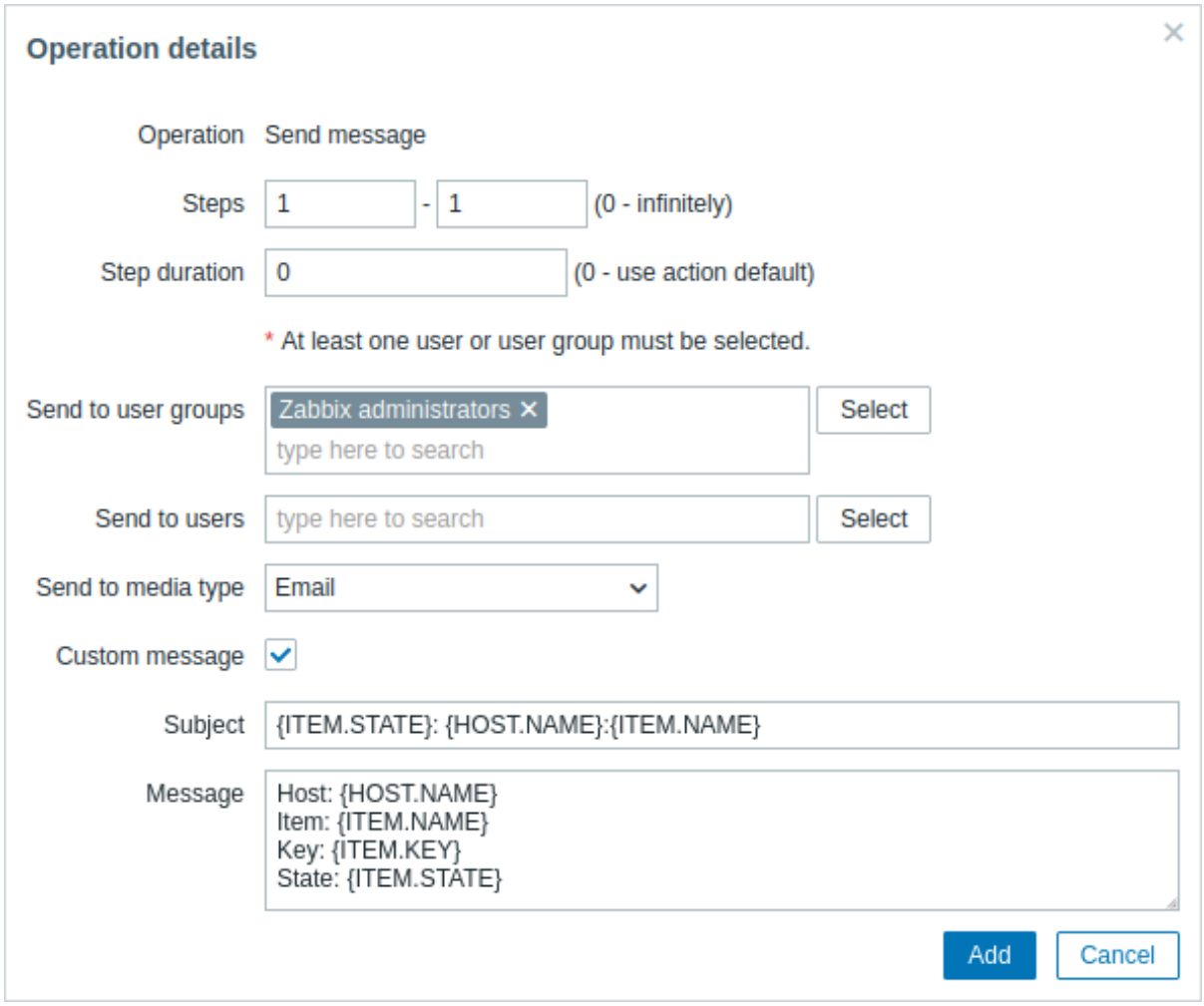

Не забудьте нажать *Add*, чтобы действительно добавить условие в блок *Conditions*.

Шаг 4

На вкладке *Operations* нажмите *Add* в блоке *Operations*, чтобы добавить новую операцию.




Выберите получателей сообщения (группы пользователей/пользователей) и тип медиа (или "All") для доставки. Установите флажок *Custom message*, если хотите указать собственные тему/содержимое сообщения о проблеме.

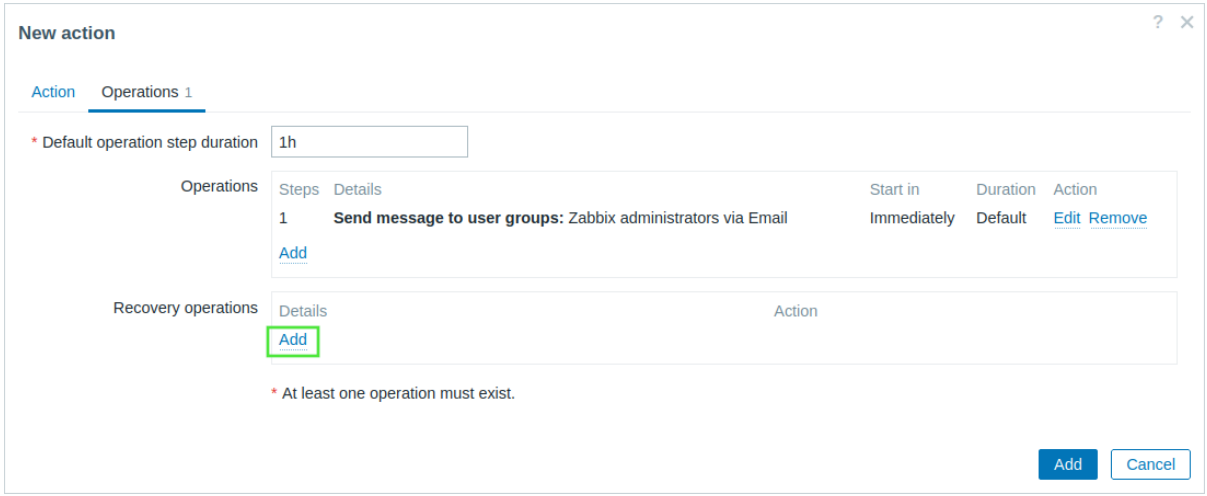



Нажмите *Add*, чтобы действительно добавить операцию в блок *Operations*.

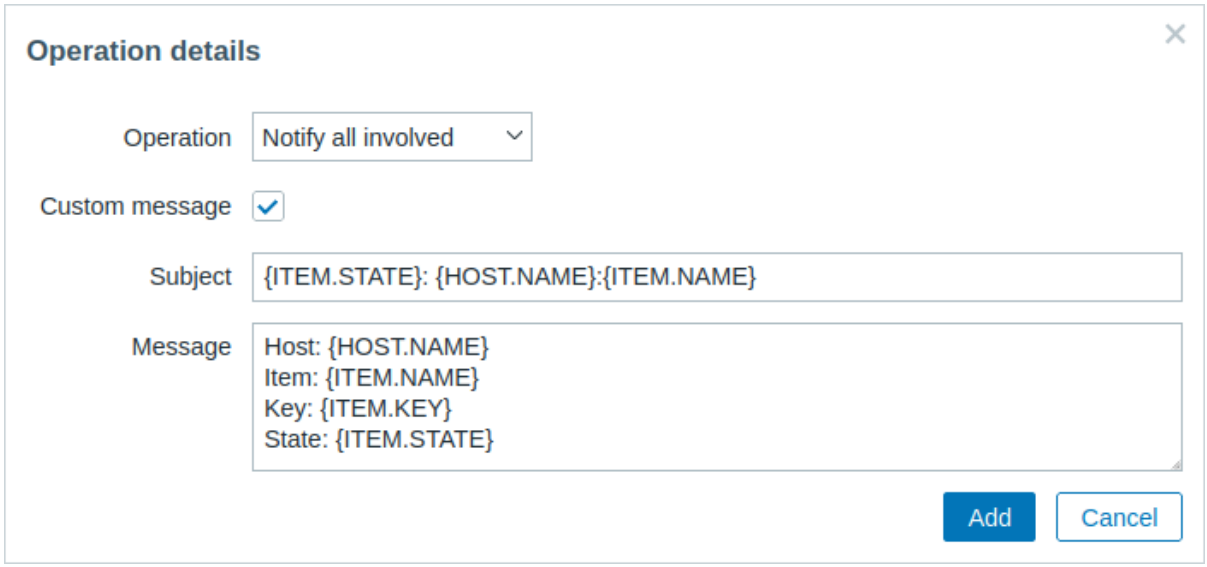
Если вы хотите получать более одного уведомления, задайте длительность шага операции (интервал между отправляемыми сообщениями) и добавьте еще один шаг.

Шаг 5

Блок *Операции восстановления* позволяет настроить уведомление о восстановлении, когда элемент данных возвращается в нормальное состояние. Нажмите *Добавить* в блоке *Операции восстановления*, чтобы добавить новую операцию восстановления.



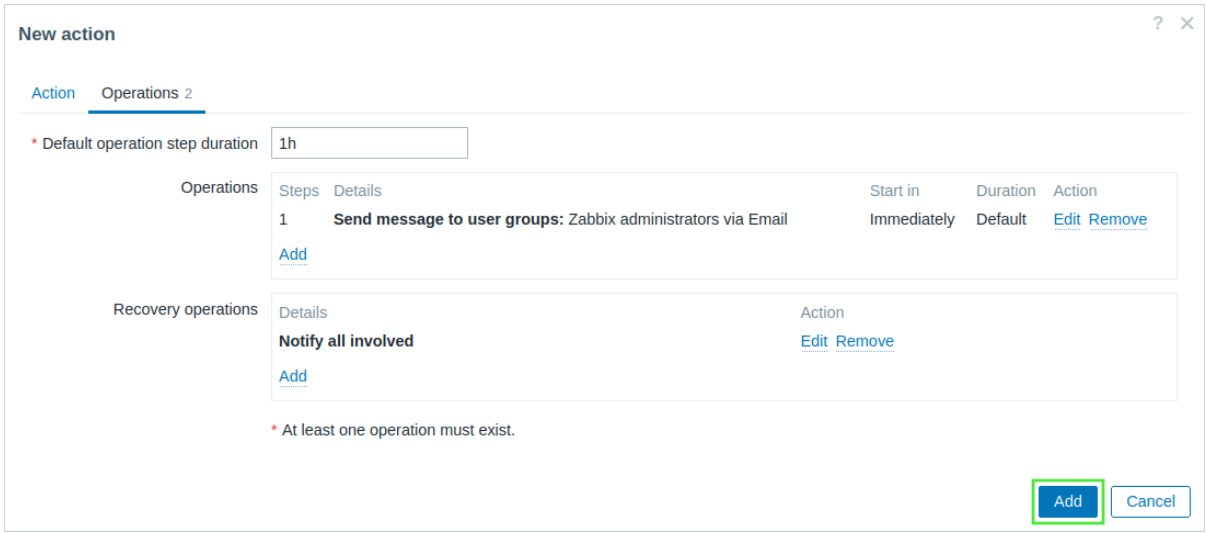

Выберите тип операции ”Notify all involved”. Установите флажок *Пользовательское сообщение*, если хотите указать собственные тему и содержимое сообщения о восстановлении.




Нажмите *Добавить* во всплывающем окне *Детали операции*, чтобы фактически добавить операцию в блок *Операции восстановления*.

Шаг 6

Когда закончите, нажмите кнопку *Add* в нижней части формы.




Вот и все, готово! Теперь вы можете ожидать получения первого уведомления от Zabbix, если какой-либо элемент данных станет неподдерживаемым.

## 11. Макросы

Обзор

Zabbix поддерживает ряд встроенных макросов, которые можно использовать в различных ситуациях. Макросами являются переменные, которые имеют специальный синтаксис:

{ }

Макросы раскрываются в требуемое значение в зависимости от контекста.

Эффективное использование макросов позволяет сохранить время и делает настройку Zabbix более прозрачной.

Одно из типичных применений макросов — в шаблонах. Так, триггер из шаблона может быть назван: «Слишком высокая загрузка процессора на {HOST.NAME}». Когда шаблон применяется к узлу сети, такому как Zabbix сервер, при отображении триггера в разделе Мониторинга такое имя будет раскрыто в «Слишком высокая загрузка процессора на Zabbix сервер».

Макросы можно использовать в параметрах ключей элементов данных. Макрос можно использовать как только часть параметра, например: `item.key[server_{HOST.HOST}_local]`. Двойные кавычки не являются обязательными, так как Zabbix позаботится о неоднозначных спецсимволах, если они присутствуют в раскрытом макросе.

В Zabbix также имеются и другие типы макросов.

Zabbix поддерживает следующие макросы:

- { } — встроенные макросы (смотрите полный список)
- {< >.< >(< >)} — функции макросов
- {$ } — пользовательские макросы, опционально с контекстом
- {# } — макросы для низкоуровневого обнаружения
- {? } — макросы выражений

### 1 Функции макросов

Обзор

Функции макросов позволяют настраивать значения макросов (например, укорачивать их или извлекать определенные подстроки), что упрощает работу с ними.

Синтаксис функции макроса:

```
{macro.func(params)}
```

где

- **macro** - макрос, который нужно настроить;
- **func** - функция, которую нужно применить (см. поддерживаемые функции);
- **params** - список параметров функции, разделенных запятыми; они должны быть **заключены в двойные кавычки**, если:
  - начинаются с пробела или двойной кавычки;
  - содержат закрывающую скобку или запятую.

Например:

```
{{TIME}.fmttime(format,time_shift)}
{{ITEM.VALUE}.regsub(pattern, output)}
{{$USERMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.regsub(pattern, output)}
```

Функции макросов поддерживаются для

- встроенных макросов
- пользовательских макросов
- макросов низкоуровневого обнаружения
- макросов выражений

Функции макросов можно использовать во всех местах, где поддерживаются перечисленные макросы. Это применимо, если явно не указано, что ожидается только макрос (например, при настройке макросов узла сети или фильтров правила низкоуровневого обнаружения).

Для каждого макроса поддерживается только одна функция; цепочки из нескольких функций макросов не поддерживаются.

**Note:**
См. примеры экранирования для случаев, когда функции макросов используются внутри других контекстов (функция, ключ элемента данных, другой макрос и т. д.).

Поддерживаемые функции

Функции перечислены без дополнительной информации. Нажмите на функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| btoa | Кодирование значения макроса в Base64. |
| fmtnum | Форматирование числа для управления количеством цифр, выводимых после десятичной точки. |
| fmttime | Форматирование времени. |
| htmldecode | Декодирование значения макроса из HTML-кодировки. |
| htmlencode | Кодирование значения макроса в HTML-кодировку. |
| iregsub | Извлечение подстроки по совпадению с регулярным выражением (без учета регистра). |
| lowercase | Преобразование символов значения макроса в строчные. |
| regrepl | Замена символа/подстроки в значении макроса. |
| regsub | Извлечение подстроки по совпадению с регулярным выражением (с учетом регистра). |
| tr | Транслитерация символов значения макроса. |
| uppercase | Преобразование символов значения макроса в прописные. |
| urldecode | Декодирование значения макроса из URL-кодировки. |
| urlencode | Кодирование значения макроса в URL-кодировку. |

Подробности функций

Необязательные параметры функций обозначены угловыми скобками (< >).

btoa

Кодирование значения макроса в Base64. Кодирование Base64 — это способ представления двоичных данных в виде текста, который полезен для хранения и безопасной передачи двоимого содержимого по текстовым протоколам.

Пример:

```
{{ITEM.VALUE}.btoa()} -            ,      "zabbix",  Base64,                "emFiYml4"
```

fmtnum(цифры)

Форматирование числа для управления количеством цифр, выводимых после десятичной точки.

Параметры:

- **цифры** — количество цифр после десятичной точки. Допустимый диапазон: 0-20. Конечные нули будут обрезаны.

Примеры:

```
{{ITEM.VALUE}.fmtnum(2)} —      «24.35»              «24.3483523»
{{ITEM.VALUE}.fmtnum(0)} —      «24»               «24.3483523»
```

fmttime(format,<time_shift>)

Форматирование времени.<br> Обратите внимание, что эту функцию можно использовать с макросами, которые преобразуются в значение в одном из следующих форматов времени:

- `hh:mm:ss`
- `yyyy-mm-ddThh:mm:ss[tz]` (стандарт ISO8601)
- UNIX timestamp

Параметры:

- **format** - обязательная строка формата, совместимая с форматированием функции `strftime`;
- **time_shift** (необязательно) - сдвиг времени, применяемый к времени перед форматированием; должен начинаться с `-<N><time_unit>` или `+<N><time_unit>`, где:
  - `N` - количество единиц времени, которые нужно прибавить или вычесть;
  - `time_unit` - h (час), d (день), w (неделя), M (месяц) или y (год).

Комментарии:

- Параметр time_shift поддерживает многошаговые операции со временем и может включать /<time_unit> для сдвига к началу единицы времени (/d - полночь, /w - первый день недели (понедельник), /M - первый день месяца и т. д.). Примеры: -1w - ровно 7 дней назад; -1w/w - понедельник предыдущей недели; -1w/w+1d - вторник предыдущей недели.
- Операции со временем вычисляются слева направо без приоритетов. Например, -1M/d+1h/w будет разобрано как ((-1M/d)+1h)/w.

Примеры:

```
{{TIMESTAMP}.fmttime(%B)} -     "October"               "1633098961"
{{TIMESTAMP}.fmttime(%d %B,-1M/M)} -     "1 September"               "1633098961"
```

htmldecode

Декодирование значения макроса из HTML-кодировки.

Поддерживаются следующие символы:

| Value | Decoded value |
|---|---|
| & | & |
| < | < |
| > | > |
| " | " |
| ' | ' |
| ' | ' |

Пример:

```
{{ITEM.VALUE}.htmldecode()} -      HTML-            ,      "&lt;"   "<"
```

htmlencode

Кодирование значения макроса в HTML-кодировку.

Поддерживаются следующие символы:

| Value | Encoded value |
|---|---|
| & | & |
| < | < |
| > | > |
| " | " |
| ' | ' |

Пример:

```
{{ITEM.VALUE}.htmlencode()} -      HTML-           ,      "<",   "&lt;"
```

iregsub(шаблон,вывод)

Извлечение подстроки помощью сопоставления с регулярным выражением (без учёта регистра).

Parameters:

- **шаблон** — регулярное выражение для поиска совпадения;
- **вывод** — опции вывода. Для захвата групп поддерживаются местозаменители **\1 - \9**. Заменитель **\0** вернёт совпавший текст.

Комментарии:

- Если нет совпадений с регулярным выражением, функция возвращает пустую строку.
- Если шаблон функции не является корректным регулярным выражением, то макрос раскрывается в «НЕИЗВЕСТНО» (за исключением макросов низкоуровневого обнаружения, где в этом случае функция будет проигнорирована и макрос останется нераскрытым).
- Ссылки на несуществующие скобочные группы в строке замены заменяются пустой строкой.

Пример:

```
{{ITEM.VALUE}.iregsub("fail|error|fault|problem","ERROR")} —           «ERROR»,              «fail», «er
```

lowercase

Преобразование всех символов значения макроса в нижний регистр.
Работает с однобайтовыми наборами символов (например, ASCII) и не поддерживает UTF-8.

Пример:

```
{{ITEM.VALUE}.lowercase()} -            ,      "Zabbix SERVER",   "zabbix server" (lowercase)
```

regrepl(pattern,replacement,<pattern2>,<replacement2>,...)

Замена символа/подстроки в значении макроса.

Параметры:

- **pattern** - регулярное выражение для сопоставления;
- **replacement** - строка замены. В строках замены поддерживаются заполнители **\1 - \9** для захвата групп.

Комментарии:

- шаблоны и замены обрабатываются последовательно, при этом каждая последующая пара применяется в соответствии с результатом предыдущей замены;
- ссылки на несуществующие группы захвата в строке замены заменяются пустой строкой.

Примеры:

```
{{ITEM.VALUE}.regrepl("oldParam", "newParam")} -      "oldParam"   "newParam"
{{ITEM.VALUE}.regrepl("([^a-z])","\\\1")} -
{$THRESHOLD:"{{#FSNAME}.regrepl(\"\\$\",\"\")}"} -                          (     ,          "C:\"   "C:")
{{ITEM.VALUE}.regrepl("_v1\.0", "_v2.0", "\(final\)", "")} -
```

regsub(шаблон,вывод)

Извлечение подстроки помощью сопоставления с регулярным выражением (с учётом регистра).

Параметры:

- **шаблон** — регулярное выражение для поиска совпадения;
- **вывод** — опции вывода. Для захвата групп поддерживаются местозаменители **\1 - \9**. Заменитель **\0** вернёт совпавший текст.

Комментарии:

- Если нет совпадений с регулярным выражением, функция ничего не возвращает.
- Если шаблон функции не является корректным регулярным выражением, то макрос раскрывается в «НЕИЗВЕСТНО» (за исключением макросов низкоуровневого обнаружения, где в этом случае функция будет проигнорирована и макрос останется нераскрытым).
- Ссылки на несуществующие скобочные группы в строке замены заменяются пустой строкой.

Примеры:

```
{{ITEM.VALUE}.regsub("^([0-9]+)", Problem ID: \1)} —           «Problem ID: 123»,                  «12
{{ITEM.VALUE}.regsub("fail|error|fault|problem","ERROR")} —           «ERROR»,              «fail», «er
```

Смотрите ещё примеры.

tr(characters,replacement)

Транслитерация значения макроса characters.

- **characters** - набор символов для замены;
- **replacement** - набор символов замены, соответствующих по позиции.

Примеры:

```
{{ITEM.VALUE}.tr(abc, xyz)} -              "a"   "x", "b"   "y", "c"   "z"
{{ITEM.VALUE}.tr(abc, xyzq)} -              "a"   "x", "b"   "y", "c"   "z" ("q"       )
{{ITEM.VALUE}.tr(abcde, xyz)} -              "a"   "x", "b"   "y", "c"   "z", "d"   "z", "e"   "z" (
{{ITEM.VALUE}.tr("\\\'", "\/\"")} -                                      ,
{{ITEM.VALUE}.tr(A-Z,a-z)} -
{{ITEM.VALUE}.tr(0-9a-z,*)} -                          "*"
{{ITEM.VALUE}.tr(0-9,ab)} -             0   "a",           1, 2, 3, 4, 5, 6, 7, 8   9   "b"
```

```
{{ITEM.VALUE}.tr(0-9abcA-L,*)} -           ,      "abc"        A-L    "*"
{{ITEM.VALUE}.tr("\n","*")} -                      *
{{ITEM.VALUE}.tr("e", "\n")} -         "e"
```

Чтобы включить буквальные символы:

```
             -                    \\
           -                    \'
         -                    \"
```

Поддерживаемые escape-последовательности с обратной косой чертой:

```
\\\\ => \\ -
\\a  => \a -
\\b  => \b - backspace
\\f  => \f -
\\n  => \n -
\\r  => \r -
\\t  => \t -
\\v  => \v -
```

верхний регистр

Преобразование всех символов значения макроса в верхний регистр.
Работает с однобайтовыми наборами символов (например, ASCII) и не поддерживает UTF-8.

Пример:

```
{{ITEM.VALUE}.uppercase()} -             ,      "Zabbix Server",   "ZABBIX SERVER" (         )
```

urldecode

Декодирование значения макроса из URL-кодировки.

Пример:

```
{{ITEM.VALUE}.urldecode()} -      URL-                  "%2F"   "/"
```

urlencode

Кодирование значения макроса в URL-кодировку.

Пример:

```
{{ITEM.VALUE}.urlencode()} -           ,      "/",   "%2F"
```

Дополнительные примеры

В таблице ниже приведено больше примеров использования функций макросов.

{#IFALIAS} — это LLD-макрос и определяется только в контекстах низкоуровневого обнаружения (правила обнаружения, прототипы и созданные на их основе элементы данных/триггеры). При использовании вне LLD токен не будет раскрыт.

| Macro function | Received value | Output |
|---|---|---|
| {{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* (invalid regular expression) |
| {{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)} | customername_1 | customername |
| {{#IFALIAS}.regsub("(.*)_([0-9]+)", \2)} | customername_1 | 1 |
| {{#IFALIAS}.regsub("(.*)_([0-9]+", \1)} | customername_1 | {{#IFALIAS}.regsub("(.*)_([0-9]+", \1)} (invalid regular expression) |

| Macro function | Received value | Output |
|---|---|---|
| {$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}"} | customername_1 | {$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)}"} (invalid regular expression) |
| "{$MACRO:"\{{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)}\"}" | customername_1 | "{$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)}\"}" (invalid regular expression) |

Просмотр полных значений элементов данных

Длинные значения раскрытых макросов {ITEM.VALUE} и {ITEM.LASTVALUE} для текстовых/логовых элементов данных в некоторых местах веб-интерфейса обрезаются до 20 символов. Чтобы увидеть полные значения этих макросов, можно использовать функции макросов, например:

```
{{ITEM.VALUE}.regsub("(.*)", \1)}
{{ITEM.LASTVALUE}.regsub("(.*)", \1)}
```

См. также: {ITEM.VALUE} и {ITEM.LASTVALUE} подробности о макросах.

#### 2. Пользовательские макросы

Обзор

Пользовательские макросы поддерживаются в Zabbix для достижения большей гибкости, в дополнение к поддерживаемым макросам «из коробки».

Пользовательские макросы можно определить на глобальном уровне, уровне шаблона и уровне узла сети. Такие макросы имеют специальный синтаксис:

```
{$    }
```

Zabbix раскрывает макросы в соответствии со следующей очерёдностью:

1. макросы, назначенные узлу сети (проверяются в первую очередь)
2. макросы, назначенные на шаблоны первого уровня узла сети (т.е. шаблоны, присоединённые напрямую к узлу сети), отсортированные по ID шаблона
3. макросы, назначенные на шаблоны второго уровня узла сети, отсортированные по ID шаблона
4. макросы, назначенные на шаблоны третьего уровня узла сети, отсортированные по ID шаблона, и так далее
5. глобальные макросы (проверяются в последнюю очередь)

Иными словами, если макрос не существует непосредственно у узла сети, тогда Zabbix попытается найти его в шаблонах, присоединённых к этому узлу сети, с учётом увеличивающейся глубины шаблонов. Если макрос всё ещё не найден, то будет использован макрос глобального уровня, если он существует.

**Warning:**
Если макросы с **одинаковыми именами** существуют у нескольких присоединённых шаблонов на одном уровне, будет использован макрос из шаблона с наименьшим значением ID. Таким образом, наличие макросов с одинаковыми именами у нескольких шаблонов — это риск конфигурации.

В случае, если Zabbix не удалось найти макрос, макрос не будет раскрыт.

**Attention:**
По задумке макросы (включая пользовательские макросы) остаются нераскрытыми в разделе Настройки (например, в списке триггеров), чтобы сделать сложную конфигурацию более прозрачной.

Пользовательские макросы можно использовать в:

- именах элементов данных
- параметрах ключей элементов данных

- интервалах обновления и гибких интервалах элементов данных
- именах и описаниях триггеров
- параметрах и константах выражений триггеров (смотрите примеры)
- многих других местах (смотрите полный список)

Общие случаи использования глобальных макросов и макросов уровня узлов сети

- используйте глобальные макросы в нескольких местах; затем измените значения этих макросов, и изменения настроек применятся во всех местах за «одно нажатие»
- получение преимуществ от шаблонов с уникальными для узлов сети атрибутами: пароли, номера портов, имена файлов, регулярные выражения и т.д.

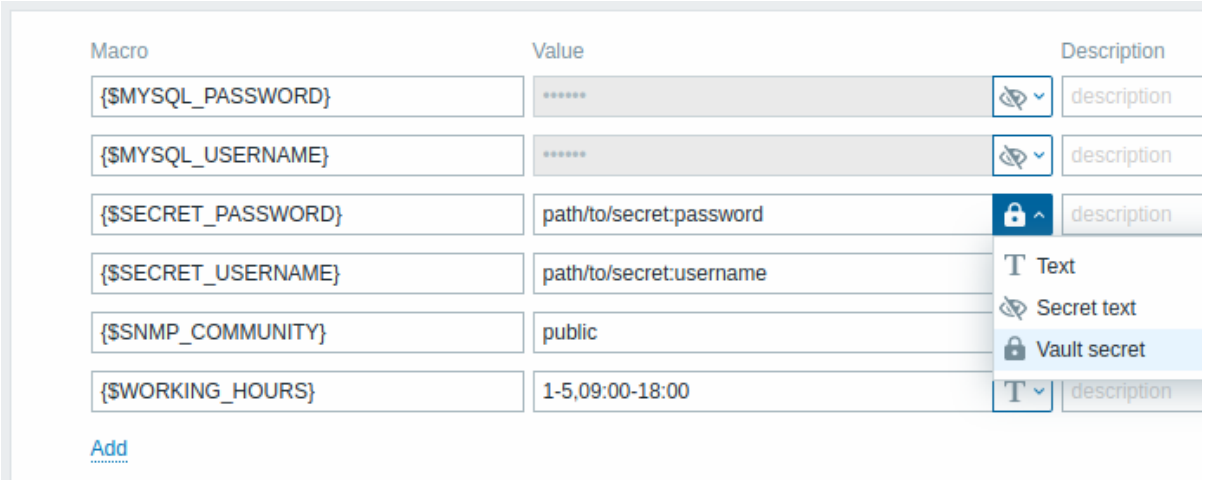
**Note:**
Рекомендуется использовать макросы узла сети вместо глобальных макросов, поскольку добавление, обновление или удаление глобальных макросов приводит к инкрементальному обновлению конфигурации для всех узлов сети. Дополнительные сведения см. в разделе Пассивные и активные проверки агента.

Конфигурация

Чтобы определить пользовательские макросы, перейдите в соответствующее место на веб-интерфейсе:

- для глобальных макросов откройте *Администрирование* → *Макросы*
- для макросов уровня узла сети и шаблона откройте свойства узла сети или шаблона и найдите вкладку *Макросы*

Пользовательский макрос имеет следующие атрибуты:




| Parameter | Description |
|---|---|
| *Macro* | Имя макроса. Имя должно быть заключено в фигурные скобки и начинаться со знака доллара.<br>Пример: {$FRONTEND_URL}. В именах макросов допускаются следующие символы: **A-Z** (только заглавные) , **0-9** , _ , **.** |
| *Value* | Значение макроса. Поддерживаются три типа значений:<br>**Text** (по умолчанию) - обычный текстовый значение<br>**Secret text** - значение скрывается звездочками<br>**Vault secret** - значение содержит путь/запрос к секрету vault.<br><br>Чтобы изменить тип значения, нажмите кнопку в конце поля ввода значения.<br><br>Максимальная длина значения пользовательского макроса составляет 2048 символов. |
| *Description* | Текстовое поле, используемое для предоставления дополнительной информации об этом макросе. |

При настройке пользовательских макросов **обратите внимание** на следующие особенности, зависящие от контекста:

- если пользовательские макросы используются в элементах данных или триггерах шаблона, рекомендуется добавить эти макросы также в шаблон (даже если они определены глобально); в этом случае макросы типа *Text* будут работать ожидаемым образом после экспорта шаблона в XML и импорта в другую систему (значения секретных макросов не экспортируются)

- если пользовательские макросы используются в выражениях триггеров, такие макросы будут подставляться только при ссылке на параметр или константу; они НЕ будут подставляться при ссылке на узел сети, ключ элемента данных, функцию, оператор или другое выражение триггера (секретные макросы нельзя использовать в выражениях триггеров)
- если пользовательские макросы используются на узле сети, для которого есть правило низкоуровневого обнаружения с прототипами узлов сети, обнаруженные узлы сети унаследуют все пользовательские макросы, определенные на этом узле сети

Примеры

Пример 1

Использование макроса уровня узла сети в ключе элемента данных «Состояние демона SSH»:

```
net.tcp.service[ssh,{$SSH_PORT}]
```

Этот элемент данных можно присоединить к нескольким узлам сети, при условии, что значение **{$SSH_PORT}** указано на этих узлах сети.

Пример 2

Использование макроса уровня узла сети в триггере «Загрузка CPU слишком высокая»:

```
last(/ca_001/system.cpu.load[,avg1])>{$MAX_CPULOAD}
```

Такой триггер можно создать на уровне шаблона, а не редактировать на отдельных узлах сети.

**Note:**
Если в качестве параметра функции вы хотите использовать количество значений (например, **max(/узел сети/ключ,#3)**), добавьте символ решётки (#) непосредственно в значение макроса, например вот так: SOME_PERIOD ⇒ #3

Пример 3

Использование двух макросов в триггере «Загрузка CPU слишком высокая»:

```
min(/ca_001/system.cpu.load[,avg1],{$CPULOAD_PERIOD})>{$MAX_CPULOAD}
```

Обратите внимание, что макрос можно использовать как параметр триггерной функции, в данном примере в функции **min()**.

Пример 4

Синхронизация условия недоступности агентов с интервалом обновления элемента данных:

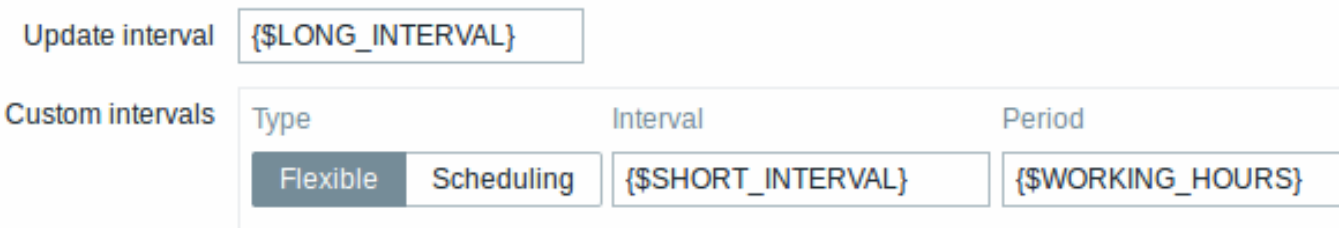
- задайте макрос {$INTERVAL} и используйте его в интервале обновления элемента данных;
- используйте {$INTERVAL} как параметр в триггере на недоступность агента:

```
nodata(/ca_001/agent.ping,{$INTERVAL})=1
```

Пример 5

Централизованная настройка рабочего времени:

- создайте глобальный макрос {$WORKING_HOURS}, равный значению 1-5,09:00-18:00;
- используйте его в поле *Рабочее время (Working time)* в разделе *Администрирование → Общие → Веб-интерфейс* (*Administration → General →GUI*);
- используйте его в поле *Когда активно (When active)* в *Пользователи → Пользователи* (*Users → Users*), вкладка пользователя *Оповещения (Media)*;
- используйте его для настройки более частого опроса элементов данных в течение рабочего времени:

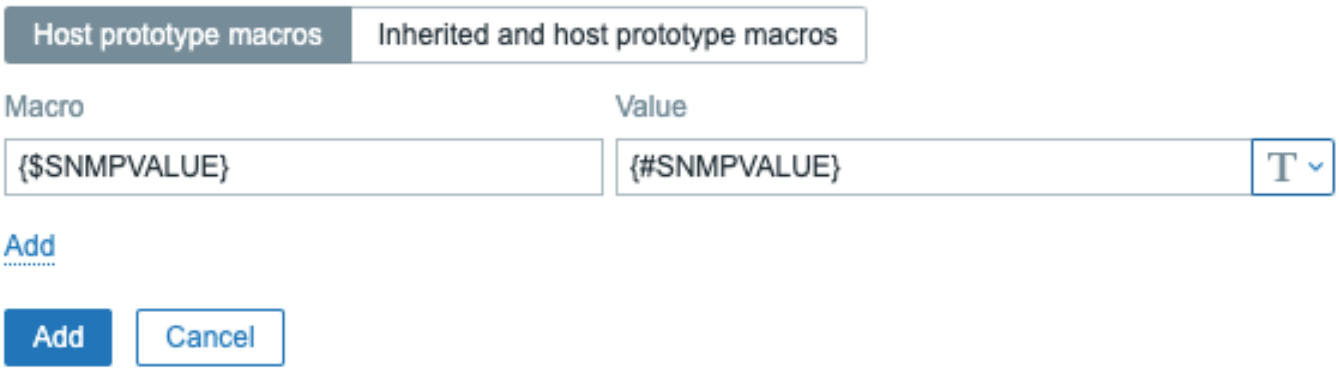



- используйте его в условии действия *Период времени (Time period)*;
- измените рабочее время в *Администрирование → Макросы* (*Administration → Macros*) при необходимости.

Пример 6

Используйте макрос прототипа узлов сети для настройки элементов данных у обнаруженных узлов сети:

- на уровне прототипа узлов сети задайте пользовательский макрос {$SNMPVALUE} с значением в виде макроса низкоуровневого обнаружения {#SNMPVALUE}:




- назначьте шаблон *Generic SNMPv2* на прототип узлов сети;
- используйте {$SNMPVALUE} в поле *SNMP OID* элементов данных шаблона *Generic SNMPv2*.

Контекст пользовательских макросов

Смотрите пользовательские макросы с контекстом.

### 3. Пользовательские макросы с контекстом

Обзор

В пользовательских макросах можно использовать опциональный контекст, который позволяет переопределять значение по умолчанию значением с учётом конкретной ситуации.

Контекст добавляется к имени макроса; синтаксис зависит от того, является ли контекст обычным текстовым значением:

```
{$    :"        "}
```

или регулярным выражением:

```
{$    :regex:"            "}
```

Обратите внимание, макрос с контекстом в виде регулярного выражения можно задавать только в настройках пользовательских макросов. Если префикс `regex:` используется в каком-либо другом месте как контекст пользовательского макроса, например в выражении триггера, этот префикс будет рассматриваться как обычный контекст.

Заключение в кавычки контекста опционально (смотрите также важные замечания).

Примеры контекста макросов:

| Пример | Описание |
|---|---|
| `{$LOW_SPACE_LIMIT}` | Пользовательский макрос без контекста. |
| `{$LOW_SPACE_LIMIT:/tmp}` | Пользовательский макрос с контекстом (статическая строка). |
| `{$LOW_SPACE_LIMIT:regex:"^/tmp$"}` | Пользовательский макрос с контекстом (регулярное выражение). То же, что и `{$LOW_SPACE_LIMIT:/tmp}`. |
| `{$LOW_SPACE_LIMIT:regex:"^/var/log/.*$"}` | Пользовательский макрос с контекстом (регулярное выражение). Соответствует всем строкам с префиксом /var/log/. |

Примеры использования

Пользовательские макросы можно задавать для получения более гибких порогов в выражениях триггеров (на основе значений, полученных низкоуровневым обнаружением). Например, вы можете задать следующие макросы:

- {$LOW_SPACE_LIMIT} = 10
- {$LOW_SPACE_LIMIT:/home} = 20
- {$LOW_SPACE_LIMIT:regex:”^/[a-z]+$”} = 30

Затем в прототипе триггеров для обнаружения примонтированных файловых систем можно использовать макрос низкоуровневого обнаружения как контекст ранее добавленного макроса:

`last(/host/vfs.fs.size[{#FSNAME},pfree])<{$LOW_SPACE_LIMIT:"{#FSNAME}"}`

После обнаружения к триггерам будут применены разные пороги срабатывания для значений свободного места, в зависимости от обнаруженных точек монтирования или типов файловых систем. События проблем будут сгенерированы, если:

- директория /home имеет менее 20% свободного места на диске
- директории, которые соответствуют шаблону регулярного выражения (такие как /etc, /tmp или /var), имеют менее 30% свободного места на диске
- директории, которые не соответствуют шаблону регулярного выражения и не /home, имеют менее 10% свободного места на диске

Важные замечания

- Если существует более одного пользовательского макроса с контекстом, Zabbix сначала попытается сопоставить макросы с простым контекстом и только затем макросы с контекстом в виде регулярного выражения в неопределённом порядке.

**Warning:**
Не создавайте разные макросы с контекстом, которые соответствуют одной строке, чтобы избежать неопределённого поведения.

- Если макрос со своим контекстом не найден на узле сети, присоединённых шаблонах или на глобальном уровне, тогда будет произведён поиск макроса без контекста.
- В контекстах макроса поддерживаются только макросы низкоуровневого обнаружения. Любые другие макросы игнорируются и обрабатываются как текстовые значения.

Технически, контекст макроса задаётся с использованием правил, похожих на параметры ключей элементов данных, за исключением того, что контекст макроса при наличии символа `,` не обрабатывается как несколько параметров:

- Контекст макроса необходимо заключать в кавычки ", если контекст содержит символ } или он начинается с символа ". Кавычки внутри заключённого в кавычки контекста необходимо экранировать при помощи символа \.
- Сам символ \ не экранируется, что означает, что невозможно задать заключённый в кавычки контекст, оканчивающийся на символ \ — макрос {$MACRO:"a:\b\c\"} ошибочный.
- Пробелы в начале контекста игнорируются, пробелы в конце не игнорируются:
  - Например, {$MACRO:A} тоже самое, что и {$MACRO: A}, но не то же самое, что {$MACRO:A }.
- Все пробелы до открывающих кавычек и после закрывающих кавычек игнорируются, но все пробелы внутри кавычек не игнорируются:
  - Макросы {$MACRO:"A"}, {$MACRO: "A"}, {$MACRO:"A" } и {$MACRO: "A" } одинаковы, но макросы {$MACRO:"A"} и {$MACRO:" A "} не одинаковы.

Следующие макросы идентичны, так как имеют один и тот же контекст: {$MACRO:A}, {$MACRO: A} и {$MACRO:"A"}. Такое поведение отлично от ключей элементов данных, где «key[a]», «key[ a]» и «key["a"]» одинаковы семантически, но различны для критерия уникальности.

#### 4. Скрытые пользовательские макросы

Обзор

Zabbix предоставляет два варианта защиты конфиденциальной информации в значениях пользовательских макросов:

- Скрытый текст
- Секрет хранилища

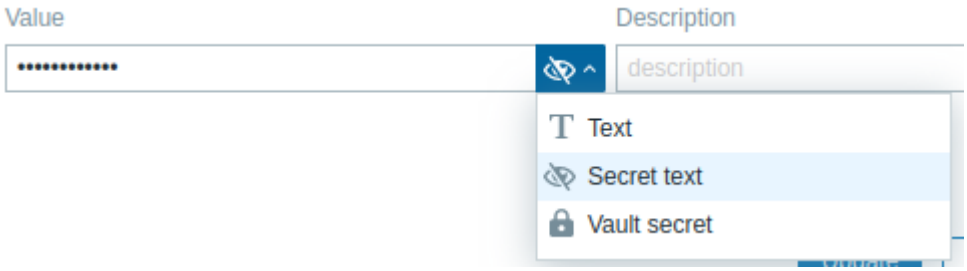
**Note:**
Хотя значение скрытого макроса и спрятано, его можно раскрыть посредством использования в элементах данных. Например, во внешнем скрипте оператор `echo`, ссылающийся на скрытый макрос, может использоваться для раскрытия значения макроса в веб-интерфейсе, поскольку сервер Zabbix имеет доступ к реальному значению макроса. Смотрите местоположения, где значения скрытых макросов не замаскированы.

Скрытые макросы нельзя использовать в выражениях триггеров.

Секретный текст

Для макросов Secret text значение макроса маскируется звездочками.

Чтобы сделать значение макроса секретным, нажмите кнопку в конце поля *Value* и выберите параметр *Secret text*:




После сохранения конфигурации просмотреть значение больше не будет возможно.

Чтобы изменить значение макроса, наведите курсор на поле *Value* и нажмите кнопку *Set new value* (появляется при наведении):

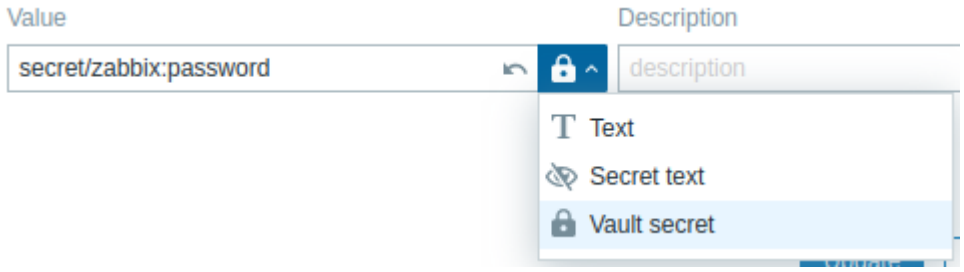
При нажатии кнопки *Set new value* (или при изменении типа значения макроса) текущее значение будет удалено. Вы можете восстановить исходное значение, нажав стрелку в конце поля *Value* (доступно только до сохранения новой конфигурации). Обратите внимание, что восстановление исходного значения не раскроет его.

**Note:**
URL-адреса, содержащие секретный макрос, не будут работать, так как макрос в них будет подставлен как "******".

Секрет Vault

С помощью макросов Vault secret значение макроса хранится во внешнем программном обеспечении для управления секретами (vault).

Чтобы настроить макрос Vault secret, нажмите кнопку в конце поля *Value* и выберите опцию *Vault secret*:




Значение макроса должно указывать на секрет vault. Формат ввода зависит от поставщика vault. Примеры настройки для конкретных поставщиков см. в разделах:

- HashiCorp
- CyberArk

Значения макросов секретов Vault извлекаются из хранилища Zabbix server (и Zabbix proxy, если для *Resolve secret vault macros by* установлено значение *Zabbix server and proxy*) при каждом обновлении данных конфигурации и затем сохраняются в кэше конфигурации.
Zabbix server и Zabbix proxy могут использовать разные хранилища.

Если для *Resolve secret vault macros by* установлено значение *Zabbix server*, то секреты Vault извлекаются только сервером, а Zabbix proxy получает значения макросов секретов Vault от Zabbix server при каждой синхронизации конфигурации и сохраняет их в собственном кэше конфигурации.
Это означает, что Zabbix proxy не может начать сбор данных после перезапуска, пока не получит обновление конфигурации от Zabbix server.

Чтобы вручную обновить значения секретов из хранилища, используйте опцию `secrets_reload` runtime control (только сервер).

Между Zabbix server и proxy должно быть включено шифрование; в противном случае в журнале будет записано предупреждающее сообщение сервера.

**Warning:**
Если значение макроса не удается успешно получить, соответствующий элемент данных, использующий это значение, перейдет в состояние unsupported.

Немаскированные расположения

Этот список содержит расположения параметров, в которых значения секретных макросов не маскируются.

**Note:**
Значения секретных макросов останутся замаскированными в указанных ниже расположениях, если на них ссылаться косвенно. Например, встроенные макросы {ITEM.KEY}, {ITEM.KEY<1-9>}, {LLDRULE.KEY} built-in macros, используемые в типах элементов данных (параметры Script или Webhook), будут разрешаться в ключи элементов данных, содержащие замаскированные секретные макросы, например `net.tcp.port[******,******]` вместо `net.tcp.port[192.0.2.1,80]`.

| Контекст | Параметр |
|---|---|
| **Элементы данных, прототипы элементов данных, правила LLD** | |
| Элемент данных | *Параметры ключа элемента данных* |
| Прототип элемента данных | *Параметры ключа прототипа элемента данных* |
| Правило низкоуровневого обнаружения | *Параметры ключа элемента данных обнаружения* |
| Агент SNMP | *SNMP community*<br>*Имя контекста* (SNMPv3)<br>*Имя безопасности* (SNMPv3)<br>*Фраза-пароль аутентификации* (SNMPv3)<br>*Фраза-пароль конфиденциальности* (SNMPv3) |
| Агент HTTP | *URL*<br>*Поля запроса*<br>*Тело запроса*<br>*Заголовки*<br>*Имя пользователя*<br>*Пароль*<br>*Пароль ключа SSL* |
| Скрипт | *Параметры*<br>*Скрипт* |
| Браузер | *Параметры*<br>*Скрипт* |
| Мониторинг базы данных | *SQL-запрос* |
| Агент TELNET | *Скрипт*<br>*Имя пользователя*<br>*Пароль* |
| Агент SSH | *Скрипт*<br>*Имя пользователя*<br>*Пароль* |
| Простая проверка | *Имя пользователя*<br>*Пароль* |
| Агент JMX | *Имя пользователя*<br>*Пароль* |
| **Предварительная обработка значений элементов данных** | |
| Шаг предварительной обработки JavaScript | *Скрипт* |

| Контекст | Параметр |
|---|---|
| **Веб-сценарии** | |
| Веб-сценарий | *Значение переменной*<br>*Значение заголовка*<br>*URL*<br>*Значение поля запроса*<br>*Значение поля POST*<br>*Сырые данные POST* |
| Аутентификация веб-сценария | *Пользователь*<br>*Пароль*<br>*Пароль ключа SSL* |
| **Коннекторы** | |
| Коннектор | *URL*<br>*Имя пользователя*<br>*Пароль*<br>*Токен*<br>*HTTP-прокси*<br>*Файл SSL-сертификата*<br>*Файл SSL-ключа*<br>*Пароль ключа SSL* |
| **Сетевое обнаружение** | |
| SNMP | *SNMP community*<br>*Имя контекста* (SNMPv3)<br>*Имя безопасности* (SNMPv3)<br>*Фраза-пароль аутентификации* (SNMPv3)<br>*Фраза-пароль конфиденциальности* (SNMPv3) |
| **Глобальные скрипты** | |
| Вебхук | *JavaScript script*<br>*Значение параметра JavaScript script* |
| Telnet | *Имя пользователя*<br>*Пароль* |
| SSH | *Имя пользователя*<br>*Пароль* |
| Скрипт | *Скрипт* |
| **Типы элементов данных** | |
| Скрипт | *Параметры скрипта* |
| Вебхук | *Параметры* |
| **Управление IPMI** | |
| Узел сети | *Имя пользователя*<br>*Пароль* |

### 5. Макросы низкоуровневого обнаружения

Обзор

Существует тип макроса, используемый в функции низкоуровневого обнаружения (LLD):

```
{#MACRO}
```

Это макрос, который используется в правиле LLD и возвращает реальные значения имени файловой системы, сетевого интерфейса, SNMP OID и т. д.

Эти макросы можно использовать для создания прототипов сущностей (элемент данных, триггер, график, правило LLD, прототип узла сети и прототипа группы узлов сети). Во время обнаружения реальных файловых систем, сетевых интерфейсов, виртуальных машин и т. д. эти макросы подставляются реальными значениями и служат основой для создания реальных сущностей.

Некоторые макросы низкоуровневого обнаружения поставляются "из коробки" вместе с функцией LLD в Zabbix - {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. Однако при создании пользовательского правила низкоуровневого обнаружения придерживаться этих имен не обязательно. В этом случае вы можете использовать любое другое имя макроса LLD и ссылаться на него.

Поддерживаемые типы данных

При определении пользовательских правил обнаружения значения свойств, возвращаемые в объектах JSON для макросов LLD, должны быть одним из следующих примитивных типов:

- string
- number
- boolean

Массивы, объекты и нулевые значения не поддерживаются. Любой макрос LLD, ссылающийся на такое значение, во время предварительной обработки и создания элемента данных останется нераскрытым и будет отображаться буквально (например, `'{#MY_MACRO}'`).

Поддерживаемые места

Макросы LLD можно использовать:

- в фильтре правила низкоуровневого обнаружения
- для прототипов элементов данных и прототипов обнаружения в
  - имени
  - параметрах ключа
  - единице измерения
  - интервале обновления[1]
  - тайм-ауте[1]
  - периоде хранения истории[1]
  - периоде хранения трендов[1]
  - шагах предварительной обработки значения элемента данных
  - SNMP OID
  - поле датчика IPMI
  - выражении вычисляемого/агрегированного элемента данных, в:
    - константах выражения и параметрах функций
    - параметрах ключа элемента данных
  - условиях фильтра агрегированного элемента данных (имя группы узлов сети и имя тега)
  - SSH-скрипте и Telnet-скрипте
  - SQL-запросе мониторинга базы данных
  - поле endpoint элемента JMX
  - описании
  - поле URL агента HTTP
  - поле HTTP-запроса агента HTTP
  - поле тела запроса агента HTTP
  - поле требуемых кодов состояния агента HTTP
  - ключе и значении полей заголовков агента HTTP
  - поле имени пользователя для HTTP-аутентификации агента HTTP
  - поле пароля для HTTP-аутентификации агента HTTP
  - поле HTTP-прокси агента HTTP
  - поле файла HTTP SSL-сертификата агента HTTP
  - поле файла HTTP SSL-ключа агента HTTP
  - поле пароля HTTP SSL-ключа агента HTTP
  - тегах
- для прототипов триггеров в
  - имени
  - оперативных данных
  - выражении (только в константах и параметрах функций)
  - URL
  - описании
  - тегах
- для прототипов графиков в
  - имени
- для прототипов узлов сети в
  - имени
  - видимом имени

- настраиваемых полях интерфейса: IP, DNS, порт, SNMP v1/v2 community, SNMP v3 context name, SNMP v3 security name, SNMP v3 authentication passphrase, SNMP v3 privacy passphrase
- имени прототипа группы узлов сети
- значении тега узла сети
- значении макроса узла сети
- (см. полный список)

Во всех этих местах, кроме фильтра правила низкоуровневого обнаружения, макросы LLD можно использовать внутри статического контекста макроса.

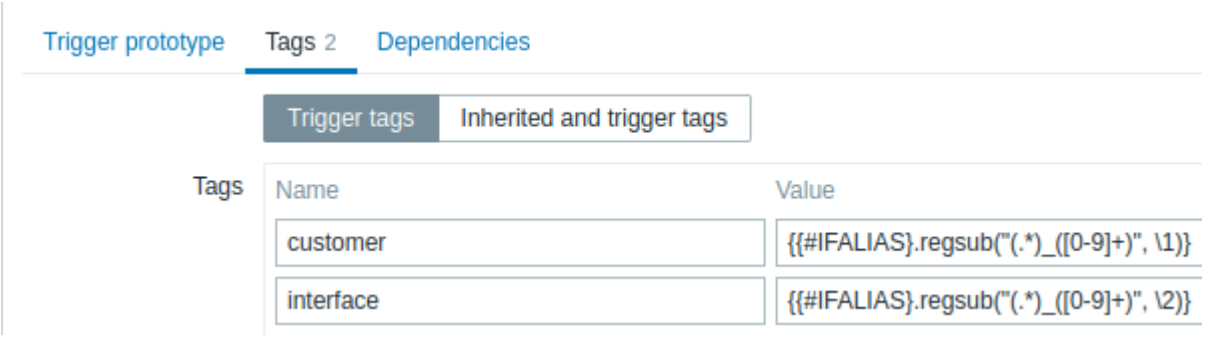
Использование функций макросов

Функции макросов поддерживаются с макросами низкоуровневого обнаружения (за исключением фильтра правила низкоуровневого обнаружения), позволяя извлекать нужную часть значения макроса с помощью регулярного выражения.

Например, вы можете захотеть извлечь имя клиента и номер интерфейса из следующего макроса LLD, чтобы назначить теги событий:

```
{#IFALIAS}=      _1
```

Чтобы это сделать, в поле значения тега события прототипа триггеров можно вместе с макросом использовать функцию макроса `regsub`:




Обратите внимание, что запятые не разрешены в не заключённых в кавычки параметрах ключей элементов данных, поэтому параметр, содержащий функцию макроса, необходимо заключить в кавычки. Символ обратной косой черты (\) следует использовать для экранирования двойных кавычек внутри параметра. Например:

```
net.if.in["{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}",bytes]
```

Для получения более подробных сведений о синтаксисе функций макросов смотрите: Функции макросов

Функции макросов в макросах низкоуровневого обнаружения поддерживаются, начиная с версии Zabbix 4.0.

Прототипы элементов данных без параметров ключа

Когда параметры ключа элемента данных не используются, поместите LLD-макрос внутрь квадратных скобок `[...]` параметров ключа элемента данных, чтобы он рассматривался в качестве параметра и подставлялся во время обнаружения:

```
v_[{#MACRO}]
```

Примечания

[1] В полях, отмеченных [1], один макрос должен заполнять целое поле. Несколько макросов в одном поле или макросы вперемешку с текстом не поддерживаются.

### 6. Макросы выражений

Обзор

Макросы выражений позволяют выполнять вычисления в полях.

Их значение вычисляется путем сначала разрешения всех вложенных макросов, а затем оценки полученного выражения.

Синтаксис:

```
{?EXPRESSION}
```

`EXPRESSION` использует тот же синтаксис и поддерживает те же функции, что и выражения триггеров.

Пример:

```
{?trendavg(/host/item1,1M:now/M)/trendavg(/host/item1,1M:now/M-1y)*100}
```

Для мест, где поддерживаются макросы выражений, найдите "{?EXPRESSION}" в таблице поддерживаемых макросов.

Примечания по использованию:

- Используйте макросы {FUNCTION.*} для ссылки на значения функций выражений триггеров/выражений восстановления.
- Используйте макросы {HOST.HOST<1-9>} и {ITEM.KEY<1-9>} для ссылки на узлы сети и элементы данных.
- В шаблонах используйте макросы {HOST.HOST<1-9>} или вообще опускайте узел сети для первого узла — например, {?avg(//item1,1h)} — вместо имен шаблонов, так как имена шаблонов не заменяются именами узлов сети во время связывания шаблонов.

```
{?{FUNCTION.VALUE2} - {FUNCTION.VALUE3}}
{?max(/{HOST.HOST}/{ITEM.KEY},3h)}
```

См. также примеры выражений триггеров для примера использования макроса выражения в имени события.

#### 12. Пользователи и группы пользователей

Обзор

Все пользователи в Zabbix работают с системой Zabbix через веб-интерфейс. Каждому пользователю в Zabbix присвоены уникальные имя пользователя и пароль.

Все пароли пользователей зашифрованы и хранятся в базе данных Zabbix. Пользователи Zabbix не могут использовать свои имя пользователя и пароль для авторизации непосредственно на сервере UNIX, если только они не были заведены соответственно также и в UNIX. Соединение между веб-сервером и браузером пользователя может быть защищено с помощью SSL.

При наличии гибкой схемы прав доступа пользователя вы можете ограничить и разграничить права доступа к:

- административным функциям веб-интерфейса Zabbix
- выполнению определённых действий в веб-интерфейсе
- доступу к наблюдаемым узлам сети в группах узлов сети
- использованию заданных API методов

##### 1. Настройка пользователя

Обзор

В начальной установке Zabbix есть два предопределенных пользователя:

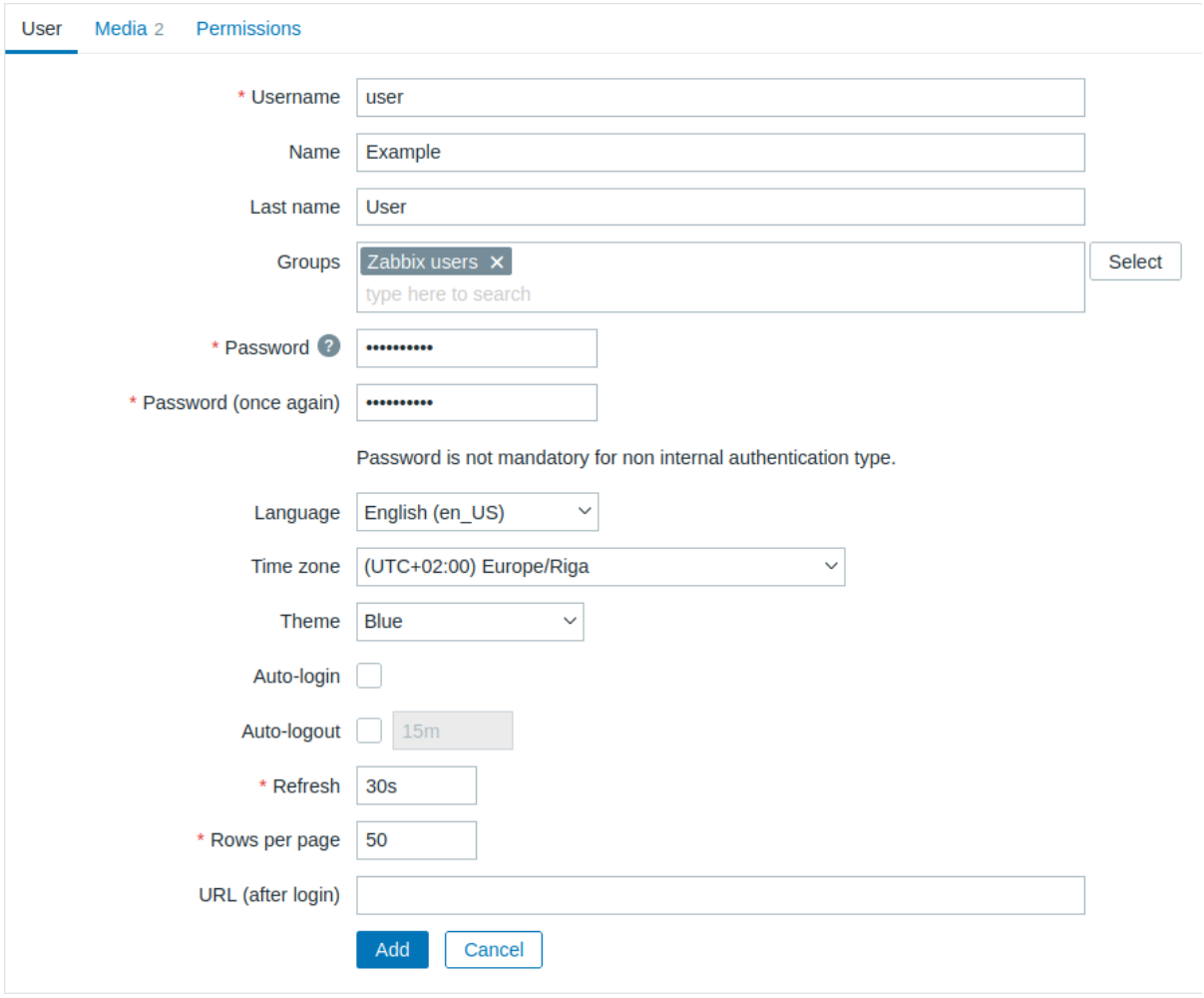
- *Admin* — суперпользователь Zabbix superuser с полными правами.
- *guest* — специальный Zabbix user. Пользователь 'guest' по умолчанию отключен. Если добавить его в группу пользователей Guests, можно войти под этим пользователем и получить доступ к страницам мониторинга в Zabbix. Обратите внимание, что по умолчанию у 'guest' нет прав на объекты Zabbix.

Чтобы настроить пользователя:

- Перейдите в *Users* → *Users*.
- Нажмите *Create user* (или на имя пользователя, чтобы изменить существующего пользователя).
- Отредактируйте атрибуты пользователя в форме.

Общие атрибуты

Вкладка *Пользователь* содержит общие атрибуты пользователя:



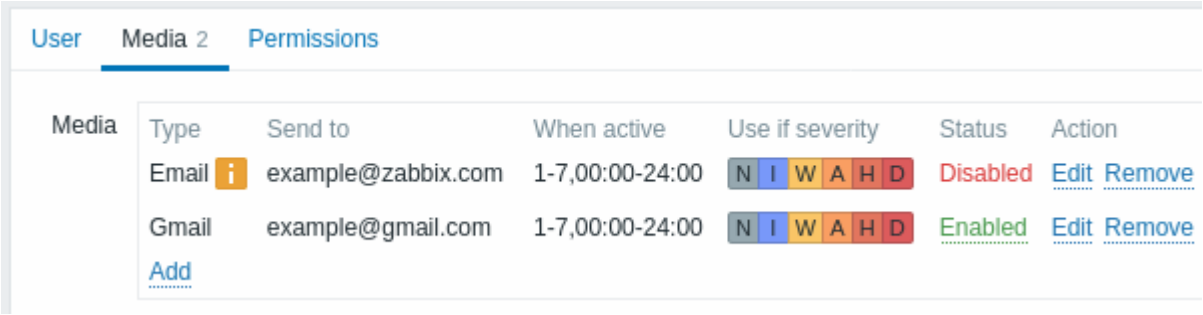

Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Username* | Уникальное имя пользователя, используемое в качестве имени для входа. |
| *Name* | Имя пользователя (необязательно).<br>Если не пусто, отображается в информации о подтверждении и в информации о получателе уведомления. |
| *Last name* | Фамилия пользователя (необязательно).<br>Если не пусто, отображается в информации о подтверждении и в информации о получателе уведомления. |
| *Groups* | Выберите группы пользователей, к которым принадлежит пользователь. Это поле поддерживает автодополнение, поэтому при вводе имени группы пользователей будет предложен выпадающий список подходящих групп. Прокрутите вниз, чтобы выбрать.<br>Также можно нажать *Select*, чтобы добавить группы. Нажмите 'x', чтобы удалить выбранное.<br>Принадлежность к группам пользователей определяет, к каким группам узлов сети и узлам сети пользователь будет иметь доступ. |
| *Password* | Два поля для ввода пароля пользователя или кнопка *Change password*, если пользователь уже существует.<br>При нажатии кнопки *Change password* открываются два поля для ввода нового пароля.<br>Для пользователя с ролью *Super admin role* при изменении собственного пароля нажатие кнопки *Change password* открывает дополнительное поле для ввода текущего (старого) пароля.<br>После успешного изменения пароля пользователь, для которого был изменен пароль, будет выведен из всех активных сеансов.<br>Обратите внимание, что пароль можно изменить только для пользователей, использующих внутреннюю аутентификацию Zabbix. |
| *Language* | Язык веб-интерфейса Zabbix.<br>Для работы переводов требуется расширение php gettext. |

| Parameter | Description |
|---|---|
| *Time zone* | Выберите часовой пояс, чтобы переопределить глобальный часовой пояс на уровне пользователя, или выберите **System default**, чтобы использовать глобальные настройки часового пояса. |
| *Theme* | Определяет внешний вид веб-интерфейса:<br>**System default** - использовать системные настройки по умолчанию<br>**Blue** - стандартная синяя тема<br>**Dark** - альтернативная темная тема<br>**High-contrast light** - светлая тема с высокой контрастностью<br>**High-contrast dark** - темная тема с высокой контрастностью |
| *Auto-login* | Отметьте этот флажок, чтобы Zabbix запоминал пользователя и автоматически выполнял вход в течение 30 дней.<br>При входе с *Remember for 30 days*:<br>- Auto-logout сбрасывается (сеанс сохраняется на 30 дней).<br>- Auto-login включается для бесшовной повторной аутентификации.<br>При входе без *Remember for 30 days*:<br>- Auto-login включается для бесшовной повторной аутентификации.<br>- Auto-logout остается под управлением стандартной настройки тайм-аута.<br>Для этого используются файлы cookie браузера. |
| *Auto-logout* | Если этот флажок установлен, пользователь будет автоматически выведен из системы по истечении заданного количества секунд (минимум 90 секунд, максимум 1 день).<br>Обратите внимание, что эта настройка переопределяется, если включен параметр *Remember for 30 days*, так как сеанс продлевается на весь период.<br>Поддерживаются суффиксы времени, например 90s, 5m, 2h, 1d.<br>Обратите внимание, что эта опция не будет работать:<br>* Если включена глобальная настройка ”Show warning if Zabbix server is down” и веб-интерфейс Zabbix остается открытым.<br>* Когда страницы меню Monitoring выполняют фоновое обновление информации.<br>* Если выполнен вход с отмеченной опцией *Remember me for 30 days*. |
| *Refresh* | Задайте частоту обновления, используемую для графиков, текстовых данных и т. д.<br>Можно установить 0, чтобы отключить.<br>Поддерживаются суффиксы времени, например 90s, 5m, 1h. |
| *Rows per page* | Вы можете определить, сколько строк на странице будет отображаться в списках. |
| *URL (after login)* | Можно настроить перенаправление пользователя на определенный URL после успешного входа, например на страницу *Problems*. |

Пользовательские медиа

Вкладка *Media* содержит список всех медиа, определенных для пользователя.
Медиа используются для отправки уведомлений.




Нажмите *Add*, чтобы назначить медиа пользователю.

Если тип медиа был отключен:

- После имени отображается желтый значок информации.
- В столбце Status отображается *Disabled*.

**Note:**
Права пользователя на изменение сведений о своих медиа могут быть предоставлены или отозваны на основе его роли пользователя (см. параметр *Create and edit own media*). Права пользователя с ролью Super admin на изменение сведений о медиа других пользователей также могут быть предоставлены или отозваны на основе его роли пользователя (см. параметр *Create and edit user media*)

См. раздел Типы медиа для получения подробной информации о настройке пользовательских медиа.

Права доступа

Вкладка *Права доступа (Permissions)* содержит информацию о следующих элементах:

- Роль пользователя (обязательна для любого вновь созданного пользователя), которую может изменить только пользователь с типом *Супер-администратор*.

**Attention:**
Пользователи не могут быть созданы без роли пользователя (за исключением использования Zabbix User API). Ранее созданных пользователей, у которых нет роли, можно редактировать без назначения им роли. Однако после назначения роли её можно только изменить, но не удалить. <br><br> Обратите внимание, что пользователи без роли могут входить в Zabbix только с использованием аутентификации LDAP или SAML, при условии, что их информация LDAP/SAML соответствует сопоставлениям групп пользователей, настроенным в Zabbix.

- Тип пользователя (*Пользователь (User)*, *Администратор (Admin)*, *Супер-администратор (Super admin)*), который задаётся в настройках роли.
- Группы узлов сети и шаблонов, к которым пользователь имеет доступ.
- Пользователи с типами *Пользователь* и *Администратор* по умолчанию не имеют прав доступа к каким-либо группам, шаблонам и узлам сети. Для получения прав доступа им необходимо находиться в группах пользователей, имеющих доступ к соответствующим объектам.
- Права доступа к разделам и элементам веб-интерфейса Zabbix, модулям и методам API.
- Элементы с разрешённым доступом отображаются зелёным цветом, а с запрещённым доступом — светло-серым цветом.
- Права на выполнение конкретных действий.
- Действия, которые пользователю разрешено выполнять, отображаются зелёным цветом, а запрещённые — светло-серым цветом.

Смотрите страницу Права доступа для получения подробных сведений.

### 2. Права доступа

Обзор

Права доступа пользователей в Zabbix зависят от типа пользователя, настраиваемых ролей пользователя и доступа к узлам сети, который задается для группы пользователей.

Типы пользователей

Права доступа в Zabbix зависят в первую очередь от типа пользователя:

- *Пользователь (User)* — имеет ограниченный доступ к разделам меню (см. ниже) и по умолчанию не имеет доступа к ресурсам. Любые права доступа к группам узлов сети или шаблонов должны быть назначены явно;
- *Администратор (Admin)* — имеет неполный доступ к разделам меню (см. ниже). По умолчанию не имеет доступа к группам узлов сети. Любые права доступа к группам узлов сети или шаблонов должны быть назначены явно;
- *Супер-администратор (Super admin)* — имеет доступ ко всем разделам меню. Этот пользователь имеет доступ на чтение/запись ко всем группам узлов сети и шаблонов. Права доступа невозможно отозвать путём запрета доступа к отдельным группам.

**Доступ к меню**

**Note:**
Ограниченный доступ к некоторым элементам пользовательского интерфейса запрещает открытие только этой страницы — он не убирает возможность доступа к соответствующим данным в других частях интерфейса.

В таблице ниже показан доступ к разделам меню Zabbix по типам пользователей:

| Раздел меню | | Пользователь | Администратор | Супер-администратор |
|---|---|---|---|---|
| **Панели** | | + | + | + |
| **Мониторинг** | | + | + | + |
| | *Проблемы* | + | + | + |
| | *Узлы сети* | + | + | + |
| | *Последние данные* | + | + | + |
| | *Карты сетей* | + | + | + |
| | *Обнаружение* | | + | + |
| **Услуги** | | + | + | + |
| | *Услуги* | + | + | + |
| | *SLA* | | + | + |
| | *Отчёт SLA* | + | + | + |
| **Инвентаризация** | | + | + | + |
| | *Обзор* | + | + | + |
| | *Узлы сети* | + | + | + |
| **Отчёты** | | + | + | + |
| | *Информация о системе* | | | + |
| | *Регулярные отчёты* | | + | + |
| | *Отчёт о доступности* | + | + | + |
| | *Топ 100 триггеров* | + | + | + |
| | *Журнал аудита* | | | + |
| | *Журнал действий* | | | + |
| | *Оповещения* | | + | + |
| **Сбор данных** | | | + | + |
| | *Группы шаблонов* | | + | + |
| | *Группы узлов сети* | | + | + |
| | *Шаблоны* | | + | + |
| | *Узлы сети* | | + | + |
| | *Обслуживание* | | + | + |
| | *Корреляция событий* | | | + |
| | *Обнаружение* | | + | + |
| **Оповещения** | | | + | + |
| | *Действия триггеров* | | + | + |
| | *Действия услуг* | | + | + |
| | *Действия обнаружения* | | + | + |
| | *Действия авторегистрации* | | + | + |
| | *Внутренние действия* | | + | + |
| | *Способы оповещений* | | | + |
| | *Скрипты* | | | + |
| **Пользователи** | | | | + |
| | *Группы пользователей* | | | + |
| | *Роли пользователей* | | | + |
| | *Пользователи* | | | + |
| | *API токены* | | | + |
| | *Аутентификация* | | | + |
| **Администрирование** | | | | + |
| | *Общие* | | | + |
| | *Журнал аудита* | | | + |
| | *Очистка истории* | | | + |
| | *Прокси* | | | + |
| | *Макросы* | | | + |
| | *Очередь* | | | + |

Роли пользователей

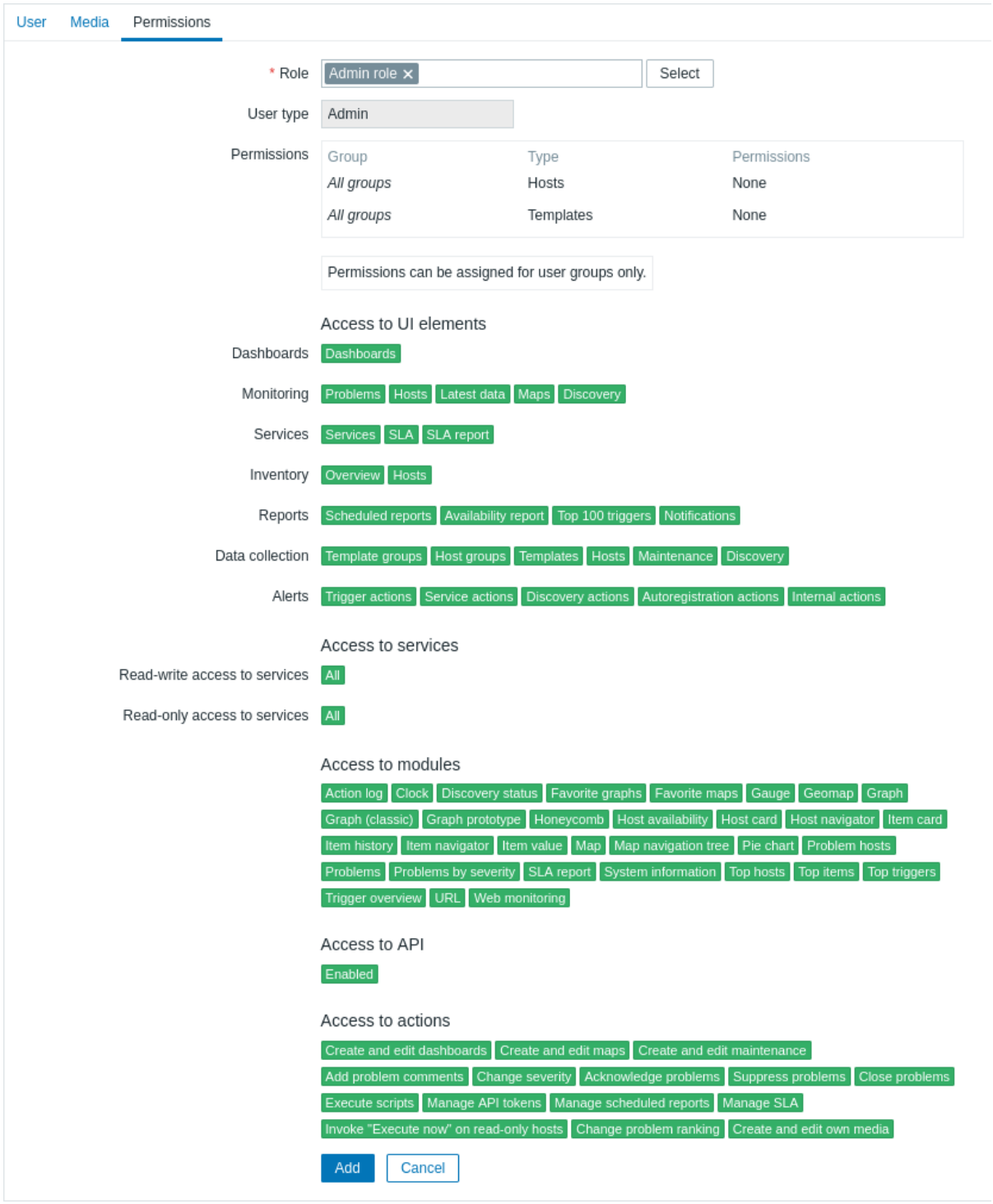
Роли пользователей позволяют по необходимости изменять права доступа, определённые типом пользователя.

Хотя права доступа нельзя добавить (сверх установленных для типа пользователя), но можно отозвать какие-то из них.

Кроме того, роль пользователя определяет доступ не только к разделам меню, но и к услугам, модулям, методам API и различным действиям в веб-интерфейсе.

Пользователи, являющиеся Супер-администраторами, могут настраивать Роли пользователей в разделе меню *Пользователи → Роли пользователей (Users → User roles)*.

Назначить роль пользователя может Супер-администратор во вкладке *Права доступа (Permissions)* в диалоге настроек пользователя.




Доступ к узлам сети

Доступ к любым данным узлов сети и шаблонов в Zabbix назначается группам пользователей только на уровне групп узлов сети/шаблонов.

Это означает, что отдельному пользователю нельзя напрямую предоставить разрешение на доступ к какому-либо узлу сети (или к группе узлов сети). Пользователь может получить доступ к узлу сети, только будучи частью группы пользователей, которая имеет доступ к группе узлов сети, содержащей необходимый узел сети.

Таким же образом пользователю можно предоставить доступ к шаблону, только включив его в группу пользователей, имеющую доступ к группе шаблонов, в которую входит соответствующий шаблон.

### 3. Группы пользователей

Обзор

Группы пользователей позволяют группировать пользователей для организационных целей и для назначения прав доступа к данным. Права доступа к данным просмотра и настроек групп узлов сети и групп шаблонов назначаются на группы пользователей, а не индивидуально пользователям.

Часто имеет смысл разделить, какая информация доступна одной группе пользователей и какая — другой. Это может быть достигнуто группировкой пользователей и последующим назначением различных прав доступа к группам узлов сети и шаблонов.

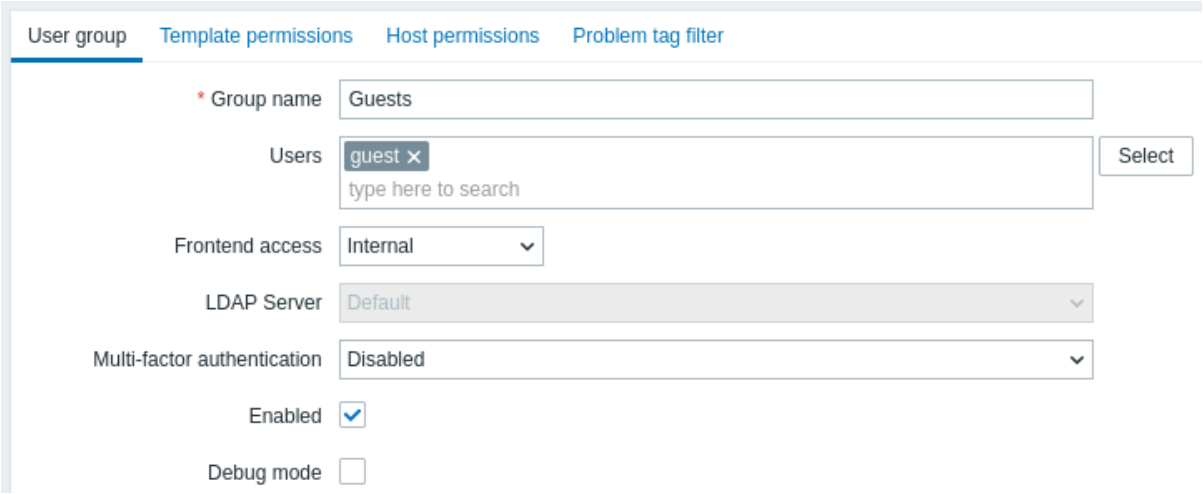
Пользователь может входить в любое количество групп.

Конфигурация

Чтобы настроить группу пользователей:

- Перейдите в *Пользователи → Группы пользователей*
- Нажмите *Создать группу пользователей* (или на имя группы, чтобы изменить существующую группу)
- Отредактируйте атрибуты группы в форме

Вкладка **Группа пользователей** содержит общие атрибуты группы:

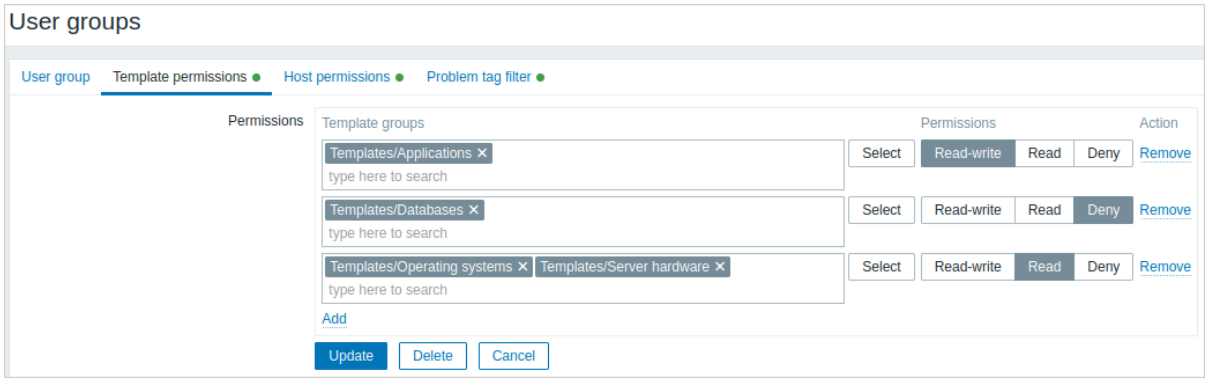



Все обязательные поля ввода отмечены красной звездочкой.

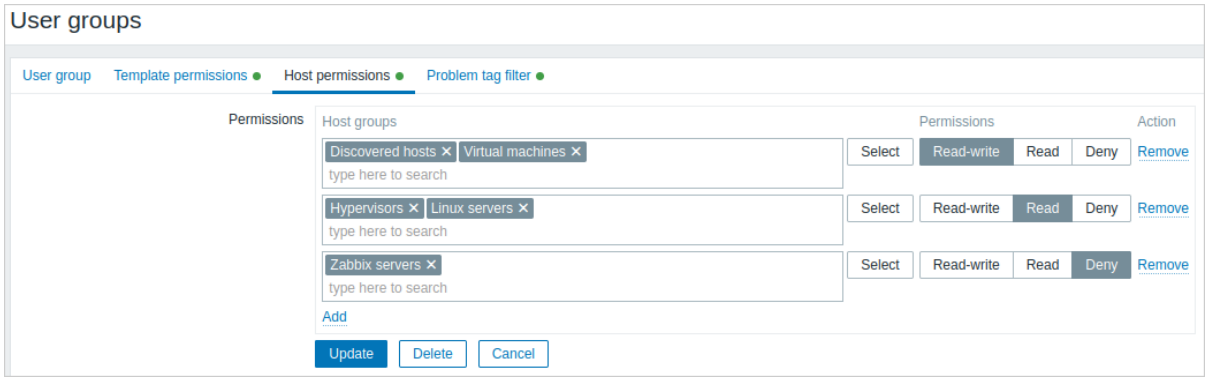
| Parameter | Description |
|---|---|
| *Group name* | Уникальное имя группы. |
| *Users* | Чтобы добавить пользователей в группу, начните вводить имя существующего пользователя. Когда появится выпадающий список с подходящими именами пользователей, прокрутите вниз и выберите нужного пользователя.<br>Либо нажмите кнопку *Select*, чтобы выбрать пользователей во всплывающем окне. |
| *Frontend access* | Способ аутентификации пользователей группы.<br>**System default** - использовать метод аутентификации по умолчанию (заданный глобально)<br>**Internal** - использовать внутреннюю аутентификацию Zabbix (даже если глобально используется аутентификация LDAP).<br>Игнорируется, если глобальным значением по умолчанию является HTTP-аутентификация.<br>**LDAP** - использовать аутентификацию LDAP (даже если глобально используется внутренняя аутентификация).<br>Игнорируется, если глобальным значением по умолчанию является HTTP-аутентификация.<br>**Disabled** - доступ к веб-интерфейсу Zabbix для этой группы запрещен |
| *LDAP server* | Выберите, какой LDAP server использовать для аутентификации пользователя.<br>Это поле доступно только если для *Frontend access* установлено значение LDAP или System default. |

| Parameter | Description |
|---|---|
| *Multi-factor authentication* | Выберите, какой method многофакторной аутентификации использовать для аутентификации пользователя:<br>**Default** - использовать метод, заданный как значение по умолчанию в конфигурации MFA; этот вариант выбран по умолчанию для новых групп пользователей, если MFA включена;<br>**<Method name>** - использовать выбранный метод (например, ”Zabbix TOTP”);<br>**Disabled** - MFA отключена для этой группы; этот вариант выбран по умолчанию для новых групп пользователей, если MFA отключена.<br>Обратите внимание: если пользователь принадлежит к нескольким группам пользователей с включенной MFA (или хотя бы в одной группе MFA включена), применяются следующие правила аутентификации: если в какой-либо группе используется метод MFA ”Default”, он будет использоваться для аутентификации пользователя; в противном случае для аутентификации будет использован первый метод (в алфавитном порядке). |
| *Enabled* | Состояние группы пользователей и ее участников.<br>*Checked* - группа пользователей и пользователи включены<br>*Unchecked* - группа пользователей и пользователи отключены |
| *Debug mode* | Отметьте этот флажок, чтобы активировать debug mode для пользователей. |

Вкладка **Права на шаблоны** позволяет задать доступ группы пользователей к данным группы шаблонов (а следовательно, и шаблонов):




Вкладка **Права на узлы сети** позволяет задать доступ группы пользователей к данным группы узлов сети (а следовательно, и узлов сети):




Нажмите Add, чтобы выбрать группы шаблонов/узлов сети (родительскую или вложенную группу) и назначить им права. Начните вводить имя группы (появится выпадающий список подходящих групп) или нажмите *Select*, чтобы открыть всплывающее окно со списком всех групп.

Затем используйте кнопки выбора, чтобы назначить права выбранным группам. Возможны следующие права:

- **Read-write** - доступ на чтение и запись к группе;
- **Read** - доступ только на чтение к группе;
- **Deny** - доступ к группе запрещен.

Если одна и та же группа шаблонов/узлов сети добавлена в несколько строк с разными правами, будет применено наиболее строгое право.

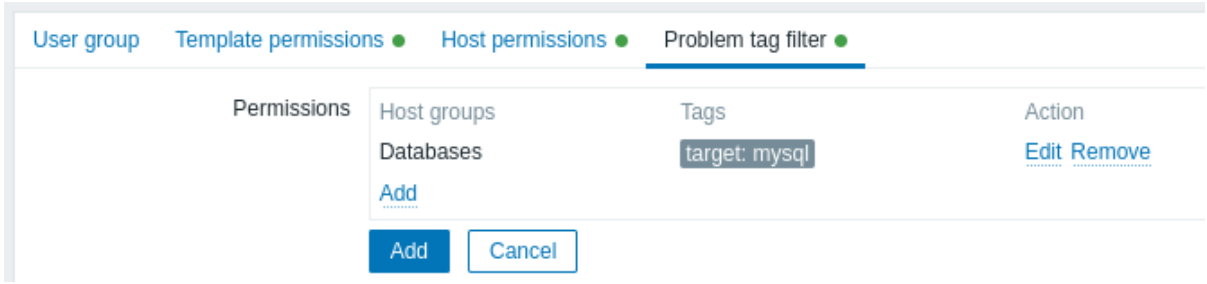
Обратите внимание: пользователь *Super admin* может принудительно задать для вложенных групп тот же уровень прав, что и для родительской группы; это можно сделать в форме настройки группы узлов сети/шаблонов.

Вкладки **Права на шаблоны** и **Права на узлы сети** поддерживают одинаковый набор параметров.

Текущие права для групп отображаются в блоке *Permissions*, где их можно изменить или удалить.

**Note:**
Если у группы пользователей есть права **Read-write** на узел сети и **Deny** или отсутствуют права на шаблон, связанный с этим узлом сети, пользователи такой группы не смогут редактировать элементы данных, унаследованные от шаблона, на узле сети, а имя шаблона будет отображаться как *Inaccessible template*.

Вкладка **Фильтр тегов проблем** позволяет задать права на основе тегов для групп пользователей, чтобы они видели проблемы, отфильтрованные по имени и значению тега:

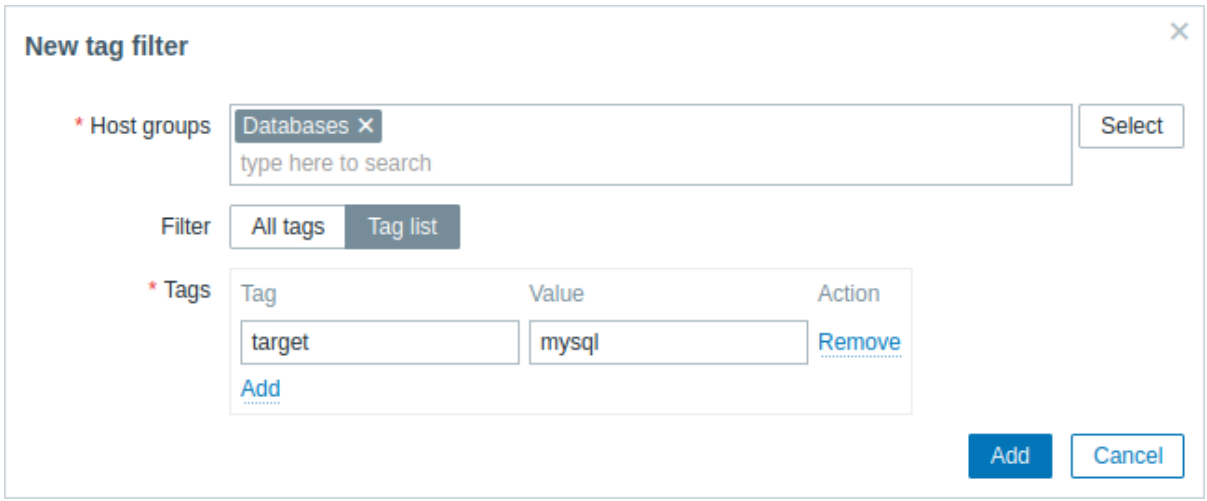



Нажмите Add, чтобы выбрать группы узлов сети. Чтобы выбрать группу узлов сети, к которой будет применен фильтр тегов, нажмите *Select*, чтобы получить полный список существующих групп узлов сети, или начните вводить имя группы узлов сети, чтобы появился выпадающий список подходящих групп. Будут отображаться только группы узлов сети, поскольку фильтр тегов проблем не может быть применен к группам шаблонов.

Затем можно переключиться с *All tags* на *Tag list*, чтобы задать конкретные теги и их значения для фильтрации. Можно добавлять имена тегов без значений, но нельзя добавлять значения без имен. Только первые три тега (со значениями, если они есть) отображаются в блоке *Permissions*; если их больше, их можно увидеть, нажав или наведя курсор на значок •••.

Фильтр тегов позволяет разделить доступ к группе узлов сети и возможность видеть проблемы.

Например, если администратору базы данных нужно видеть только проблемы базы данных ”MySQL”, сначала необходимо создать группу пользователей для администраторов баз данных, а затем указать имя тега ”target” и значение ”mysql”.




Если указано имя тега ”target”, а поле значения оставлено пустым, группа пользователей будет видеть все проблемы с именем тега ”target” для выбранной группы узлов сети. Если выбрано *All tags*, группа пользователей будет видеть все проблемы для указанной группы узлов сети.

Убедитесь, что имя тега и значение тега указаны правильно, иначе группа пользователей не увидит никаких проблем.

Рассмотрим пример, когда пользователь является участником нескольких выбранных групп пользователей. В этом случае фильтрация будет использовать условие OR для тегов.

| **User group A** | **User group B** | **Visible result for a user (member) of both groups** |
|---|---|---|
| *Tag filter* | | |

| *Host group* | *Tag name* | *Tag value* | *Host group* | *Tag name* | *Tag value* | |
|---|---|---|---|---|---|---|
| Databases | target | mysql | Databases | target | oracle | Проблемы target:mysql или target:oracle видны |
| Databases | set to: *All tags* | | Databases | target | oracle | Видны все проблемы |
| Not configured in the **Problem tag filter** | | | Databases | target | oracle | Видны проблемы target:oracle |

**Attention:**
Добавление фильтра (например, всех тегов в определенной группе узлов сети ”Databases”) приводит к тому, что проблемы других групп узлов сети становятся недоступны для просмотра.

Доступ из нескольких групп пользователей

Пользователь может принадлежать к любому числу групп пользователей. Эти группы могут иметь разные права доступа к узлам сети или шаблонам.

Поэтому важно понимать, к каким объектам в результате сможет получить доступ пользователь без привилегий. В следующем примере рассмотрим, как доступ к узлу сети **X** (в группе узлов сети 1) будет изменяться в различных ситуациях для пользователя, который состоит в группах пользователей A и B.

- Если у группы A есть только доступ *Read* к группе узлов сети 1, а у группы B доступ *Read-write* к группе узлов сети 1, пользователь получит доступ **Read-write** к ’X’.

**Attention:**
Права *Read-write* имеют приоритет над правами *Read*.

- В той же ситуации, что и выше, если ’X’ одновременно также входит в группу узлов сети 2, в которой доступ для группы A или B **запрещен**, доступ к ’X’ будет **недоступен**, несмотря на доступ *Read-write* к группе узлов сети 1.
- Если у группы A не определены права доступа, а у группы B есть доступ *Read-write* к группе узлов сети 1, пользователь получит доступ *Read-write* к ’X’.
- Если у группы A есть доступ *Deny* к группе узлов сети 1, а у группы B есть доступ *Read-write* к группе узлов сети 1, пользователю будет **запрещен** доступ к ’X’.

Другая информация

- Пользователь с уровнем Администратор с правами доступа *Чтение-запись* к узлу сети не сможет присоединять/отсоединять шаблоны, если у него нет прав доступа к группе шаблонов, к которой они относятся. При наличии прав доступа на *Чтение* к группе шаблонов этот пользователь сможет присоединять/отсоединять шаблоны от этого узла сети, однако, он не будет видеть шаблоны в списке шаблонов и не сможет оперировать шаблонами в других местах.
- Пользователь с уровнем Администратор с правами доступа на *Чтение* к узлу сети не увидит этот узел сети в списке узлов сети раздела настройки; однако, триггеры этого узла сети будут доступны при настройке Услуг IT.
- Любой не Zabbix Супер-администратор пользователь (включая «guest») может просматривать карты сети, пока карта пустая или имеет только изображения. Права соблюдаются при наличии на карте добавленных узлов сети, групп узлов сети или триггеров.
- Zabbix сервер не будет отправлять оповещения пользователям, которые указаны получателями в операции действия, если доступ к соответствующему узлу сети явным образом «запрещён».

### 13. Хранение секретов

Обзор

Zabbix можно настроить для получения конфиденциальной информации из защищенного хранилища. Поддерживаются следующие службы управления секретами: HashiCorp Vault KV Secrets Engine - Version 2, CyberArk Vault CV12.

Секреты можно использовать для получения:

- значений пользовательских макросов
- учетных данных для доступа к базе данных

Zabbix предоставляет доступ только для чтения к секретам в хранилище, предполагая, что управление секретами осуществляется кем-то другим.

Информацию о настройках конкретного поставщика хранилища смотрите в разделах:

- Настройка HashiCorp
- Настройка CyberArk

Кэширование секретных значений

По умолчанию значения макросов секретов vault извлекаются сервером Zabbix при каждом обновлении данных конфигурации и затем сохраняются в кэше конфигурации. Прокси Zabbix получает значения макросов секретов vault от сервера Zabbix при каждой синхронизации конфигурации и сохраняет их в собственном кэше конфигурации.

**Attention:**
Между сервером Zabbix и прокси должно быть включено шифрование; в противном случае в журнале будет записано предупреждающее сообщение сервера.

Также можно настроить независимое получение значений макросов сервером Zabbix и прокси Zabbix.

Чтобы вручную запустить обновление кэшированных секретных значений из vault, используйте командную опцию `secrets_reload`.

Для учетных данных базы данных веб-интерфейса Zabbix кэширование по умолчанию отключено, но его можно включить, установив в zabbix.conf.php параметр `$DB['VAULT_CACHE'] = true`. Учетные данные будут храниться в локальном кэше с использованием временного каталога файловой системы. Веб-сервер должен разрешать запись в частную временную папку (например, для Apache необходимо задать параметр конфигурации `PrivateTmp=True`). Чтобы управлять тем, как часто кэш данных обновляется или становится недействительным, используйте константу ZBX_DATA_CACHE_TTL constant
.

Конфигурация TLS

Чтобы настроить TLS для связи между компонентами Zabbix и хранилищем, добавьте сертификат, подписанный центром сертификации (CA), в системное хранилище CA по умолчанию.
Чтобы использовать другое расположение, укажите каталог в параметре конфигурации Zabbix SSLCALocation сервер/прокси, поместите файл сертификата в этот каталог, затем выполните CLI команду:

```
c_rehash .
```

#### 1. Настройка CyberArk

В этом разделе объясняется, как настроить Zabbix для извлечения секретов из CyberArk Vault CV12.

Хранилище должно быть развёрнуто и настроено, как описано в официальной документации CyberArk [en].

Чтобы узнать о настройке TLS в Zabbix, смотрите *Хранение секретов*.

Учетные данные базы данных

Доступ к секрету с учетными данными базы данных настраивается отдельно для каждого компонента Zabbix.

Сервер и прокси

Чтобы получить учетные данные базы данных из хранилища для Zabbix server или proxy, укажите следующие параметры конфигурации в файле конфигурации:

- `Vault` - какой поставщик хранилища следует использовать;
- `VaultURL` - HTTP[S]-URL сервера хранилища;
- `VaultDBPath` - запрос к секрету хранилища, содержащему учетные данные базы данных, которые будут получены по ключам "Content" и "UserName" (этот параметр можно использовать только если не указаны DBUser и DBPassword);
- `VaultTLSCertFile`, `VaultTLSKeyFile` - имена файлов SSL-сертификата и ключа; настройка этих параметров не обязательна, но настоятельно рекомендуется;
- `VaultPrefix` - пользовательский префикс для пути или запроса к хранилищу, в зависимости от хранилища; если не указан, будет использован наиболее подходящий вариант по умолчанию.

**Attention:**
Параметры конфигурации `Vault`, `VaultURL`, `VaultTLSCertFile`, `VaultTLSKeyFile` и `VaultPrefix` также используются для аутентификации в хранилище при обработке макросов секретов хранилища сервером Zabbix (и прокси Zabbix, если настроен). Сервер Zabbix и прокси не будут открывать макросы секретов хранилища, которые содержат учетные данные базы данных из VaultDBPath.

Сервер Zabbix и прокси Zabbix считывают связанные с хранилищем параметры конфигурации из файлов *zabbix_server.conf* и *zabbix_proxy.conf* при запуске.

Пример

1. В *zabbix_server.conf* укажите следующие параметры:

```
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 отправит следующий запрос API в хранилище:

```
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. Ответ хранилища будет содержать ключи «Content» и «UserName»:

```
{
    "Content": <    >,
    "UserName": <  _        >,
    "Address": <   >,
    "Database": <   _    >,
    "PasswordChangeInProcess":<            >
}
```

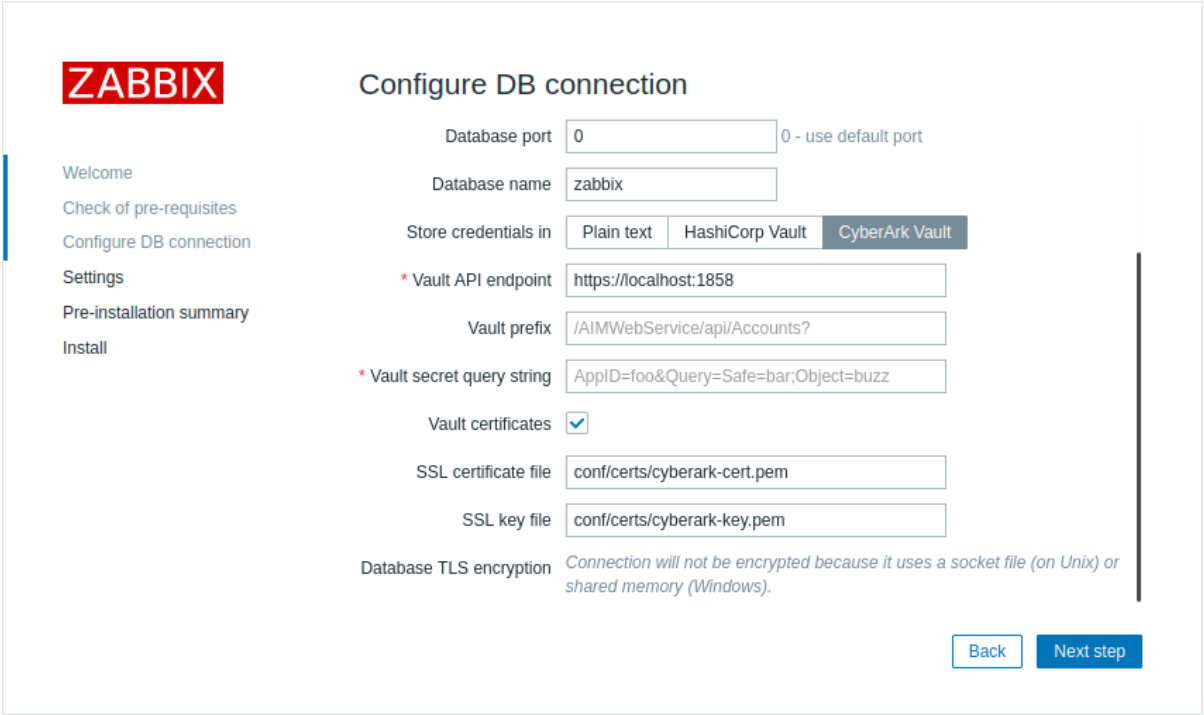
4. В результате Zabbix будет использовать следующие учётные данные для аутентификации в базе данных:

- Имя пользователя: <имя_пользователя>
- Пароль: <пароль>

Веб-интерфейс

Чтобы получить учётные данные базы данных из хранилища для веб-интерфейса Zabbix, укажите следующие параметры во время установки веб-интерфейса.

1. На шаге *Настройка соединения с базой данных (Configure DB Connection)* установите параметр *Хранить учётные данные в (Store credentials in)* в значение «CyberArk Vault».



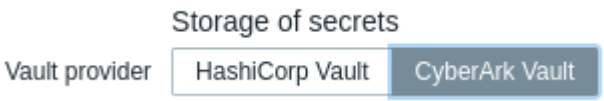

2. Затем заполните дополнительные параметры:

| Параметр | Обязательный? | Значение по умолчанию | Описание |
|---|---|---|---|
| API endpoint Хранилища (Vault API endpoint) | да | https://localhost:1858 | Укажите URL для подключения к хранилищу в формате `://:` |
| Префикс Хранилища (Vault prefix) | нет | /AIMWebService/api/Accounts? | Укажите пользовательский префикс для пути или запроса хранилища. Если не указано, используется значение по умолчанию. |
| Строка запроса к секрету Хранилища (Vault secret query string) | да | | Запрос, который указывает, откуда следует извлекать учётные данные базы данных. Пример: `AppID=foo&Query=Safe=bar;Object=buzz` |
| Сертификаты хранилища (Vault certificates) | нет | | После установки флажка появятся дополнительные параметры, позволяющие настроить аутентификацию клиента. Хотя этот параметр является необязательным, настоятельно рекомендуется включить его для взаимодействия с хранилищем CyberArk. |
| Файл сертификата SSL (SSL certificate file) | нет | conf/certs/cyberark-cert.pem | Путь к файлу сертификата SSL. Файл должен быть в формате PEM. Если файл сертификата также содержит закрытый ключ, оставьте параметр файла ключа SSL пустым. |
| Файл ключа SSL (SSL key file) | нет | conf/certs/cyberark-key.pem | Имя файла закрытого ключа SSL, используемого для аутентификации клиента. Файл должен быть в формате PEM. |

Значения пользовательских макросов

Чтобы использовать CyberArk Vault для хранения значений пользовательских макросов *Vault secret*, убедитесь, что:

- сервер Zabbix настроен для работы с CyberArk Vault;
- параметр *Vault provider* в *Administration → General → Other* установлен в "CyberArk Vault".



**Note:**
Сервер Zabbix (и прокси Zabbix, если настроен) требует доступа к значениям макросов *Vault secret* из хранилища. Веб-интерфейс Zabbix не нуждается в таком доступе.

Значение макроса должно содержать запрос (в формате `query:key`).

Подробную информацию об обработке значений макросов Zabbix см. в разделе *Vault secret macros*.

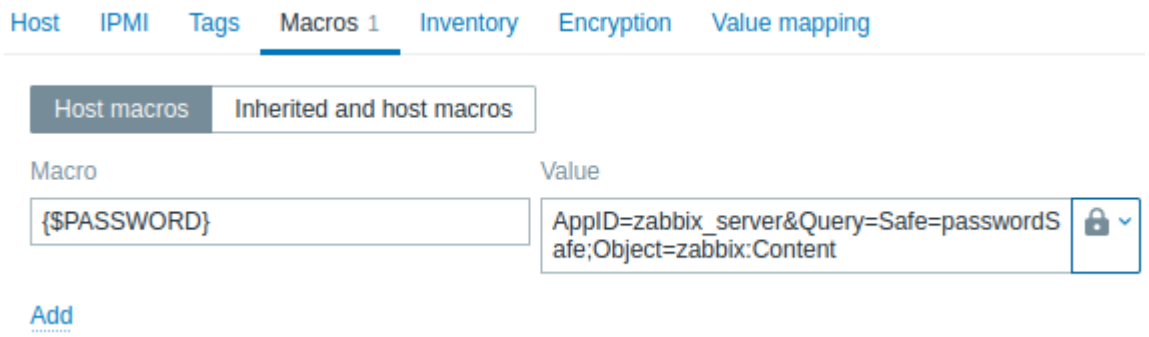
Синтаксис запроса

Символ двоеточия («:») зарезервирован для отделения запроса от ключа.

Если сам запрос содержит косую черту или двоеточие, эти символы должны указываться с использованием URL-кодирования («/» кодируется как «%2F», «:» кодируется как «%3A»).

Пример

1. В Zabbix добавьте пользовательский макрос {$PASSWORD} с типом «Секрет Хранилища» и со значением `AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix:Content`




2. Zabbix отправит следующий запрос API в хранилище:

```
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. Ответ хранилища будет содержать ключ «Content»:

```
{
    "Content": <   >,
    "UserName": < _       >,
    "Address": <   >,
    "Database" :<   _    >,
    "PasswordChangeInProcess":<            >
}
```

4. В результате Zabbix раскроет макрос {$PASSWORD} в значение — <пароль>.

Обновление существующей конфигурации

Чтобы обновить существующую конфигурацию для извлечения секретов из хранилища CyberArk:

1. Обновите параметры файла конфигурации сервера Zabbix или прокси, как описано в разделе *Учётные данные базы данных*.
2. Обновите параметры подключения к БД, перенастроив веб-интерфейс Zabbix и указав требуемые параметры, как описано в разделе *Веб-интерфейс*. Чтобы перенастроить веб-интерфейс Zabbix, откройте URL-адрес настройки веб-интерфейса в браузере:

- для Apache: http://<ip_или_имя_сервера>/zabbix/setup.php
- для Nginx: http://<ip_или_имя_сервера>/setup.php

Кроме того, эти параметры можно задать в файле конфигурации веб-интерфейса (*zabbix.conf.php*):

```
$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. Настройте пользовательские макросы, как описано в разделе *Значения пользовательских макросов*, если необходимо.

Чтобы обновить существующую конфигурацию для извлечения секретов из хранилища HashiCorp, смотрите *Настройка HashiCorp*.

#### 2. Настройка HashiCorp

Обзор

В этом разделе объясняется, как настроить Zabbix для извлечения секретов из HashiCorp Vault KV Secrets Engine - Version 2.

Хранилище должно быть развёрнуто и настроено, как описано в официальной документации HashiCorp [en].

Чтобы узнать о настройке TLS в Zabbix, смотрите *Хранение секретов*.

Получение учетных данных базы данных

Чтобы успешно получить секрет с учетными данными базы данных, необходимо настроить оба компонента:

- Zabbix server/proxy
- Zabbix веб-интерфейс

Server/proxy

Чтобы настроить Zabbix server или proxy, укажите следующие параметры конфигурации в файле конфигурации:

- `Vault` - какой поставщик хранилища следует использовать;
- `VaultToken` - токен аутентификации в хранилище (подробности см. в файле конфигурации Zabbix server/proxy);
- `VaultURL` - HTTP[S]-URL сервера хранилища;
- `VaultDBPath` - путь к секрету в хранилище, содержащему учетные данные базы данных (этот параметр можно использовать только если не указаны DBUser и DBPassword); Zabbix server или proxy получит учетные данные по ключам ”password” и ”username”;
- `VaultPrefix` - пользовательский префикс для пути или запроса к хранилищу, в зависимости от хранилища; если не указан, будет использовано наиболее подходящее значение по умолчанию.

**Attention:**
Параметры конфигурации `Vault`, `VaultToken`, `VaultURL` и `VaultPrefix` также используются для аутентификации в хранилище при обработке макросов секретов хранилища Zabbix server (и Zabbix proxy, если настроен). Zabbix server и прокси не будут открывать макросы секретов хранилища, содержащие учетные данные базы данных из VaultDBPath.<br><br>Настоятельно рекомендуется использовать разные токены для разных прокси.

Zabbix server и Zabbix proxy считывают связанные с хранилищем параметры конфигурации из *zabbix_server.conf* и *zabbix_proxy.conf* при запуске. Кроме того, Zabbix server и Zabbix proxy один раз при запуске считывают переменную окружения `VAULT_TOKEN` и затем удаляют ее, чтобы она не была доступна через порожденные скрипты; это ошибка, если и `VaultToken`, и `VAULT_TOKEN` содержат значение.

**Пример**

1. В *zabbix_server.conf* укажите следующие параметры:

```
Vault=HashiCorp
VaultToken=hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTYONFNSaUp3ZnpWbDF1RUNjUkNTZEg
VaultURL=https://127.0.0.1:8200
VaultDBPath=database
VaultPrefix=/v1/secret/data/zabbix/
```

2. Выполните следующие команды CLI, чтобы создать требуемый секрет в хранилище:

```
####                «secret/»,              ;          ,              «kv-v2».
vault secrets enable -path=secret/ kv-v2
```

```
####                    username   password              «secret/»      «secret/zabbix/database».
vault kv put -mount=secret zabbix/database username=zabbix password=<    >

####      ,                 .
vault kv get secret/zabbix/database

####     ,            Curl;          ,                  «data»                «/v1»                ,
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/database
```

3. В результате Zabbix сервер получит следующие учётные данные для аутентификации базы данных:

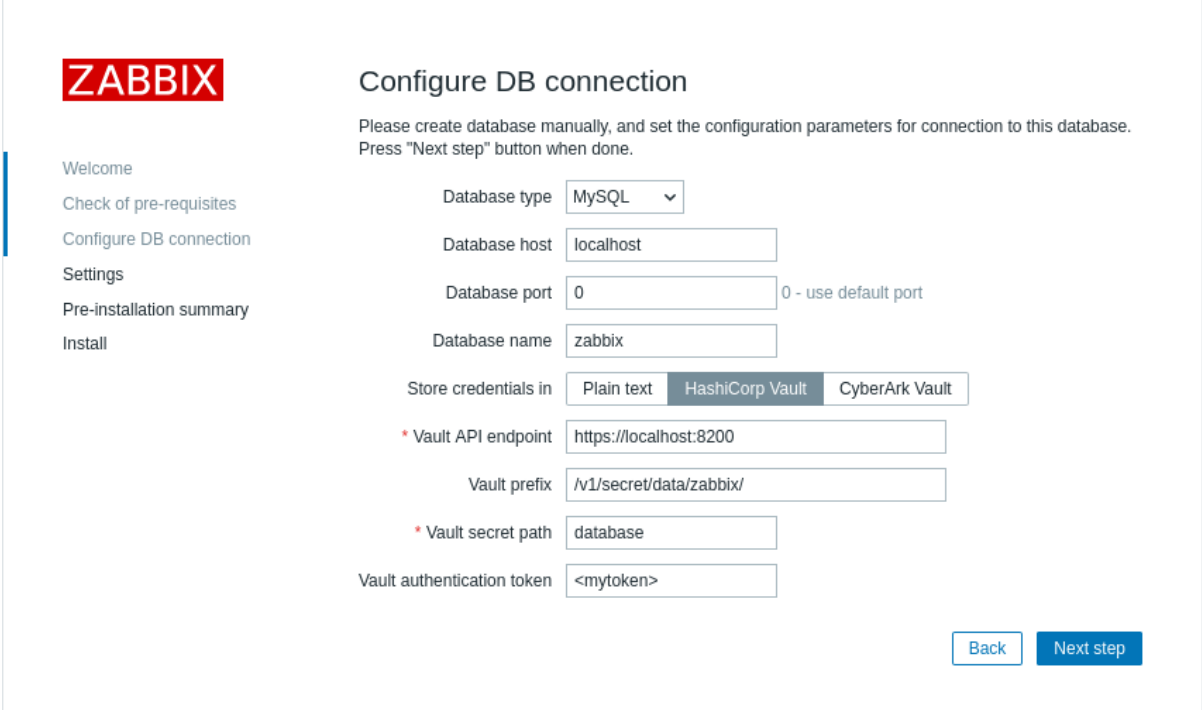
- Имя пользователя: zabbix
- Пароль: <пароль>

Веб-интерфейс

Веб-интерфейс Zabbix можно настроить для получения учётных данных базы данных из хранилища либо во время установки веб-интерфейса, либо путём обновления файла конфигурации веб-интерфейса (`zabbix.conf.php`).

**Attention:**
Если учётные данные хранилища были изменены с момента предыдущей установки веб-интерфейса, перезапустите установку веб-интерфейса или обновите `zabbix.conf.php`. Смотрите также: Обновление существующей конфигурации.

Во время **установки веб-интерфейса** параметры конфигурации должны быть указаны на шаге *Настройка соединения с базой данных (Configure DB Connection)*:




- Установите параметр *Хранить учетные данные в (Store credentials in)* в значение «HashiCorp Vault».
- Укажите параметры подключения:

| Параметр | Обязательный | Значение по умолчанию | Описание |
|---|---|---|---|
| *API endpoint Хранилища (Vault API endpoint)* | да | https://localhost:8200 | Укажите URL для подключения к хранилищу в формате `://:` |
| *Префикс Хранилища (Vault prefix)* | нет | /v1/secret/data/ | Укажите пользовательский префикс для пути или запроса хранилища. Если не указано, используется значение по умолчанию.<br>Пример: `/v1/secret/data/zabbix/` |
| *Путь к секрету Хранилища (Vault secret path)* | нет | | Путь к секрету, из которого будут извлечены учётные данные для базы данных по ключам «password» и «username».<br>Пример: `database` |

| Параметр | Обязательный? | Значение по умолчанию | Описание |
|---|---|---|---|
| *Токен аутентификации Хранилища (Vault authentication token)* | нет | | Предоставьте токен аутентификации для доступа только для чтения к секретному пути. Смотрите документацию HashiCorp [en] для получения информации о создании токенов и политик хранилища. |

Получение значений пользовательских макросов

Чтобы использовать HashiCorp Vault для хранения значений пользовательских макросов *Vault secret*, убедитесь, что:

- сервер/прокси Zabbix настроен для работы с HashiCorp Vault;
- параметр *Vault provider* в *Administration → General → Other* установлен в значение "HashiCorp Vault" (по умолчанию);

**Note:**
Сервер Zabbix (и прокси Zabbix, если настроен) требует доступа к значениям макросов *Vault secret* из хранилища. Веб-интерфейс Zabbix не нуждается в таком доступе.

Значение макроса должно содержать ссылочный путь (в виде `path:key`, например `macros:password`). Токен аутентификации, указанный при настройке сервера/прокси Zabbix (с помощью параметра `VaultToken`), должен предоставлять доступ только для чтения к этому пути.

Подробную информацию о том, как Zabbix обрабатывает значения макросов, см. в разделе *Vault secret macros*.

**Синтаксис пути**

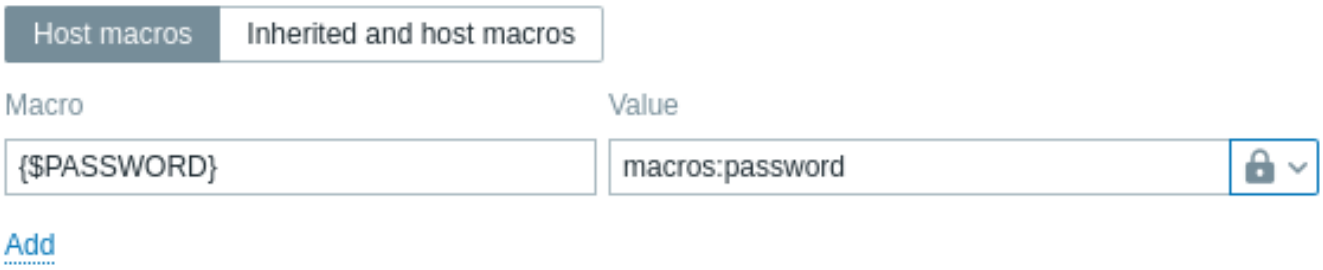
Прямая косая черта («/») и двоеточие («:») являются зарезервированными символами.

Прямая косая черта может использоваться только для отделения точки монтирования от пути (например, *secret/zabbix*, где точка монтирования — «secret», а путь — «zabbix»). В случае макросов Хранилища двоеточие может использоваться только для отделения пути/запроса от ключа.

Можно использовать URL-кодирование символов косой черты и двоеточия, если необходимо создать точку монтирования с именем, разделённым косой чертой (например, *foo/bar/zabbix*, где точка монтирования — «foo/bar», а путь — «zabbix», можно закодировать как «foo%2Fbar/zabbix»), и если имя или путь точки монтирования должны содержать двоеточие.

**Пример**

1. В Zabbix добавьте пользовательский макрос {$PASSWORD} с типом «Секрет Хранилища» и со значением `macros:password`




2. Выполните следующие команды CLI, чтобы создать требуемый секрет в хранилище:

```
####                «secret/»,              ;          ,              «kv-v2».
vault secrets enable -path=secret/ kv-v2

####                  password              «secret/»      «secret/zabbix».
```

```
vault kv put -mount=secret zabbix/macros password=<    >

####      ,                  .
vault kv get secret/zabbix/macros

####     ,             Curl,          ,                    «data»                  «/v1»                ,
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/macros
```

3. В результате Zabbix раскроет макрос {$PASSWORD} в значение: <пароль>

Обновление существующей конфигурации

Чтобы обновить существующую конфигурацию для извлечения секретов из хранилища HashiCorp:

1. Обновите параметры файла конфигурации сервера Zabbix или прокси, как описано в разделе *Учётные данные базы данных*.
2. Обновите параметры подключения к БД, перенастроив веб-интерфейс Zabbix и указав требуемые параметры, как описано в разделе *Веб-интерфейс*. Чтобы перенастроить веб-интерфейс Zabbix, откройте URL-адрес настройки веб-интерфейса в браузере:

- для Apache: http://<ip_или_имя_сервера>/zabbix/setup.php
- для Nginx: http://<ip_или_имя_сервера>/setup.php

Кроме того, эти параметры можно задать в файле конфигурации веб-интерфейса (*zabbix.conf.php*):

```
$DB['VAULT']                    = 'HashiCorp';
$DB['VAULT_URL']                = 'https://localhost:8200';
$DB['VAULT_DB_PATH']            = 'database';
$DB['VAULT_TOKEN']              = '<  _   >';
$DB['VAULT_CERT_FILE']          = '';
$DB['VAULT_KEY_FILE']           = '';
$DB['VAULT_PREFIX']             = '/v1/secret/data/zabbix/';
```

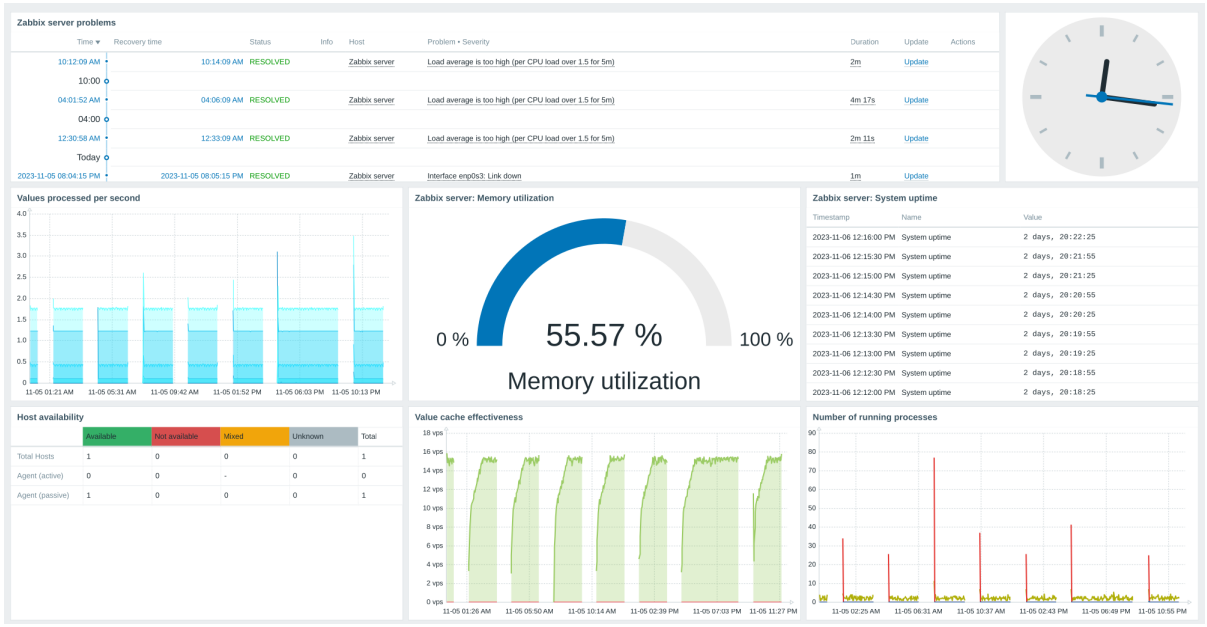
3. Настройте пользовательские макросы, как описано в разделе *Значения пользовательских макросов*, если необходимо.

Чтобы обновить существующую конфигурацию для извлечения секретов из хранилища CyberArk, смотрите *Настройка CyberArk*.

### 14. Регулярные отчёты

Обзор

С помощью функции *Запланированные отчеты* вы можете настроить отправку PDF-версии выбранной панели мониторинга указанным получателям через заданные интервалы.




Требования:

- Веб-служба Zabbix должна быть установлена и правильно настроена, чтобы обеспечить создание запланированных отчетов — см. инструкции в разделе Настройка запланированных отчетов.
- У пользователя должна быть роль пользователя типа *Admin* или *Super admin* со следующими правами:
  - *Scheduled reports* в блоке *Access to UI elements* (для просмотра настроек отчета)
  - *Manage scheduled reports* в блоке *Access to actions* (для создания/редактирования отчетов)

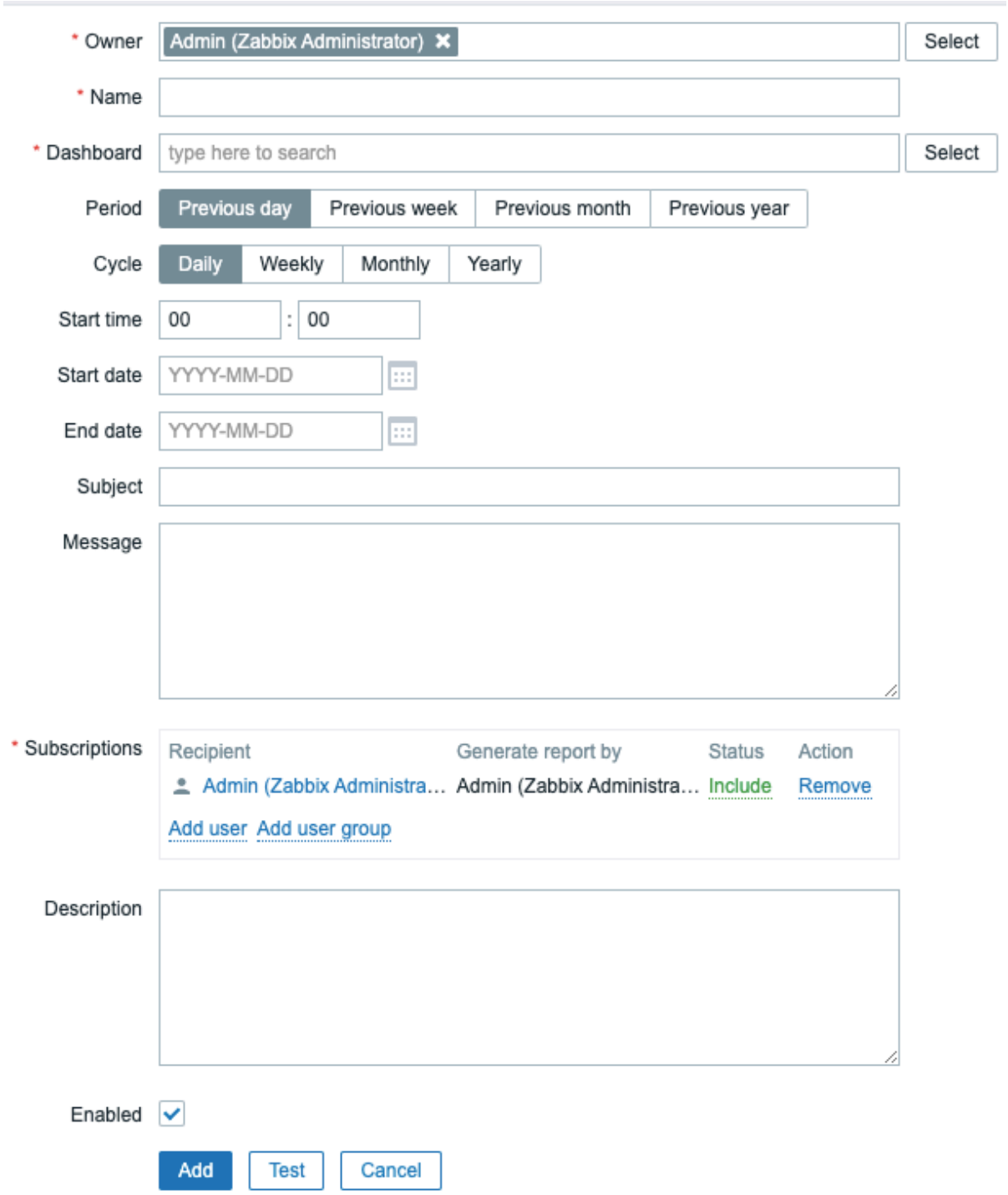
Чтобы создать запланированный отчет в веб-интерфейсе Zabbix, выполните следующие действия:

- Перейдите в: *Reports* > *Scheduled reports*.
- Нажмите *Create report* в правом верхнем углу экрана.
- Введите параметры отчета в форму.

Вы также можете создать отчет, открыв существующий, нажав кнопку *Clone*, а затем сохранив его под другим именем.

Конфигурация

Вкладка *Запланированные отчеты* содержит общие атрибуты отчета.




Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Owner* | Пользователь, который создает отчет. Пользователям уровня *Super admin* разрешено изменять владельца. Для пользователей уровня *Admin* это поле доступно только для чтения. |
| *Name* | Имя отчета; должно быть уникальным. |
| *Dashboard* | Панель мониторинга, на основе которой формируется отчет; одновременно можно выбрать только одну панель мониторинга. Чтобы выбрать панель мониторинга, начните вводить имя - появится список подходящих панелей мониторинга; прокрутите вниз, чтобы выбрать нужную. Также можно нажать *Select* рядом с полем и выбрать панель мониторинга из отображаемого списка. |
| *Period* | Период времени, за который будет подготовлен отчет. Выберите предыдущий день, неделю, месяц или год. |
| *Cycle* | Частота формирования отчета. Отчеты могут отправляться ежедневно, еженедельно, ежемесячно или ежегодно. Режим ”Weekly” позволяет выбрать дни недели, когда будет отправляться отчет. |

| Parameter | Description |
|---|---|
| *Start time* | Время суток в формате hh:mm, когда будет подготовлен отчет. Обратите внимание, что будет использоваться часовой пояс сервера Zabbix. |
| *Repeat on* | Дни недели, когда будет отправляться отчет. Это поле доступно только если для *Cycle* задано значение ”Weekly”. |
| *Start date* | Дата, когда следует начать регулярное формирование отчета. |
| *End date* | Дата, когда следует остановить регулярное формирование отчета. |
| *Subject* | Тема письма с отчетом. Поддерживаемые макросы: {TIME}, {TIMESTAMP}. |
| *Message* | Текст письма с отчетом. Поддерживаемые макросы: {TIME}, {TIMESTAMP}. |
| *Subscriptions* | Список получателей отчета. По умолчанию включает только владельца отчета. В качестве получателя отчета можно указать любого пользователя Zabbix с настроенным типом медиа email.<br>Нажмите *Add user* или *Add user group*, чтобы добавить дополнительных получателей.<br>Нажмите на имя пользователя, чтобы изменить настройки:<br>*Generate report by* - на основе прав доступа к панели мониторинга текущего пользователя или получателя должны быть сгенерированы данные отчета.<br>*Status* - выберите ”Include”, чтобы отправить отчет пользователю, или ”Exclude”, чтобы не отправлять отчет этому пользователю. Хотя бы один пользователь должен иметь статус ”Include”. Статус ”Exclude” можно использовать, чтобы исключить отдельных пользователей из включенной группы пользователей.<br><br>Обратите внимание, что пользователи с недостаточными правами (то есть пользователи с ролью на основе типа пользователя *Admin*, которые не являются членами той же группы пользователей, что и получатель или владелец отчета) будут видеть ”Inaccessible user” или ”Inaccessible user group” вместо фактических имен в полях *Recipient* и *Generate report by*; поля *Status* и *Action* будут отображаться только для чтения. |
| *Enabled* | Статус отчета. Снятие этого флажка отключит отчет. |
| *Description* | Необязательное описание отчета. Это описание предназначено для внутреннего использования и не будет отправляться получателям отчета. |

Кнопки диалога

Кнопки в нижней части диалога позволяют выполнить несколько видов операций.

| | |
|---|---|
| Add (Добавить) | Добавление отчёта. Эта кнопка доступна только для новых отчётов. |
| Update (Обновить) | Обновление свойств отчёта. |
| Clone (Клонировать) | Создание другого отчёта на основе свойств текущего отчёта. |
| Test (Тест) | Проверка корректности настройки отчёта путём отправки отчёта текущему пользователю. |
| Delete (Удалить) | Удаление отчёта. |
| Cancel (Отмена) | Отмена изменения свойств отчёта. |

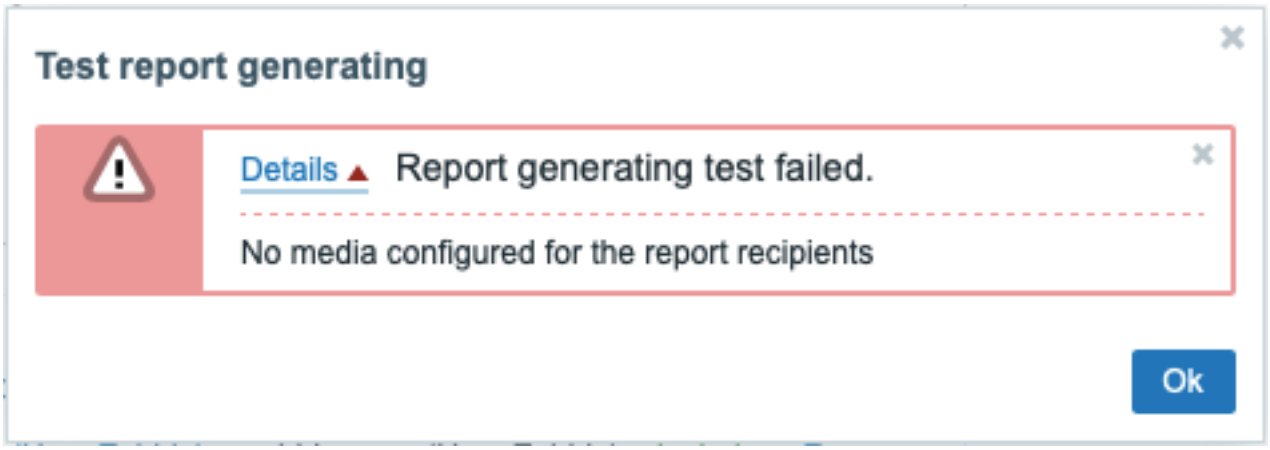
Тестирование

Для тестирования отчёта нажмите на кнопку *Тест (Test)* внизу диалога настройки отчёта.

**Note:**
Кнопка *Тест* недоступна, если диалог настройки отчёта открыт из панели меню действий.

Если конфигурация корректна, тестовый отчёт будет отправлен текущему пользователю незамедлительно. Для тестовых отчётов настройки пользователей для подписчиков и поля «Генерировать отчёт от» игнорируются.

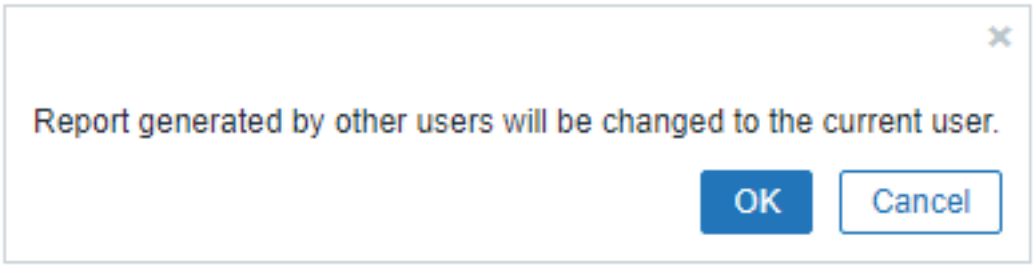
Если конфигурация некорректна, отобразится сообщение об ошибке с описанием возможной причины.




Изменение отчёта

Чтобы изменить существующий отчёт, нажмите на имя отчёта, затем внесите необходимые изменения и нажмите на кнопку *Обновить (Update)*.

Если существующий отчёт обновлён другим пользователем и этот пользователь меняет Панель, при нажатии на кнопку *Обновить* отобразится сообщение с предупреждением: «Отчёт, сгенерированный другими пользователями, изменится на текущего пользователя».




Нажатие *OK* на этом шаге приведёт к следующим изменениям:

- Настройки *Сгенерировано (Generate report by)* будут обновлены, чтобы отображался тот пользователь, который редактировал отчёт в последний раз (если только *Сгенерировано* не задано значением Получатель).
- Пользователи, которые отображались как «Недоступный пользователь» или «Недоступная группа пользователей», будут удалены из списка подписчиков отчёта.

Нажатие *Отмена (Cancel)* закроет всплывающее окно и отменит обновление отчёта.

Клонирование отчёта

Для быстрого клонирования существующего отчёта нажмите кнопку *Клонировать (Clone)* внизу диалога настройки существующего отчёта. При клонировании отчёта, созданного другим пользователем, текущий пользователь становится владельцем нового отчёта.

Настройки отчета будут скопированы в диалог настройки нового отчёта с учётом прав пользователя:

- Если пользователь, который клонирует отчёт, не имеет прав доступа к панели, поле *Панель (Dashboard)* очищается.
- Если пользователь, который клонирует отчёт, не имеет прав доступа к некоторым пользователям или группам пользователей из списка *Подписки (Subscriptions)*, недоступные получатели не клонируются.
- Настройки *Сгенерировано (Generate report by)* будут обновлены, чтобы отображался текущий пользователь (если только *Сгенерировано* не задано значением Получатель).

Измените требуемые настройки и имя отчёта, затем нажмите на *Добавить (Add)*.

### 15. Экспорт данных

Обзор

Zabbix поддерживает экспорт данных в режиме реального времени двумя способами:

- экспорт в файлы
- потоковая передача во внешние системы

Могут быть экспортированы следующие объекты:

- триггерные события
- значения элементов данных
- динамика изменений (только экспорт в файлы)

### 1. Экспорт в файлы

Обзор

Можно настроить экспорт в реальном времени событий триггеров, значений элементов данных и трендов в формате JSON, разделенном по строкам.

Экспорт выполняется в файлы, где каждая строка экспортного файла является объектом JSON. Преобразования значений не применяются.

В случае ошибок (данные не могут быть записаны в экспортный файл, либо экспортный файл не может быть переименован, либо после переименования не может быть создан новый файл), элемент данных отбрасывается и никогда не записывается в экспортный файл. Он записывается только в базу данных Zabbix. Запись данных в экспортный файл возобновляется, когда проблема с записью будет устранена.

Начиная с Zabbix 7.4.6, экспортный файл создается с правами чтения и записи только для владельца файла. Кроме того, файл доступен для чтения группе владельца. Все остальные права запрещены.

Подробные сведения о том, какая информация экспортируется, см. на странице протокола экспорта.

Обратите внимание, что у узла сети/элемента данных может отсутствовать метаданные (группы узлов сети, имя узла сети, имя элемента данных), если узел сети/элемент данных был удален после получения данных, но до того, как сервер экспортировал данные.

Настройка

Экспорт в режиме реального времени событий по триггерам, значений элементов данных и динамики изменений настраивается путём задания директории для файлов экспорта — смотрите параметр ExportDir в настройках сервера.

Доступны два других параметра:

- `ExportFileSize` можно использовать, чтобы задать максимально разрешённый размер отдельного файла экспорта. Когда процессу требуется записать в файл, он сначала проверяет размер файла. Если он превышает настроенное ограничение размера, файл переименовывается путём добавления «.old» к его имени и создаётся новый файл с исходным именем.

**Attention:**
Файл будет создан на каждый процесс, который будет писать данные (т.е. примерно 4-30 файлов). Поскольку размер по умолчанию на один файл экспорта составляет 1ГБ, хранение больших файлов экспорта может быстро исчерпать место на диске.

- `ExportType` позволяет указать, какие типы объектов (события, история, динамика изменений) будут экспортироваться.

### 2. Потоковая передача во внешние системы

Обзор

Возможна потоковая передача значений элементов данных и событий из Zabbix во внешние системы по HTTP (смотрите подробности протокола).

Для потоковой передачи подмножеств значений элементов данных или событий можно использовать фильтр тегов.

За потоковую передачу данных отвечают два типа процессов сервера Zabbix: менеджер коннекторов (`connector manager`) и рабочий процесс коннектора (`connector worker`). Внутренний элемент данных Zabbix `zabbix[connector_queue]` позволяет отслеживать количество значений, поставленных в очередь коннектора.

Настройка

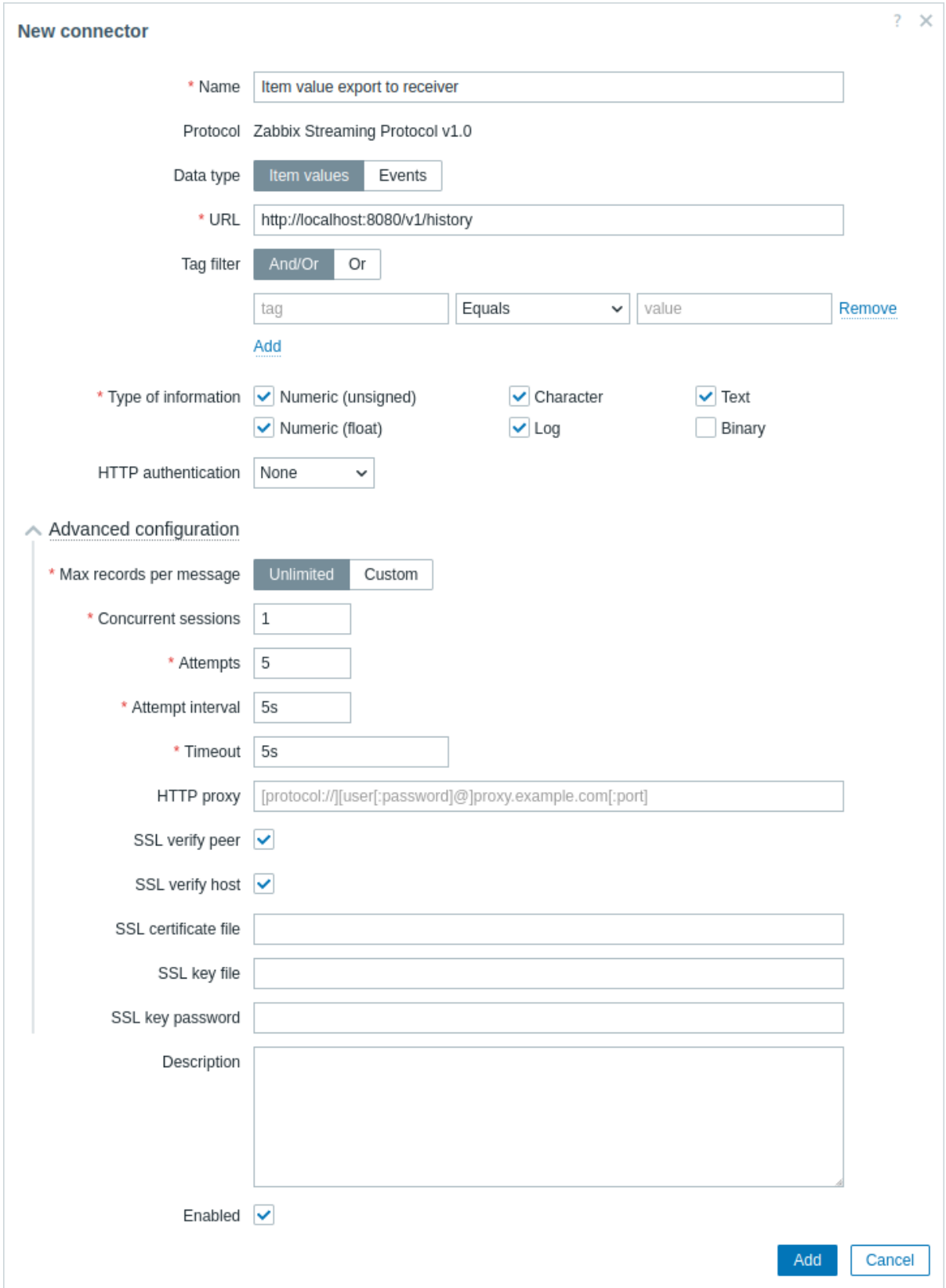
Для настройки потоковой передачи данных во внешнюю систему необходимы следующие шаги:

1. Наличие удалённой системы, настроенной на получение данных из Zabbix. Для этой цели доступны следующие инструменты:

- Пример простого получателя [en], который записывает принимаемую информацию в файлы `events.ndjson` и `history.ndjson`.
- Kafka connector for Zabbix server [en] — написанный на Go легковесный сервер, предназначенный для пересылки значений элементов данных и событий от сервера Zabbix брокеру Kafka.

2. Настройте нужное количество рабочих процессов коннектора в Zabbix путём регулирования параметра `StartConnectors` в `zabbix_server.conf`. Количество рабочих процессов коннектора должно соответствовать (или превышать, если одновременных сессий больше одной) настроенному в веб-интерфейсе Zabbix числу сконфигурированных коннекторов. После этого перезапустите Zabbix сервер.

3. Сконфигурируйте новый коннектор в веб-интерфейсе Zabbix (*Администрирование* → *Общие* → *Коннекторы* (*Administration* → *General* → *Connectors*)) и перезагрузите кэш сервера с помощью команды `zabbix_server -R config_cache_reload`.




Обязательные поля отмечены звёздочкой.

| Параметр | Описание |
|---|---|
| *Имя (Name)* | Введите имя коннектора. |
| *Тип данных (Data type)* | Выберите тип данных для потоковой передачи:<br>**Значения элемента данных (Item values)** — трансляция значений элементов данных из Zabbix во внешние системы;<br>**События (Events)** — трансляция событий из Zabbix во внешние системы. |
| *URL* | Введите URL получателя. Поддерживаются пользовательские макросы. |

| Параметр | Описание |
|---|---|
| *Фильтр тегов (Tag filter)* | Экспортировать только значения элементов данных или события, соответствующие фильтру тегов. Если не задано, то экспортировать всё.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно задать несколько условий. Имена тегов всегда сопоставляются с учётом регистра.<br><br>Для каждого условия доступно несколько операторов:<br>**Существует (Exists)** — включать указанные имена тегов;<br>**Равно (Equals)** — включать указанные имена тегов и значения (с учётом регистра);<br>**Содержит (Contains)** — включать указанные имена тегов, где значения тегов содержат введённую строку (сопоставление подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключать указанные имена тегов;<br>**Не равно (Does not equal)** — исключать указанные имена тегов и значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключать указанные имена тегов, где значения тегов содержат введённую строку (сопоставление подстроки, без учёта регистра).<br><br>Для условий есть два типа вычислений:<br>**И/ИЛИ (And/Or)** — должны выполняться все условия, условия с тем же именем тега будут сгруппированы с условием ИЛИ;<br>**Или (Or)** — достаточно выполнения одного условия. |
| *Тип информации (Type of information)* | Выберите тип информации (числовой (целое положительное), числовой (с плавающей точкой), символ и т.д.), по которому фильтровать значения элементов данных, транслируемых коннектором.<br>Это поле доступно, если в поле *Тип данных* выбрано «Значения элемента данных». |
| *HTTP аутентификация (HTTP authentication)* | Выберите вариант аутентификации:<br>**Нет (None)** — аутентификация не используется;<br>**Простая (Basic)** — используется простая (basic) аутентификация;<br>**NTLM** — используется аутентификация NTLM (Windows NT LAN Manager);<br>**Kerberos** — используется аутентификация Kerberos (смотрите также: Настройка Kerberos с Zabbix);<br>**Digest** — используется аутентификация Digest;<br>**Bearer** — используется аутентификация Bearer. |
| *Имя пользователя (Username)* | Введите имя пользователя (до 255 символов). Поддерживаются пользовательские макросы.<br>Это поле доступно, если параметр *HTTP аутентификация* установлен в значение «Basic», «NTLM», «Kerberos» или «Digest». |
| *Пароль (Password)* | Введите пароль пользователя (до 255 символов). Поддерживаются пользовательские макросы.<br>Это поле доступно, если параметр *HTTP аутентификация* установлен в значение «Basic», «NTLM», «Kerberos» или «Digest». |
| *Токен на предъявителя (Bearer token)* | Введите Токен на предъявителя. Поддерживаются пользовательские макросы.<br>Это поле доступно, если параметр *HTTP аутентификация* установлен в значение «Bearer». |
| *Расширенная настройка (Advanced configuration)* | Нажмите на заголовок *Расширенная настройка (Advanced configuration)* для показа дополнительных параметров конфигурации (смотрите ниже). |
| *Макс. записей на сообщение (Max records per message)* | Укажите максимальное количество значений или событий, которые могут быть переданы в одном сообщении. |
| *Одновременные сессии (Concurrent sessions)* | Выберите количество отсылающих процессов, которые будут запущены для этого коннектора. Можно указать до 100 сессий; значение по умолчанию — «1». |
| *Попыток (Attempts)* | Число попыток для потоковой передачи данных. Можно указать до 5 попыток; значение по умолчанию — «1». |

| Параметр | Описание |
|---|---|
| *Интервал между попытками (Attempt interval)* | Укажите, как долго коннектор должен ждать после неудачной попытки потоковой передачи данных. Можно указать до 10s; значение по умолчанию — «5s».<br>Это поле доступно, если в поле *Попыток* указано «2» или больше.<br>Неудачными считаются попытки, при которых не удалось установить соединение или код ответа HTTP не равен 200, 201, 202, 203, 204. Повторные попытки **инициируются** в случае ошибок связи или когда код ответа HTTP не равен 200, 201, 202, 203, 204, 400, 401, 403, 404, 405, 415, 422. Перенаправления выполняются, поэтому 302 -> 200 является успешным ответом; в то время как 302 -> 503 будет инициировать повторную попытку. |
| *Время ожидания (Timeout)* | Укажите время ожидания для сообщения (1-60 секунд, по умолчанию — 5 секунд).<br>Поддерживаются суффиксы времени, например: 30s, 1m. Поддерживаются пользовательские макросы. |
| *HTTP прокси (HTTP proxy)* | Можно указать используемый HTTP прокси в следующем формате:<br>`[ ://][ _ [: ]@]proxy.example.com[: ]`<br>Поддерживаются пользовательские макросы.<br><br>Необязательный префикс `://` можно использовать для указания альтернативных протоколов прокси (поддержка префикса протокола была добавлена в cURL 7.21.7). Без указания протокола прокси будет считаться как HTTP прокси. По умолчанию будет использоваться порт 1080.<br><br>Если параметр *HTTP прокси* задан, он перезапишет относящиеся к прокси переменные окружения — такие как `http_proxy` и `HTTPS_PROXY`. Если не указан, относящиеся к прокси переменные окружения не будут перезаписаны. Введённое значение передаётся как есть, проверка его на корректность не выполняется.<br>Также можно ввести адрес SOCKS прокси. Если вы укажете ошибочный протокол, коннектор не сможет успешно передать значения элементов данных или события из Zabbix.<br><br>Обратите внимание, что с HTTP прокси поддерживается только простая аутентификация. |
| *Проверка SSL узла (SSL verify peer)* | Отметьте флажок для проверки SSL сертификата веб-сервера.<br>Сертификат сервера будет автоматически взят из местоположения общесистемного центра сертификации (certificate authority, CA). Вы можете переопределить местоположение файлов CA, используя параметр конфигурации Zabbix сервера или прокси `SSLCALocation`. |
| *Проверка SSL хоста (SSL verify host)* | Отметьте флажок для проверки того, что поле *Common Name* или *Subject Alternate Name* сертификата веб-сервера соответствует.<br>Это устанавливает опцию cURL `CURLOPT_SSL_VERIFYHOST` [en]. |
| *Файл SSL сертификата (SSL certificate file)* | Имя файла SSL сетрификата, используемого для аутентификации клиента. Файл сертификата должен быть в формате PEM[1]. Поддерживаются пользовательские макросы.<br>Если файл сертификата также содержит закрытый ключ, оставьте поле *Файл SSL ключа* пустым. Если ключ зашифрован, укажите пароль в поле *Пароль к SSL ключу*. Папка, содержащая этот файл, задаётся параметром конфигурации Zabbix сервера или прокси `SSLCertLocation`. |
| *Файл SSL ключа (SSL key file)* | Имя файла закрытого ключа SSL, используемого для аутентификации клиента. Файл закрытого ключа должен быть в формате PEM[1]. Поддерживаются пользовательские макросы.<br>Папка, содержащая этот файл, задаётся параметром конфигурации Zabbix сервера или прокси `SSLKeyLocation`. |
| *Пароль к SSL ключу (SSL key password)* | Пароль к файлу закрытого ключа SSL. Поддерживаются пользовательские макросы. |
| *Описание (Description)* | Введите описание коннектора. |
| *Активировано (Enabled)* | Отметьте флажок, чтобы коннектор был активирован. |

Если коннектор Kafka настроен с использованием разделённого запятыми списка адресов брокеров начальной загрузки (например, `Kafka.URL=kafka1.example.com:9093,kafka2.example.com:9093`), то клиент Kafka подключается к брокеру(ам), который(е) ответил(ли) первым(и), и использует метаданные их кластера. Если список содержит адреса из разных кластеров Kafka, будет использован только кластер, ответивший первым, а остальные адреса будут отмечены в журнале как недоступные; в результате при старте могут появиться предупреждения, подобные следующим, несмотря на то, что коннектор подключён:

```
kafka cluster connected, but broker(s) "kafka1.example.com:9093, kafka2.example.com:9093" unavailable; wil
```

В некоторых средах (частных сетях, контейнерных сетях или нестандартных настройках DNS/хостов) имена хостов или IP-адреса могут преобразовываться в loopback-адреса (например, `127.0.0.1/localhost`) или нормализоваться клиентом, что может сделать такие предупреждения вводящими в заблуждение. Чтобы избежать путаницы, убедитесь, что все адреса `Kafka.URL` относятся к одному кластеру Kafka, проверьте разрешение DNS с хоста коннектора и `advertised.listeners` брокера и отдавайте предпочтение адресам, которые преобразуются в объявленный адрес брокера.

Протокол

Обмен данными между сервером и получателем осуществляется по HTTP с использованием REST API, NDJSON, "Content-Type: application/x-ndjson".

Более подробную информацию смотрите в Протоколе экспорта JSON с разделителем в виде перевода строки.

Запрос сервера

Пример потоковой передачи значений элементов данных:

```
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
```

Пример потоковой передачи событий:

```
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}
```

Ответ получателя

Ответ состоит из кода состояния ответа HTTP и полезной нагрузки JSON. Код состояния ответа HTTP должен быть «200», «201», «202», «203» или «204» для запросов, которые были обработаны успешно, и другим для невыполненных запросов.

Пример успешного приёма:

```
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"}
```

Пример с ошибками:

```
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"}
```

### 3. SNMP gateway

Обзор

Zabbix SNMP gateway — это расширение AgentX для snmpd, поддерживающее как опрос SNMP, так и прием SNMP traps.

С помощью Zabbix SNMP gateway можно использовать протокол SNMP для получения:

- данных триггера;
- данных проблемного триггера;
- состояния группы узлов сети (количество триггеров по состоянию триггера в каждой группе)

Данные извлекаются по OID, который представляет собой комбинацию общего базового значения и конкретного суффикса. Общий **base** задается в конфигурационном файле SNMP gateway, например:

- `BaseOID=1.3.6.1.4.1.3043.7.55` - для любых данных триггера;
- `ProblemBaseOID=1.3.6.1.4.1.3047.7.55` - для данных проблемного триггера;
- `BaseOID=1.3.6.1.4.1.3046.7.55` - для состояния группы узлов сети.

OID **suffix** задается в конфигурации триггеров на узле сети как **tag** (например, `OIDSuffix:3`) во веб-интерфейсе.

В этом случае вся информация для триггера будет доступна по адресу `OID=1.3.6.1.4.1.3043.7.55.X.3`. Здесь "X" — это количество полей данных триггера (то есть 1 - suffix, 2 - ID, 3 - expression, 4 - description и т. д.).

Более подробное описание и пример файла конфигурации см. в файле readme SNMP gateway.

Установка и настройка

См. файл readme шлюза SNMP для инструкций по:

- установке и настройке snmpd;
- включению поддержки AgentX;
- настройке шлюза Zabbix SNMP;
- настройке SNMP traps для изменений состояния триггера.

Получение данных

После того как все правильно настроено, вы можете использовать команды `snmpwalk` и `snmpget` для получения данных:

```
[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
...
```

Параметры фильтрации

Вы можете ограничить информацию о триггерах с проблемами в конфигурации SNMP gateway:

- по уровню серьезности (по умолчанию `ProblemMinSeverity=-1`)
- путем скрытия подтвержденных проблем (по умолчанию `ProblemHideAck=false`)

Вы можете ограничить количество проблем на группу узлов сети в конфигурации SNMP gateway:

- по триггерам в неизвестном состоянии (по умолчанию `CountUnknown=false`)
- по триггерам с подтвержденными/неподтвержденными/всеми проблемами (по умолчанию `CountAcknowledgeStatus=all`)

## 8. Мониторинг услуг

**Обзор** Мониторинг услуг — это мониторинг бизнес-уровня, который можно использовать для обзора всего дерева услуг IT инфраструктуры, выявлять слабые места инфраструктуры, вычислять SLA по различным IT услугам, а также проверять другую информацию на более высоком уровне. Мониторинг услуг сфокусирован на общей доступности услуги вместо низкоуровневых деталей, таких как нехватка места на диске, высокая загрузка процессора и т.д. Мониторинг услуг также обеспечивает возможность поиска основной причины проблемы, если услуга не работает, как ожидается.

Мониторинг услуг позволяет создать иерархическое представление наблюдаемых данных.

Очень простая структура услуг может выглядеть следующим образом:

```
|
|-
|  |
|  |-          1
|  |
|  |-          2
|
|-
```

Каждый узел структуры имеет атрибут состояния. Это состояние вычисляется и распространяется на верхние уровни в соответствии с выбранным алгоритмом. На состояние отдельных узлов влияет состояние связанных с ними проблем. Сопоставление проблем выполняется при помощи тегов.

Zabbix может отправлять оповещения или автоматически выполнять скрипт на стороне Zabbix сервера, в случае обнаружения изменения состояния услуги. Можно задать гибкие правила, должна ли родительская услуга переходить в «Состояние проблемы» на основании состояний дочерних услуг. Данные проблем услуг можно затем использовать для вычисления SLA и отправки SLA отчётов на основе гибкого набора условий.

Мониторинг услуг настраивается в меню «Услуги», которое состоит из следующих разделов:

- Услуги

Раздел услуг позволяет создать иерархию вашей инфраструктуры, которую вы наблюдаете, путем добавления родительских услуг, и затем — дочерних услуг к родительским.

Помимо настройки дерева услуг, этот раздел предусматривает обзор всей инфраструктуры и позволяет быстро идентифицировать проблемы, которые привели к изменению состояния услуги.

- SLA

В этом разделе вы можете задать соглашения об уровне обслуживания и указать нормы уровней обслуживания по отдельным услугам.

- SLA отчёт

В этом разделе вы можете просматривать SLA отчёты.

**Действия на услуги**

Вы также можете настроить действия службы.

Действия на услуги являются необязательными и позволяют:

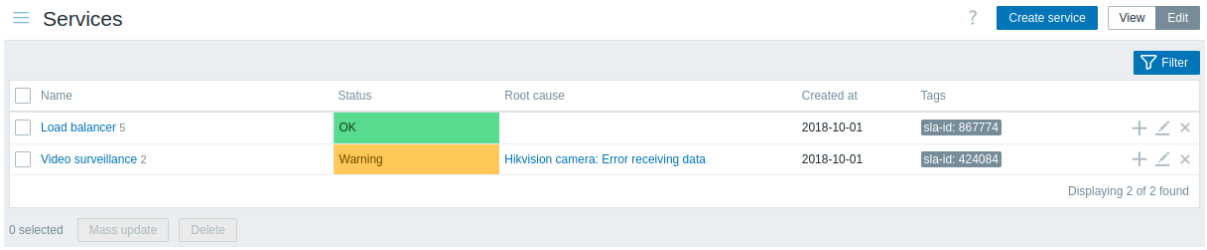
- отправить оповещение о том, что услуга недоступна;
- выполнить удалённую команду на стороне Zabbix сервера при изменении состояния услуги;
- отправить оповещение о восстановлении, когда услуга снова заработает.

**Смотрите также:**

- Пример настройки SLA мониторинга
- Заметки об обновлении услуг с версий Zabbix ниже 6.0

**1 Дерево услуг**

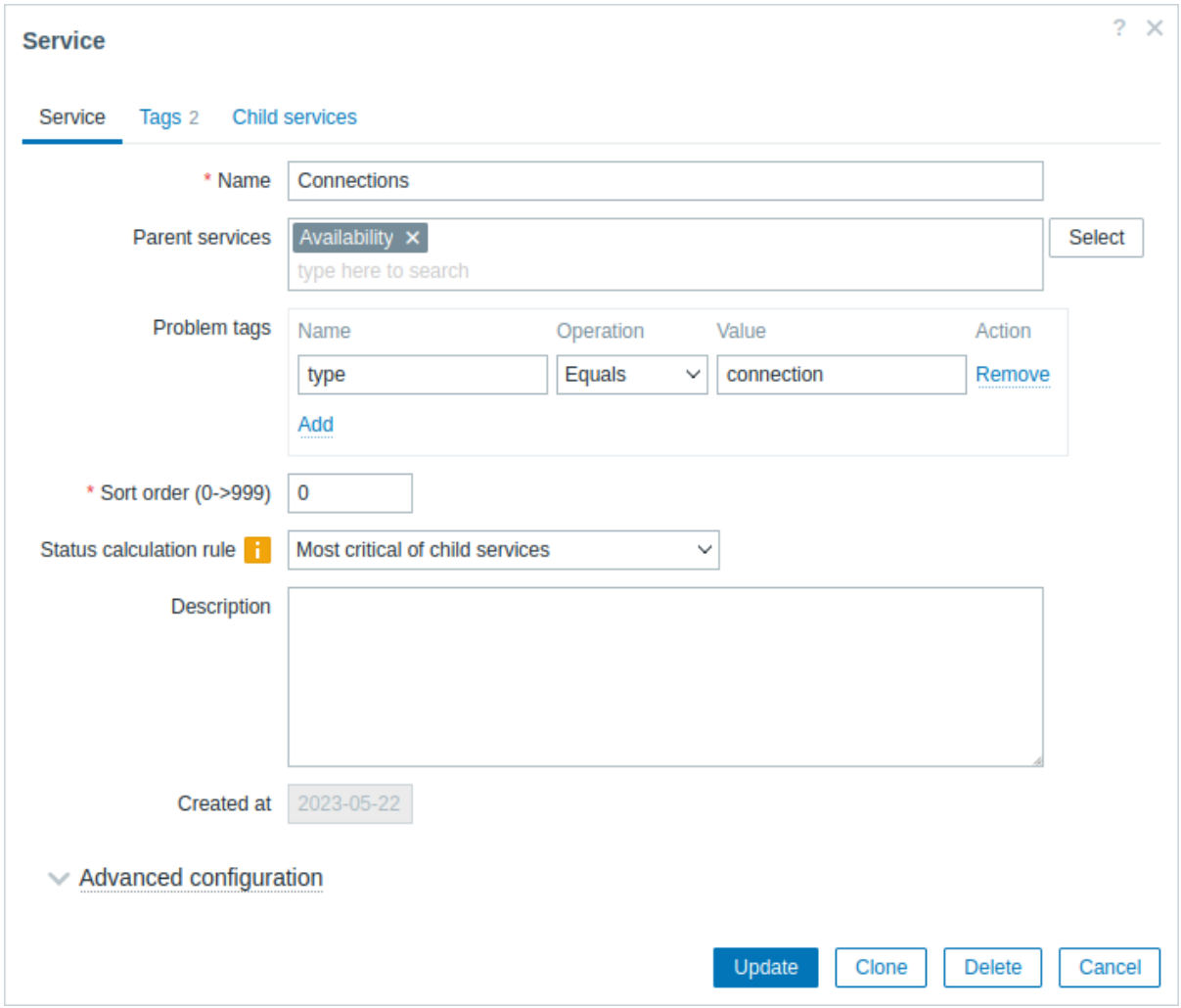
Дерево услуг настраивается в разделе меню *Services* -> *Services*. В правом верхнем углу переключитесь из режима View в режим редактирования.




Чтобы настроить новую услугу, нажмите кнопку *Create service* в правом верхнем углу.

Чтобы быстро добавить дочернюю услугу, можно также нажать значок плюса рядом с родительской услугой. Откроется та же форма настройки услуги, но параметр *Parent services* будет предварительно заполнен.

**Конфигурация сервиса** На вкладке **Service** укажите требуемые параметры сервиса:

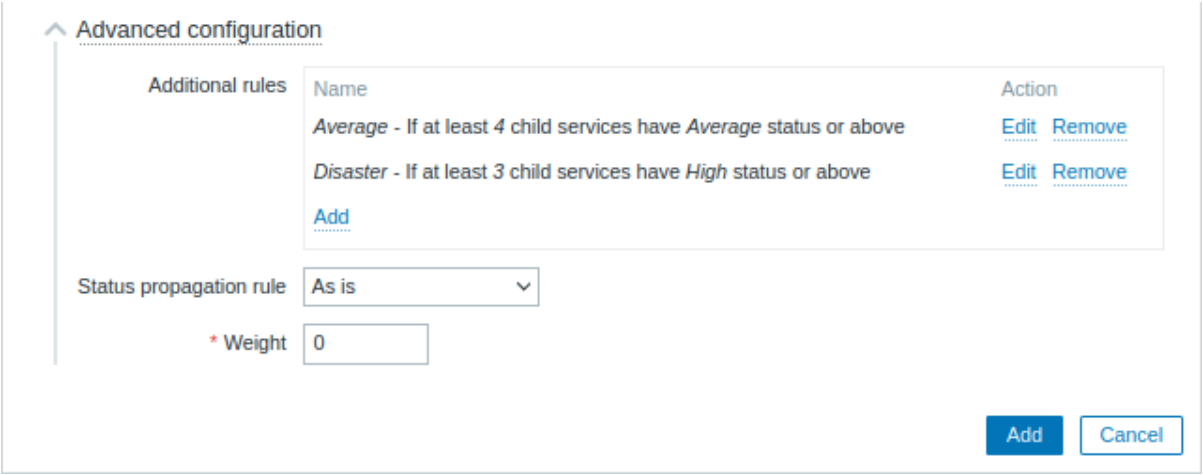



Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| ***Name*** | Имя сервиса. |
| ***Parent services*** | Родительские сервисы, к которым принадлежит сервис.<br>Оставьте это поле пустым, если вы добавляете сервис самого верхнего уровня.<br>Один сервис может иметь несколько родительских сервисов. В этом случае он будет отображаться в дереве сервисов под каждым из родительских сервисов. |
| ***Problem tags*** | Укажите теги для сопоставления данных о проблемах с сервисом.<br>Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны два оператора:<br>**Equals** - включать указанные имена и значения тегов (с учетом регистра)<br>**Contains** - включать указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, с учетом регистра) |

| Parameter | Description |
|---|---|
| ***Sort order*** | Порядок сортировки для отображения, меньшие значения идут первыми. |
| ***Status calculation rule*** | Правило вычисления статуса сервиса:<br>**Most critical if all children have problems** - наиболее критичная проблема в дочерних сервисах используется для окраски статуса сервиса, если у всех дочерних сервисов есть проблемы<br>**Most critical of child services** - наиболее критичная проблема в дочерних сервисах используется для окраски статуса сервиса<br>**Set status to OK** - не вычислять статус сервиса<br>Дополнительные правила вычисления статуса можно настроить в параметрах advanced configuration. |
| ***Description*** | Описание сервиса. |
| ***Created at*** | Время создания сервиса; отображается при редактировании существующего сервиса. |
| ***Advanced configuration*** | Нажмите заголовок *Advanced configuration*, чтобы отобразить параметры advanced configuration. |

Расширенная настройка




| Parameter | Description |
|---|---|
| ***Additional rules*** | Нажмите *Add*, чтобы настроить дополнительные правила расчета статуса. |
| *Set status to* | Установите статус сервиса в *OK* (по умолчанию), *Not classified*, *Information*, *Warning*, *Average*, *High* или *Disaster* при совпадении условия. |
| *Condition* | Выберите условие для прямых дочерних сервисов:<br>**если как минимум (N) дочерних сервисов имеют статус (Status) или выше**<br>**если как минимум (N%) дочерних сервисов имеют статус (Status) или выше**<br>**если менее чем (N) дочерних сервисов имеют статус (Status) или ниже**<br>**если менее чем (N%) дочерних сервисов имеют статус (Status) или ниже**<br>**если вес дочерних сервисов со статусом (Status) или выше составляет как минимум (W)**<br>**если вес дочерних сервисов со статусом (Status) или выше составляет как минимум (N%)**<br>**если вес дочерних сервисов со статусом (Status) или ниже меньше (W)**<br>**если вес дочерних сервисов со статусом (Status) или ниже меньше (N%)**<br><br>Если указано несколько условий и ситуация соответствует более чем одному условию, будет установлен наивысший уровень серьезности. |
| *N (W)* | Укажите значение N или W (1-100000), либо N% (1-100) в условии. |
| *Status* | Выберите значение *Status* в условии: *OK* (по умолчанию), *Not classified*, *Information*, *Warning*, *Average*, *High* или *Disaster*. |
| ***Status propagation rule*** | Правило распространения статуса сервиса на родительский сервис:<br>**As is** - статус передается без изменений<br>**Increase by** - можно увеличить передаваемый статус на 1-5 уровней серьезности<br>**Decrease by** - можно уменьшить передаваемый статус на 1-5 уровней серьезности<br>**Ignore this service** - статус вообще не передается родительскому сервису<br>**Fixed status** - статус передается статически, то есть всегда остается одинаковым |

| Parameter | Description |
|---|---|
| ***Weight*** | Вес сервиса (целое число в диапазоне от 0 (по умолчанию) до 1000000). |

**Note:**
Дополнительные правила расчета статуса можно использовать только для повышения уровня серьезности относительно уровня, рассчитанного по основному параметру *Status calculation rule*. Если согласно дополнительным правилам статус должен быть Warning, но согласно *Status calculation rule* статус - Disaster, сервис будет иметь статус Disaster.

Вкладка **Tags** содержит теги уровня сервиса. Теги уровня сервиса используются для идентификации сервиса. Теги этого типа не используются для сопоставления проблем с сервисом (для этого используйте *Теги проблем* на первой вкладке).

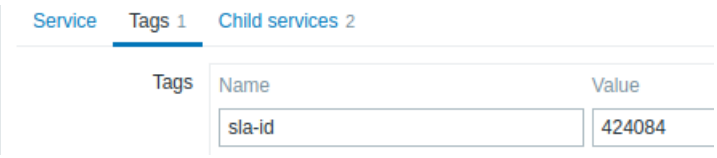
Вкладка **Child services** позволяет указать зависимые сервисы. Нажмите *Add*, чтобы добавить сервис из списка существующих сервисов. Если вы хотите добавить новый дочерний сервис, сначала сохраните этот сервис, а затем нажмите значок плюса рядом с только что созданным сервисом.

**Теги** В услугах имеется два разных типа тегов:

- Теги услуг
- Теги проблем

Теги услуг

Теги услуг используются для сопоставления услуг с действиями на услуги и SLA. Эти теги задаются на вкладке *Теги* настройки услуги. Для поиска совпадения SLA используется логика *ИЛИ*: услуга будет сопоставлена с SLA, если она имеет по крайней мере один подходящий тег. В действиях на услуги правила сопоставления настраиваются и могут использовать логику *И*, *ИЛИ* или *И/ИЛИ*.

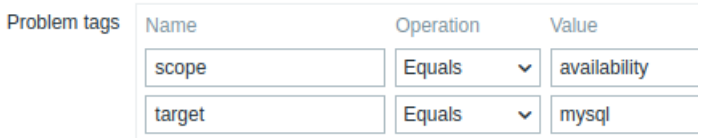



Теги проблем

Теги проблем используются для сопоставления проблем и услуг. Эти теги указываются на основной вкладке настройки услуги.

Только дочерние услуги самого нижнего уровня иерархии могут иметь теги проблем и быть непосредственно связаны с проблемами. Если теги проблем совпадают, состояние услуги изменится на то же самое состояние, что и проблема. В случае наличия нескольких проблем, услуге присваивается состояние самой серьёзной проблемы. Затем состояние родительской услуги вычисляется на основе состояний дочерних услуг в соответствии с Правилами вычисления состояния.

Если указано несколько тегов, используется логика *И*: проблема должна иметь все теги, заданные в настройках услуги, чтобы сопоставление имело место.




**Note:**
Проблема в Zabbix наследует теги со всей цепочки шаблонов, узлов сети, элементов данных, веб-сценариев и триггеров. Любые из этих тегов можно использовать для сопоставления проблем с услугами.

*Пример:*

Проблема *Веб камера 3 недоступна* имеет теги : , :1 и : −3 и состояние *Предупреждение*

Услуга **Веб камера 3** имеет единственный тег проблемы: : 3

При обнаружении этой проблемы состояние услуги изменится с *OK* на *Предупреждение*.

Если услуга **Веб камера 3** имеет теги проблем : −3 и :2, её состояние не изменится при обнаружении проблемы, так как условия удовлетворены лишь частично.

**Изменение существующих сервисов**

**Note:**
Кнопки, описанные ниже, видны только тогда, когда раздел *Services* находится в реж

Чтобы изменить существующий сервис, нажмите на значок карандаша рядом с сервисом.

Чтобы клонировать существующий сервис, нажмите на значок карандаша, чтобы открыть его конфигурацию, а затем нажмите кнопку *Clone*. При клонировании сервиса его связи с родительскими элементами сохраняются, а связи с дочерними элементами - нет.

Чтобы удалить сервис, нажмите на значок x рядом с ним. При удалении родительского сервиса его дочерние сервисы не будут удалены и перейдут на один уровень выше в дереве сервисов (дочерние сервисы 1-го уровня получат тот же уровень, что и удаленный родительский сервис).

Две кнопки под списком сервисов предлагают некоторые варианты массового редактирования:

- *Mass update* - массовое обновление свойств сервиса
- *Delete* - удаление сервисов

Чтобы использовать эти параметры, отметьте флажки перед соответствующими сервисами, затем нажмите нужную кнопку.

**2. SLA**

**Обзор** После создания услуг вы можете начать мониторинг их производительности в соответствии с соглашением об уровне услуг (SLA).

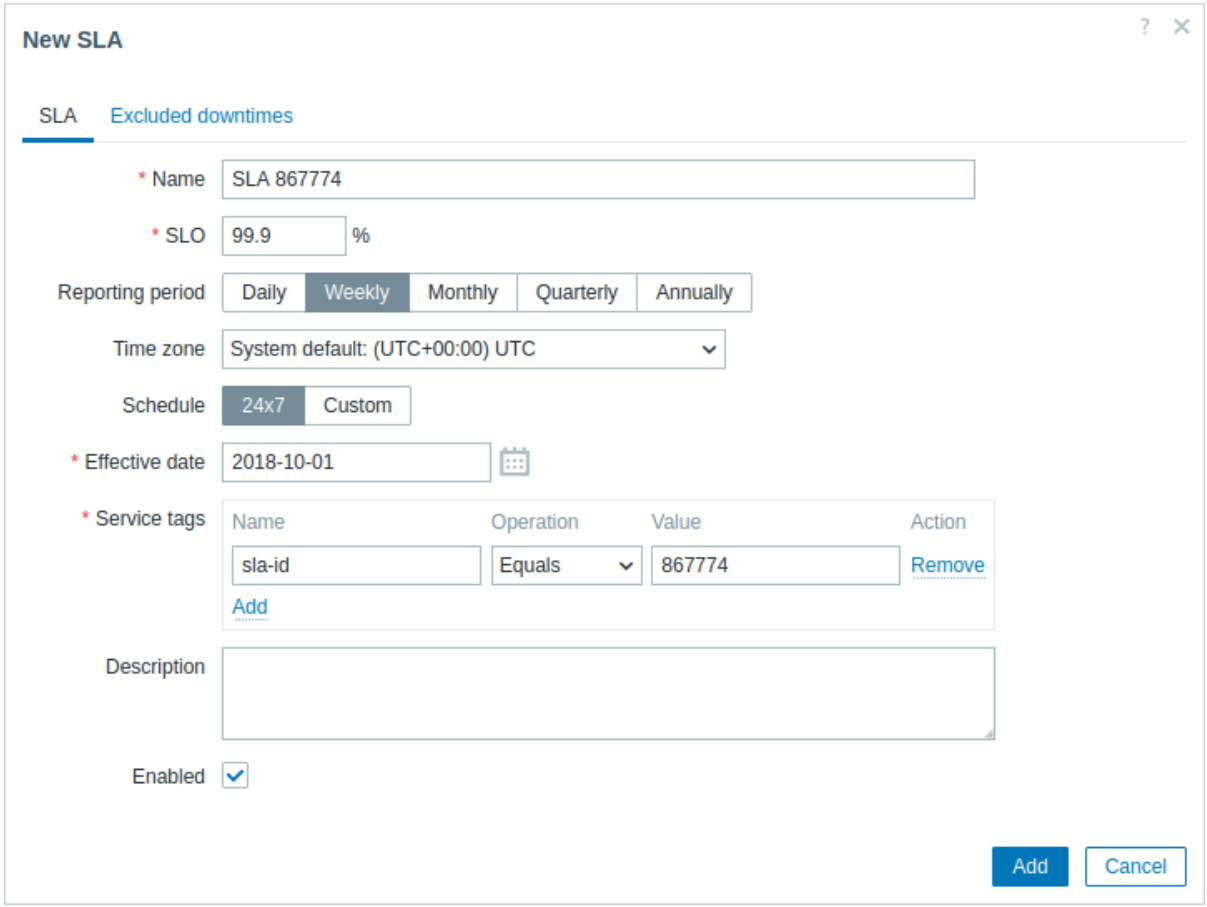
Раздел меню *Услуги->SLA (Services->SLA)* позволяет настроить SLA по различным услугам. SLA в Zabbix определяет цель уровня обслуживания (SLO), расписание ожидаемого времени работы и запланированного времени простоя.

SLA и услуги сопоставляются при помощи тегов услуг. К нескольким услугам можно применить один и тот же SLA — производительность будет измеряться по каждой соответствующей услуге отдельно. Одна услуга может иметь несколько назначенных SLA — данные по каждому SLA будут отображаться отдельно.

В отчётах SLA Zabbix отображает данные Индикатора уровня услуги (SLI), который замеряет фактическую доступность услуги. Соответствие услуги целевому показателю SLA определяется сравнением SLO (ожидаемая доступность в %) с SLI (фактическая доступность в %).

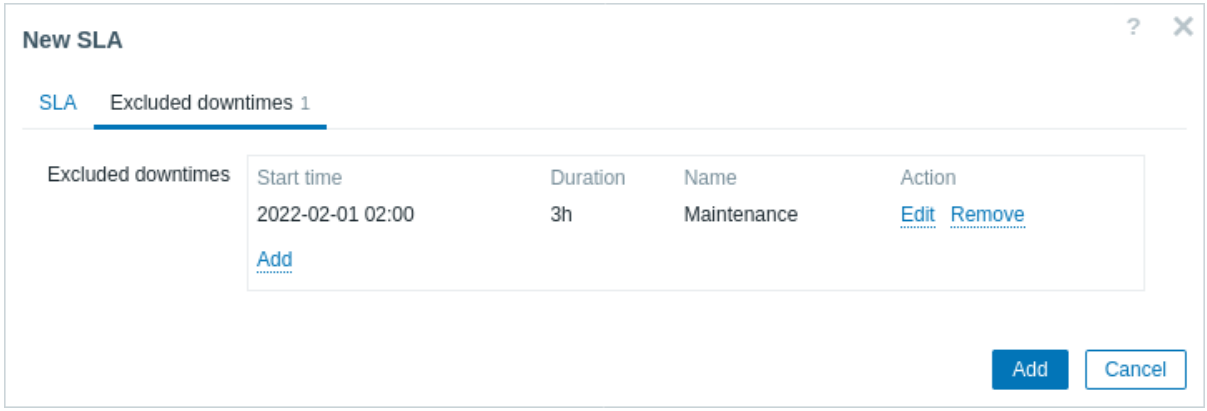
**Конфигурация** Чтобы создать новый SLA, нажмите кнопку *Create SLA*.

Вкладка **SLA** позволяет указать общие параметры SLA.




| Parameter | Description |
|---|---|
| *Name* | Введите имя SLA. |
| *SLO* | Введите целевой уровень обслуживания (SLO) в процентах. |
| *Reporting period* | Выберите период отчетности SLI, используемый для расчета данных SLI в отчете SLA:<br>**Daily** - каждый день, с 00:00:00 до 23:59:59.<br>**Weekly** - каждую неделю, с воскресенья 00:00:00 до субботы 23:59:59.<br>**Monthly** - каждый месяц, с первого дня 00:00:00 до последнего дня 23:59:59.<br>**Quarterly** - каждый календарный квартал: Q1 (янв.-мар.), Q2 (апр.-июн.), Q3 (июл.-сен.), Q4 (окт.-дек.).<br>**Annually** - каждый календарный год, с 1 января 00:00:00 до 31 декабря 23:59:59. |
| *Time zone* | Выберите часовой пояс SLA. |
| *Schedule* | Выберите расписание SLA - 24x7 или пользовательское. |
| *Effective date* | Выберите дату начала расчета SLA. |
| *Service tags* | Добавьте теги сервиса, чтобы определить сервисы, к которым следует применять этот SLA.<br>**Name** - имя тега сервиса, должно точно совпадать, с учетом регистра.<br>**Operation** - выберите *Equals*, если значение тега должно совпадать точно (с учетом регистра), или *Contains*, если должна совпадать часть значения тега (без учета регистра).<br>**Value** - значение тега сервиса для поиска в соответствии с выбранной операцией.<br>SLA применяется к сервису, если совпадает хотя бы один тег сервиса. |
| *Description* | Добавьте описание SLA. |
| *Enabled* | Установите флажок, чтобы включить расчет SLA. |

Вкладка **Excluded downtimes** позволяет указать периоды недоступности, которые исключаются из расчета SLA.




Нажмите *Add*, чтобы настроить исключенные периоды недоступности, затем введите имя периода, дату начала и длительность.

**Отчеты SLA** Как сервис выполняет SLA, можно увидеть в отчете SLA. Отчеты SLA можно просматривать:

- в разделе *SLA*, щелкнув по гиперссылке отчета SLA;
- в разделе *Services*, щелкнув по имени SLA на вкладке информации;
- в виджете *SLA report* на панели Dashboard.

После настройки SLA вкладка *Info* в разделе services также будет отображать некоторую информацию о производительности сервиса.

### 3. Пример настройки

**Обзор** В этом разделе описана простая настройка мониторинга кластера высокой доступности Zabbix как услуги.

**Предварительные требования** Перед тем как настраивать мониторинг услуг, вам необходимо настроить следующие узлы сети:

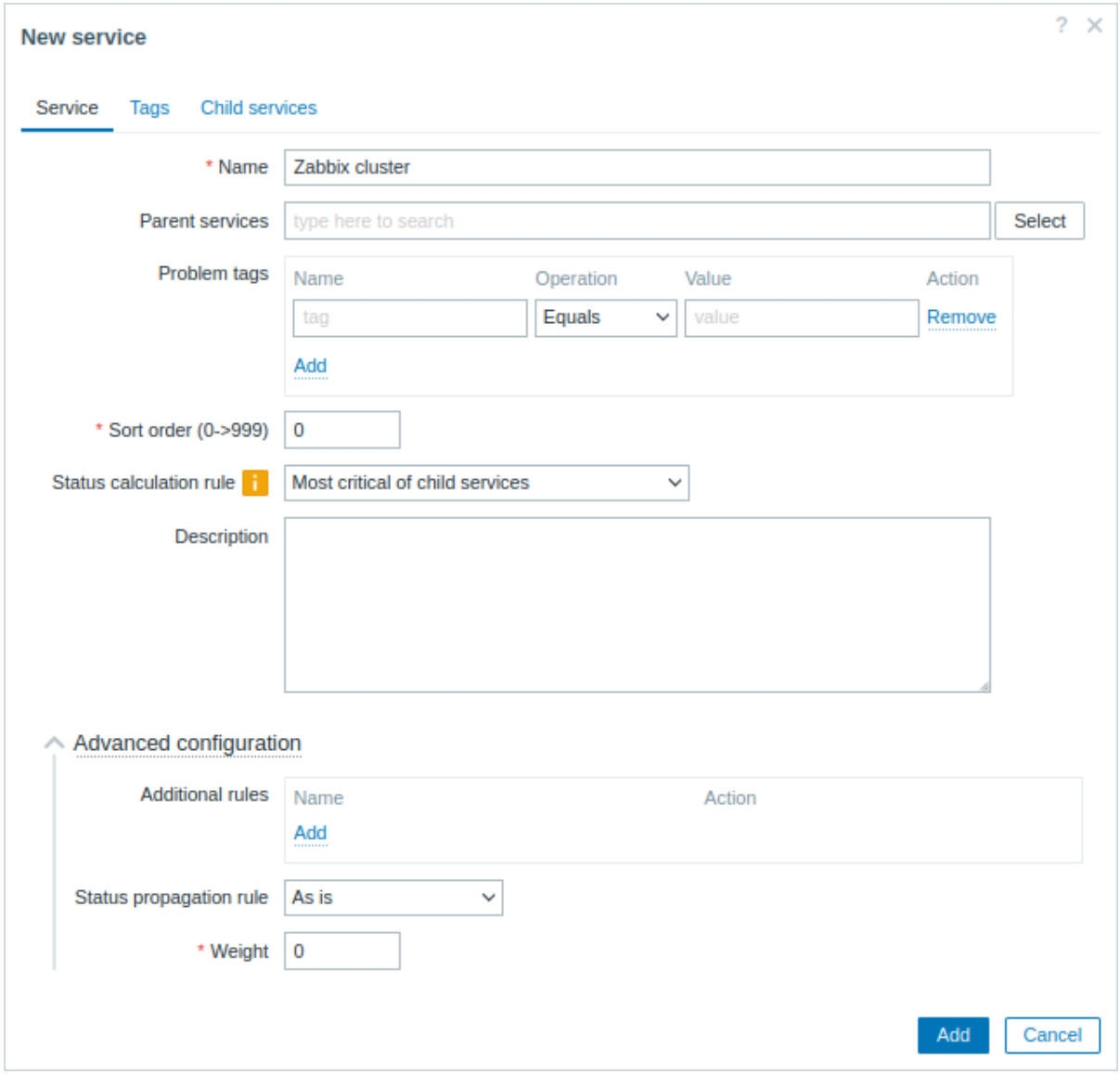
- *HA нода 1* с по крайней мере одним триггером и тегом (желательно задать на уровне триггера) `component:ha-node-1`
- *HA нода 2* с по крайней мере одним триггером и тегом (желательно задать на уровне триггера) `component:ha-node-2`

**Дерево сервисов** Следующий шаг — построить дерево сервисов. В этом примере инфраструктура очень проста и состоит из трех сервисов: *Zabbix cluster* (родительский) и двух дочерних сервисов *Zabbix server node 1* и *Zabbix server node 2*.

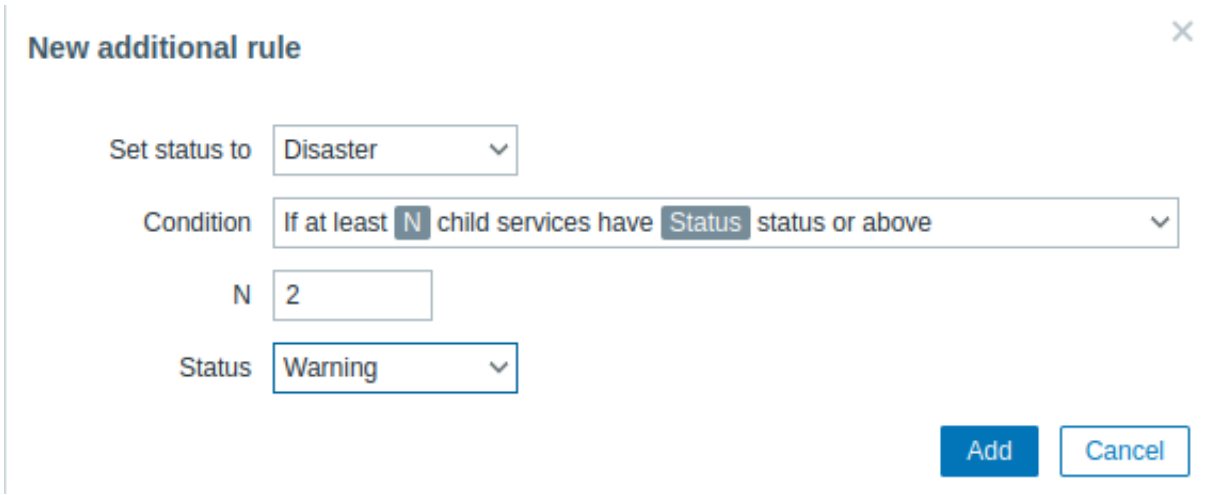
```
Zabbix cluster
|
|- Zabbix server node 1
|- Zabbix server node 2
```

На странице Services включите режим *Edit* и нажмите *Create service*:

В окне настройки сервиса введите имя *Zabbix cluster* и нажмите заголовок *Advanced configuration*, чтобы отобразить дополнительные параметры настройки.




Настройте дополнительное правило:

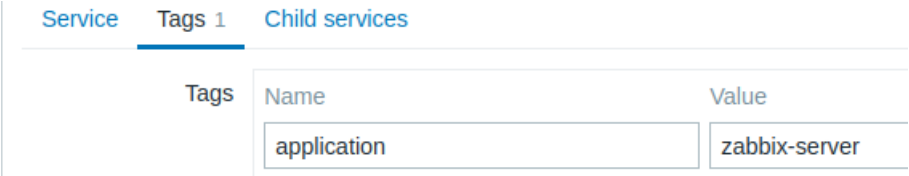



У *Zabbix cluster* будет два дочерних сервиса — по одному для каждого из узлов HA. Если у обоих узлов HA возникнут проблемы со статусом не ниже *Warning*, статус родительского сервиса должен быть установлен в *Disaster*. Чтобы добиться этого, дополнительное правило следует настроить так:

- Set status to: Disaster
- Condition: If at least `N` child services have `Status` status or above
- N: 2
- Status: Warning

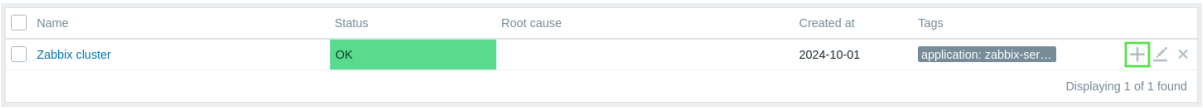
Перейдите на вкладку *Tags* и добавьте тег `application:zabbix-server`. Этот тег будет использоваться позже для

действий сервиса и отчетов SLA.




Сохраните новый сервис.

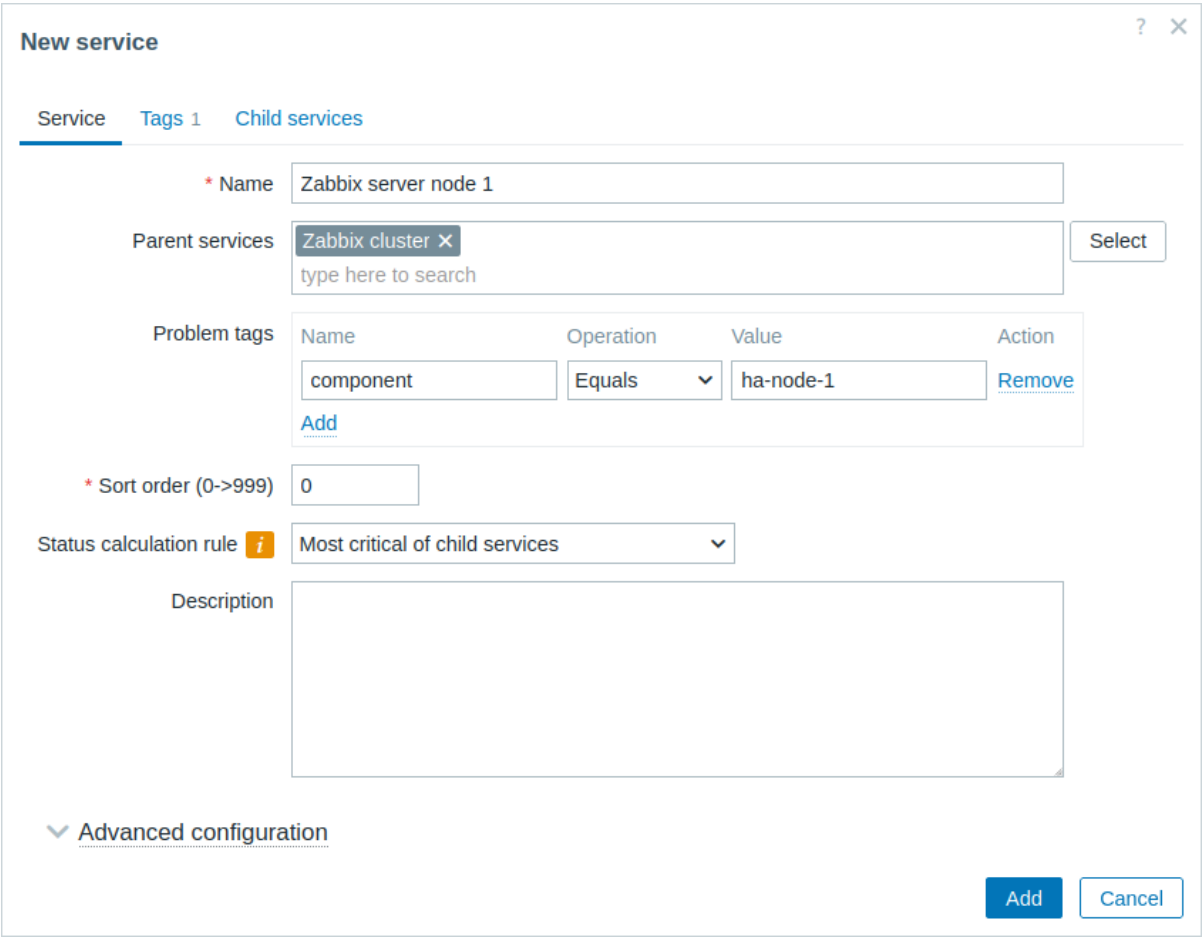
Чтобы добавить дочернюю службу, нажмите значок плюса рядом со службой кластера Zabbix (значок виден только в режиме *Edit*).



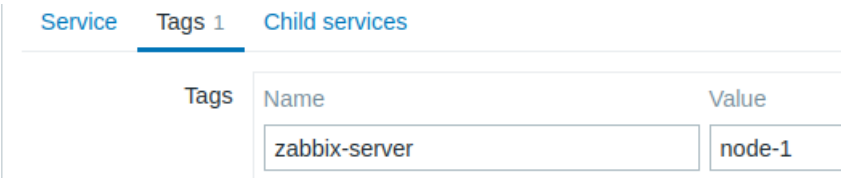

В окне настройки службы введите имя *Zabbix server node 1*. Обратите внимание, что параметр Parent services уже предварительно заполнен значением *Zabbix cluster*.

Доступность этой службы зависит от проблем на узле сети *HA node 1*, отмеченных тегом проблемы `component:ha-node-1`. В параметре Problem tags введите:

- Name: component
- Operation: Equals
- Value: ha-node-1




Перейдите на вкладку *Tags* и добавьте тег службы: `zabbix-server:node-1`. Этот тег будет использоваться позже для действий со службами и отчетов SLA.



Сохраните новую службу.

Создайте еще одну дочернюю для Zabbix кластера услугу с именем «Zabbix сервер нода 2».

Задайте теги проблем следующим образом:

- Имя: component
- Операция: Равно
- Значение: ha-node-2

Переключитесь на вкладку *Теги* и добавьте тег услуги: `zabbix-server:node-2`.

Сохраните новую услугу.

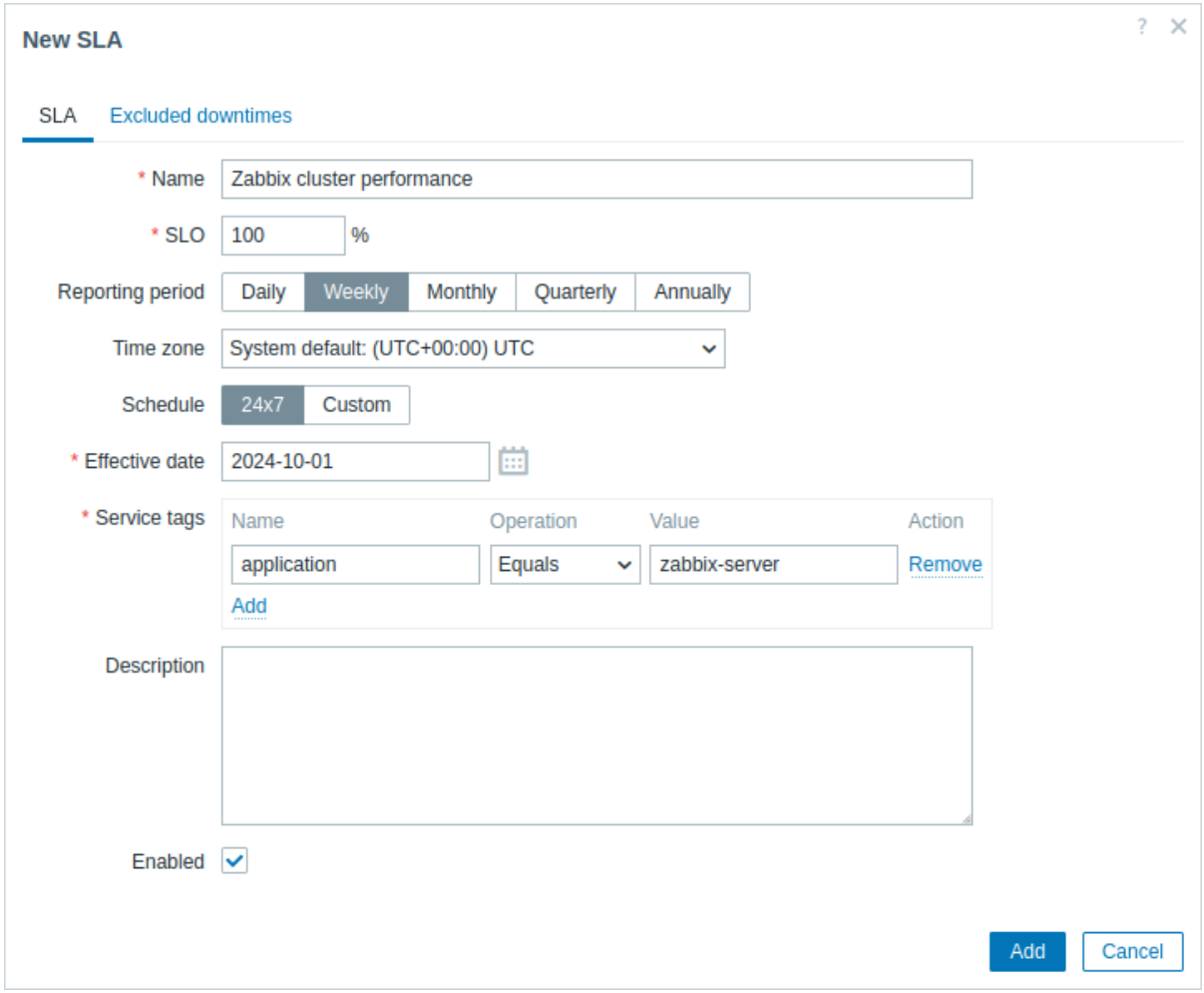
**SLA** В этом примере ожидаемая производительность кластера Zabbix составляет 100% без учета полугодового периода обслуживания длительностью один час.

Сначала необходимо добавить новое соглашение об уровне обслуживания.

Перейдите в раздел меню *Services->SLA* и нажмите *Create SLA*. Введите имя *Zabbix cluster performance* и установите SLO на 100%.

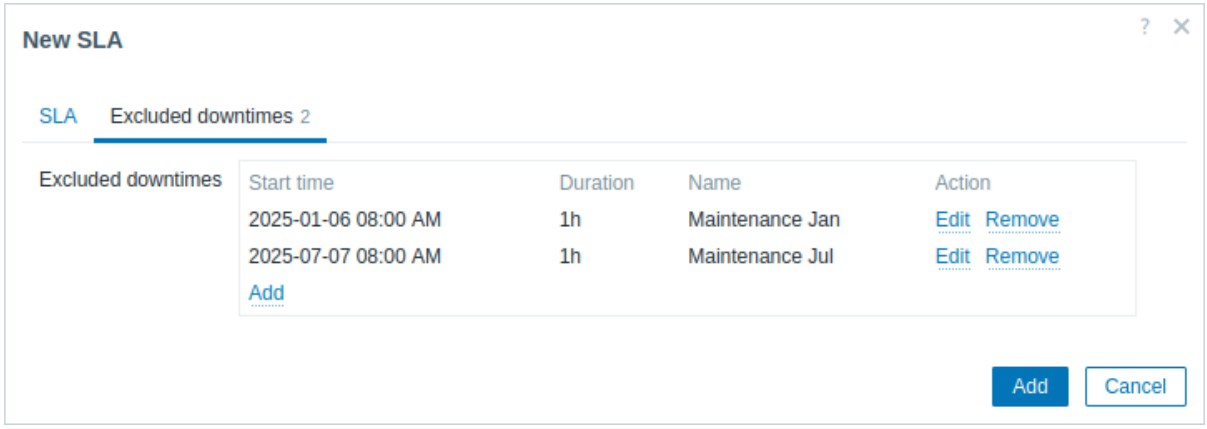
Служба Zabbix cluster имеет тег службы `application:zabbix-server`. Чтобы использовать этот SLA для измерения производительности кластера Zabbix, в параметре *Service tags* укажите:

- Name: application
- Operation: Equals
- Value: zabbix-server



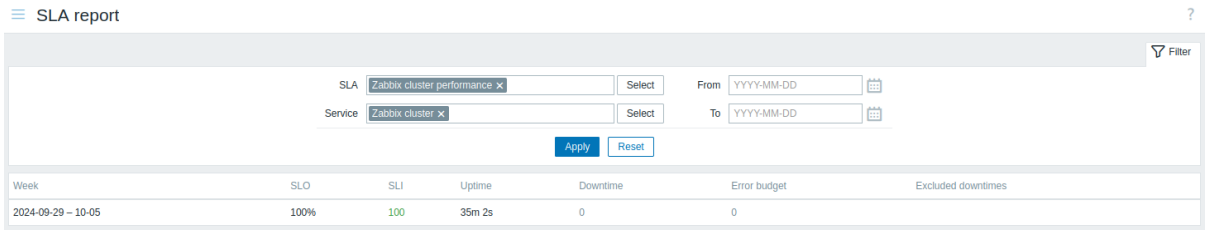

В реальной среде также можно изменить требуемый отчетный период, часовой пояс и дату начала либо сменить расписание с 24/7 на пользовательское. Для этого примера настройки по умолчанию достаточны.

Перейдите на вкладку *Excluded downtimes* и добавьте периоды простоя для запланированного обслуживания, чтобы исключить эти периоды из расчета SLA. В разделе *Excluded downtimes* нажмите *Add link*, введите имя простоя, запланированное время начала и длительность.

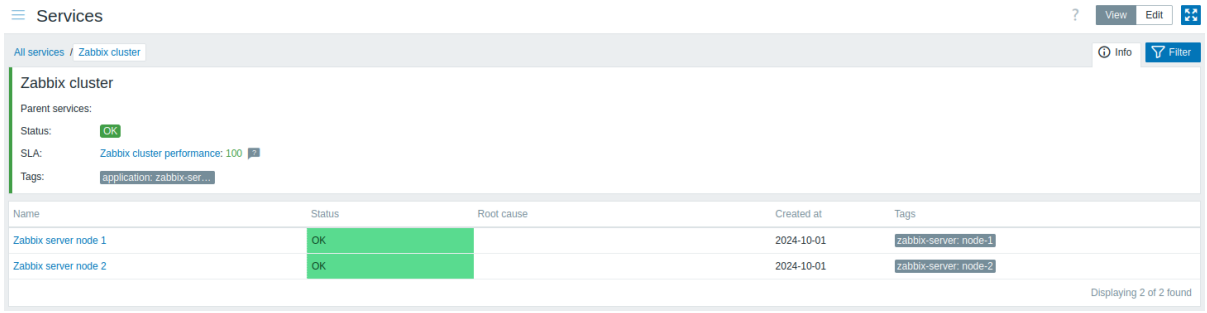



Нажмите *Add*, чтобы сохранить новый SLA.

Перейдите в раздел отчетов SLA, чтобы просмотреть отчет SLA для кластера Zabbix.




Информацию о SLA также можно проверить в разделе *Services*.




## 9. Веб-мониторинг

**Обзор** Благодаря Zabbix вы можете проверять несколько аспектов доступности веб-сайтов.

> **Attention:**
> Для выполнения веб-мониторинга Zabbix сервер должен быть изначально сконфигурирован с поддержкой cURL (libcurl).

Для активации веб-мониторинга вам необходимо определить веб-сценарии. Веб-сценарий состоит из одного или нескольких запросов HTTP, или «шагов». Шаги периодически выполняются Zabbix сервером в предопределённом порядке. Если узел сети наблюдается через прокси, тогда шаги выполняются этим прокси.

Веб-сценарии привязываются к узлам сети/шаблонам тем же образом, как элементы данных, триггеры и т.д. Это означает, что веб-сценарии можно создавать на уровне шаблона и далее применять к нескольким узлам сети одним движением.

Каждым веб-сценарием собирается следующая информация:

- средняя скорость загрузки в секунду для всех шагов для всего сценария
- номер шага, который завершился с ошибкой
- последнее сообщение об ошибке

На каждом шаге веб-сценария собирается следующая информация:

- скорость загрузки в секунду
- время ответа

- код ответа

Для получения более детальных сведений смотрите элементы данных веб мониторинга.

Собранные данные с выполненных веб-сценариев хранятся в базе данных. Эти данные автоматически используются для графиков, триггеров и оповещений.

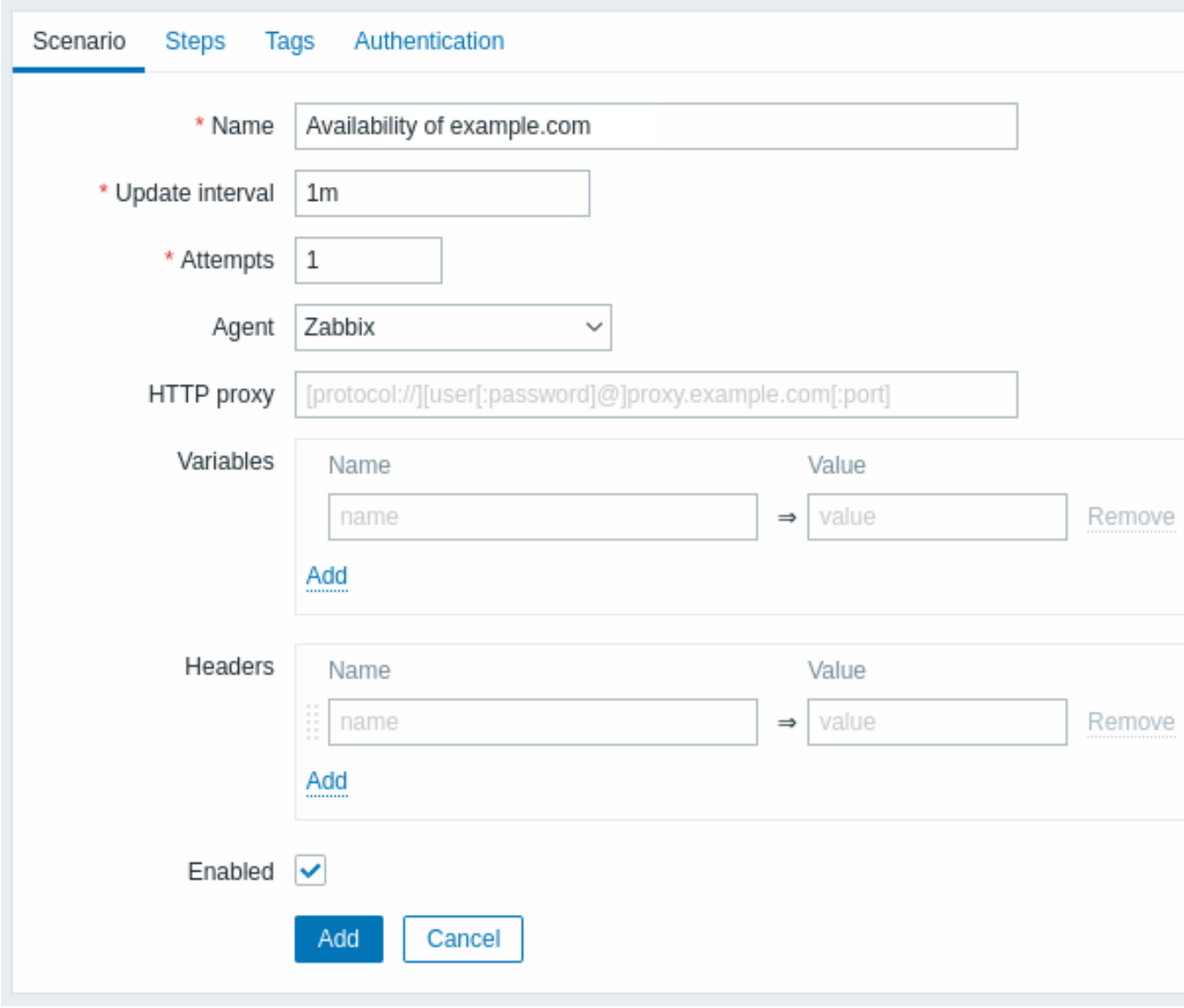
Zabbix может также проверять, содержит ли полученная HTML страница заданную строку. Он может выполнить эмуляцию входа и следовать пути, эмулируя нажатия мышкой на странице.

Веб-мониторинг в Zabbix поддерживает и HTTP, и HTTPS. При выполнении веб-сценария, Zabbix сервер может следовать перенаправлениям (смотрите опцию *Следовать перенаправлениям* ниже). Максимальное количество перенаправлений жёстко задано в исходном коде и равно 10 (используется cURL опция CURLOPT_MAXREDIRS [en]). Все cookies запоминаются на протяжении выполнения одного сценария.

**Настройка веб-сценария** Чтобы настроить веб-сценарий:

- Перейдите в: *Сбор данных* → *Узлы сети* (или *Шаблоны*)
- Нажмите *Web* в строке узла сети/шаблона
- Нажмите *Создать веб-сценарий* справа (или на имени сценария, чтобы изменить существующий сценарий)
- Введите параметры сценария в форме

Вкладка **Сценарий** позволяет настроить общие параметры веб-сценария.




Все обязательные поля ввода отмечены красной звездочкой.

Параметры сценария:

| Parameter | Description |
|---|---|
| *Name* | Уникальное имя сценария.<br>Поддерживаются пользовательские макросы. *Note*: если используются пользовательские макросы, эти макросы останутся неразрешенными в именах элементов данных веб-мониторинга. |
| *Update interval* | Как часто будет выполняться сценарий.<br>Поддерживаются суффиксы времени, например 30s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы. *Note*: если используется пользовательский макрос и его значение изменяется (например, 5m → 30s), следующий опрос будет выполнен в соответствии с предыдущим значением (в примере — позже).<br>Новые веб-сценарии будут проверены в течение 60 секунд после создания. |
| *Attempts* | Количество попыток выполнения шагов веб-сценария. В случае сетевых проблем (тайм-аут, отсутствие подключения и т. д.) Zabbix может повторить выполнение шага несколько раз. Указанное значение одинаково влияет на каждый шаг сценария. Можно указать до 10 попыток, значение по умолчанию — 1.<br>*Note*: Zabbix не будет повторять шаг из-за неверного кода ответа или несоответствия требуемой строки. |
| *Agent* | Выберите клиентский агент.<br>Zabbix будет выдавать себя за выбранный браузер. Это полезно, когда веб-сайт возвращает разное содержимое для разных браузеров.<br>В этом поле можно использовать пользовательские макросы. |
| *HTTP proxy* | Можно указать HTTP-прокси для использования в формате `[protocol://][username[:password]@]proxy.example.com[:port]`.<br>Это задает параметр cURL CURLOPT_PROXY.<br>Необязательный префикс `protocol://` можно использовать для указания альтернативных протоколов прокси (поддержка префикса протокола была добавлена в cURL 7.21.7). Если протокол не указан, прокси будет считаться HTTP-прокси.<br>По умолчанию будет использоваться порт 1080.<br>Если указано, прокси переопределит связанные с прокси переменные окружения, такие как http_proxy, HTTPS_PROXY. Если не указано, прокси не будет переопределять связанные с прокси переменные окружения. Введенное значение передается ”как есть”, без проверки корректности.<br>Также можно указать адрес SOCKS-прокси. Если указать неверный протокол, соединение завершится ошибкой, и элемент данных станет неподдерживаемым.<br>*Note*: с HTTP-прокси поддерживается только простая аутентификация.<br>В этом поле можно использовать пользовательские макросы. |
| *Variables* | Переменные, которые можно использовать в шагах сценария (URL, переменные POST).<br>Они имеют следующий формат:<br>**{macro1}**=value1<br>**{macro2}**=value2<br>**{macro3}**=regex:<regular expression><br>**{macro4}**=jsonpath:<jsonpath><br>**{macro5}**=xmlxpath:<xmlxpath><br>**{macro6}**={{macro}.function()} (см. функции макросов)<br>Например:<br>{username}=Alexei<br>{password}=kj3h5kJ34bd<br>{hostid}=regex:hostid is ([0-9]+)<br>{url}=jsonpath:$.host_url<br>{status}=xmlxpath://host/response/status<br>{newvar}={{myvar}.btoa()}<br>Затем на макросы можно ссылаться в шагах как на {username}, {password}, {hostid} и т. д. Zabbix автоматически заменит их фактическими значениями. Обратите внимание, что для переменных с `regex:` требуется один шаг, чтобы получить значение регулярного выражения, поэтому извлеченное значение можно применить только к следующему шагу.<br>Если часть значения начинается с `regex:`, то часть после нее рассматривается как регулярное выражение, которое выполняет поиск на веб-странице и, если совпадение найдено, сохраняет его в переменной. Должна присутствовать как минимум одна подгруппа, чтобы можно было извлечь совпавшее значение.<br>Поддерживаются пользовательские макросы и макросы {HOST.*}.<br>Переменные автоматически URL-кодируются при использовании в полях запроса или данных формы для переменных POST, но должны кодироваться вручную при использовании в raw post или непосредственно в URL. |

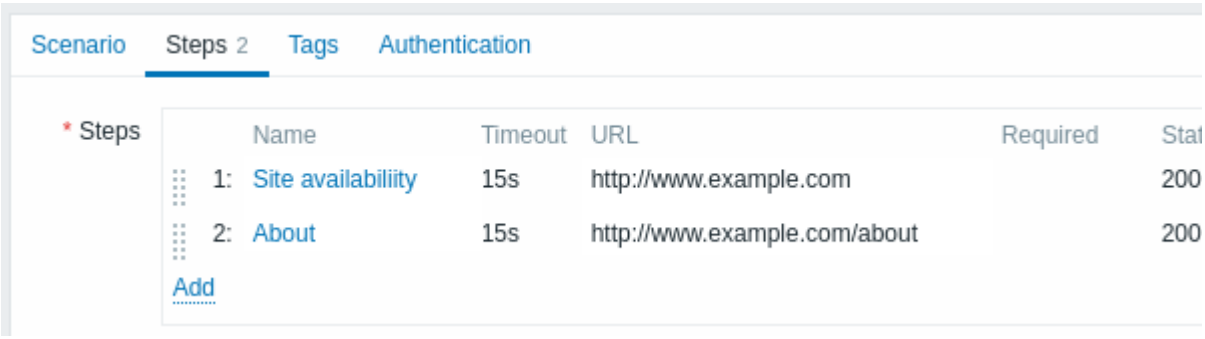
| Parameter | Description |
|---|---|
| *Headers* | HTTP-заголовки используются при выполнении запроса. Можно использовать стандартные и пользовательские заголовки.<br>Заголовки будут назначаться с использованием настроек по умолчанию в зависимости от типа Agent, выбранного из раскрывающегося списка на уровне сценария, и будут применяться ко всем шагам, если только они не заданы как пользовательские на уровне шага.<br>**Следует отметить, что задание заголовка на уровне шага автоматически отменяет все ранее заданные заголовки, кроме стандартного заголовка, который назначается путем выбора 'User-Agent' из раскрывающегося списка на уровне сценария.**<br>Однако даже стандартный заголовок 'User-Agent' можно переопределить, указав его на уровне шага.<br>Чтобы снять заголовок на уровне сценария, заголовок следует назвать и указать без значения на уровне шага.<br>Заголовки следует перечислять, используя тот же синтаксис, что и в HTTP-протоколе, при необходимости с использованием дополнительных возможностей, поддерживаемых параметром cURL CURLOPT_HTTPHEADER.<br>Например:<br>Accept-Charset=utf-8<br>Accept-Language=en-US<br>Content-Type=application/xml; charset=utf-8<br>Поддерживаются пользовательские макросы и макросы {HOST.*}. |
| *Enabled* | Сценарий активен, если этот флажок установлен, в противном случае — отключен. |

Обратите внимание, что при редактировании существующего сценария в форме доступны две дополнительные кнопки:

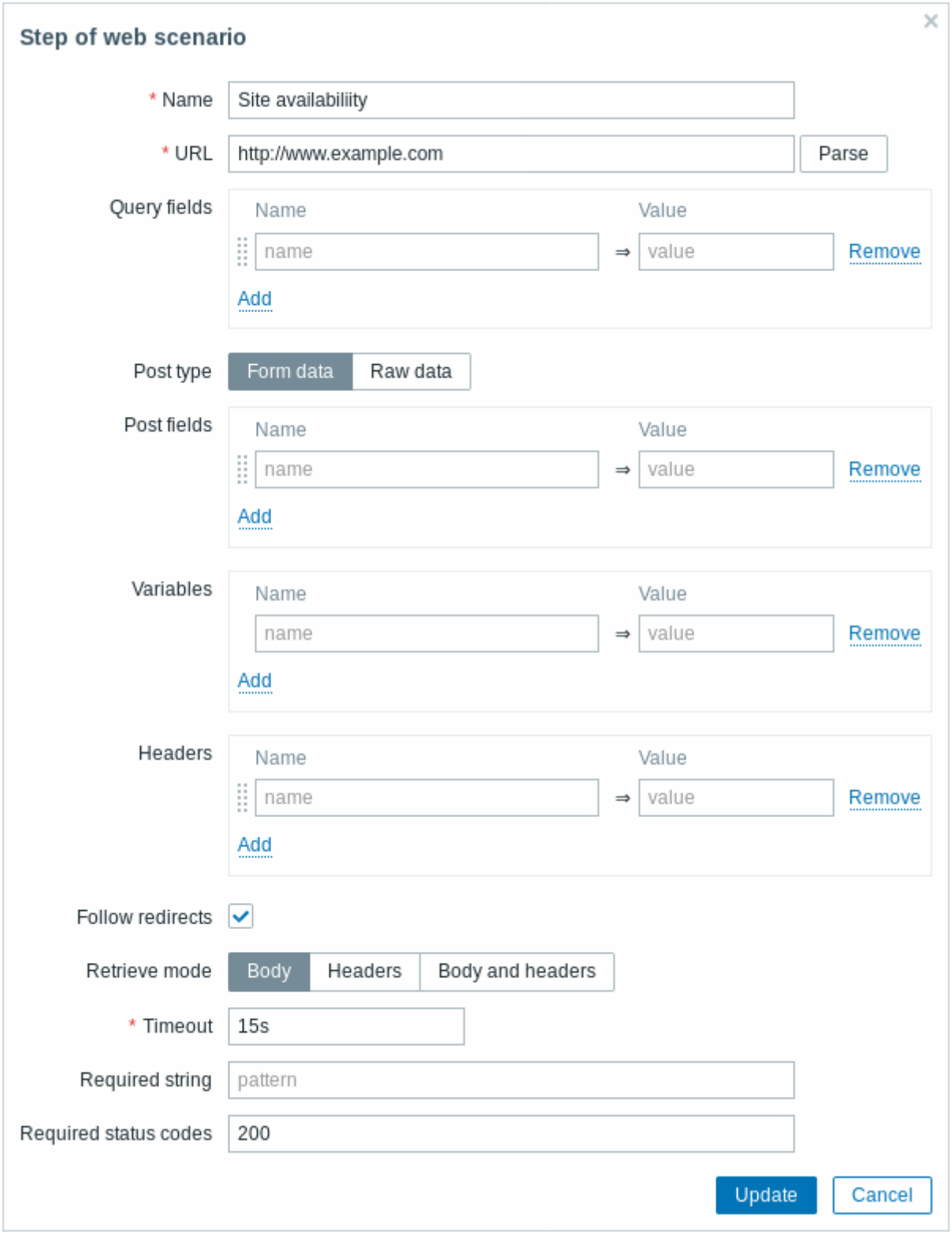
| | |
|---|---|
| Clone | Создать еще один сценарий на основе свойств существующего. |
| Clear history and trends | Удалить историю и данные трендов для сценария. Это заставит сервер немедленно выполнить сценарий после удаления данных. |

**Note:**
Если поле *HTTP proxy* оставить пустым, другой способ использовать HTTP-прокси — задать связанные с прокси переменные окружения.
Для HTTP-проверок — задайте переменную окружения **http_proxy** для пользователя сервера Zabbix. Например, `http_proxy=http://proxy_ip:proxy_port`.
Для HTTPS-проверок — задайте переменную окружения **HTTPS_PROXY**. Например, `HTTPS_PROXY=http://proxy_ip:proxy_port`. Дополнительные сведения можно получить, выполнив команду оболочки: *# man curl*.

Вкладка **Шаги** позволяет настроить шаги веб-сценария. Чтобы добавить шаг веб-сценария, нажмите *Добавить* в блоке *Шаги*.




**Note:**
Секретные пользовательские макросы нельзя использовать в URL, так как они будут преобразованы в "******".




**Настройка шагов**

Параметры шага:

| Параметр | Описание |
|---|---|
| *Имя (Name)* | Уникальное имя шага.<br>Поддерживаются пользовательские макросы. *Обратите внимание* что если используется пользовательские макросы, то эти макросы останутся нераскрытыми в именах элементов данных веб-мониторинга. |

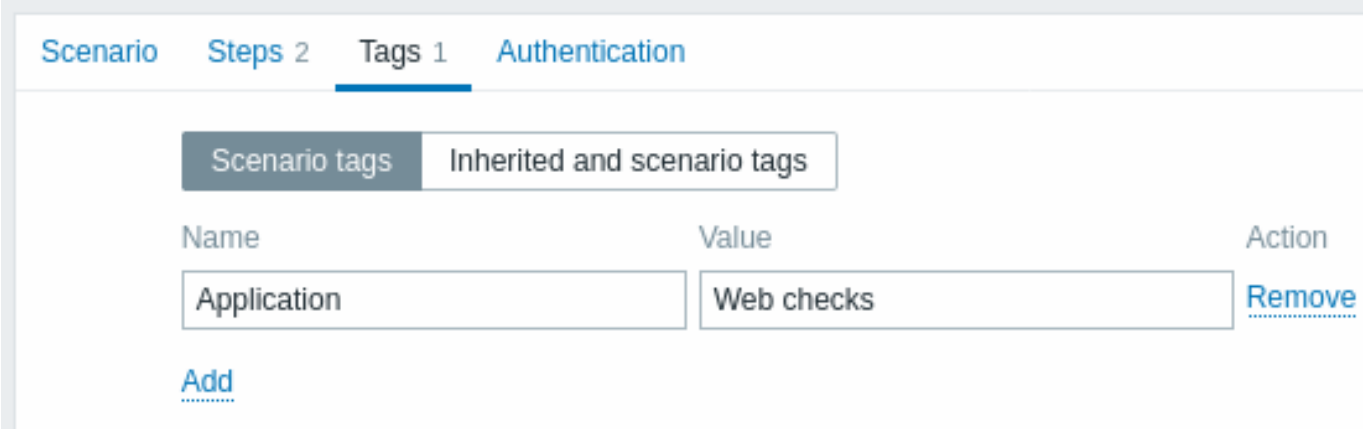
| Параметр | Описание |
|---|---|
| *URL* | URL для подключения и получения данных. Например:<br>https://www.example.com<br>http://www.example.com/download<br>Имена доментов можно указывать Юникод символами. Они автоматически конвертируются методом punycode в ASCII при выполнении шага веб-сценария.<br>Кнопку *Анализ* можно использовать, чтобы отделить из URL опциональные поля запроса (например, ?name=Admin&password=mypassword), переместив атрибуты и значения в *Поля запроса*, чтобы URL кодировка выполнялась автоматически.<br>Переменные можно использовать в URL, воспользовавшись синтаксисом {макрос}.<br>Переменные можно URL кодировать вручную, используя {{macro}.urlencode()} синтаксис.<br>Поддерживаются пользовательские макросы и макросы {HOST.*}.<br>Ограничено 2048 символами. |
| *Поля запроса (Query fields)* | Переменные HTTP GET для URL.<br>Задаются в виде пар атрибутов и значений.<br>Значения URL кодируются автоматически. Значения из переменных сценария, пользовательских макросов или макросов {HOST.*} раскрываются и затем URL кодируются автоматически. При использовании {{макрос}.urlencode()} синтаксиса будет выполнена двойная URL кодировка этих переменных.<br>Поддерживаются пользовательские макросы и {HOST.*} макросы. |
| *Post* | Переменные HTTP POST.<br>В режиме **Данные формы** задаются в виде пар атрибутов и значений.<br>Значения URL кодируются автоматически. Значения из переменных сценария, пользовательских макросов или {HOST.*} макросов раскрываются и затем URL кодируются автоматически.<br>В режиме **Сырые данные** атрибуты/значения отображаются в одной строке и объединяются через символ **&**.<br>Сырые значения можно URL кодировать/декодировать вручную, используя синтаксис {{макрос}.urlencode()} или {{макрос}.urldecode()}.<br>Например: id=2345&userid={user}<br>Если {user} задан переменной на уровне веб-сценария, он будет заменён своим значением при выполнении шага. Если вы желаете URL кодировать значение, замените {user} на {{user}.urlencode()}.<br>Поддерживаются пользовательские макросы и макросы {HOST.*}. |
| *Переменные (Variables)* | Список переменных на уровне шага, которые можно использовать в GET и POST функциях.<br>Задаются в виде пар атрибутов и значений.<br>Переменные на уровне шага переопределяют переменные уровня сценария и переменные из предыдущих шагов. Однако, значение переменной уровня шага влияет только на последующие шаги (а не на текущий шаг).<br>Такие переменные имеют следующий формат:<br>**{макрос}**=значение<br>**{макрос}**=regex:<регулярное выражение><br>Более подробную информацию вы найдете в описании переменных уровня сценария.<br>Переменные автоматически URL кодируются, когда используются в полях запросов или в данных формы для переменных post, но их необходимо вручную URL кодировать, когда они используются в сыром post или напрямую в URL. |
| *Заголовки (Headers)* | Пользовательские HTTP заголовки, которые будут отправлены при выполнении запроса.<br>Задаются в виде пар атрибутов и значений.<br>Заголовок, заданный на уровне шага, будет использоваться для этого конкретного шага.<br>**Следует заметить, что при задании заголовка на уровне шага автоматически уничтожаются все ранее заданные заголовки, за исключением заголовков «User-Agent» по умолчанию, назначенных при выборе из выпадающего списка на уровне сценария.**<br>Тем не менее, даже заголовок «User-Agent» по умолчанию может быть переопределён путём назначения ему значения на уровне шага.<br>Например, если указать имя заголовка без указания значения, то это удалит заголовок по умолчанию, указанный на уровне сценария.<br>Поддерживаются пользовательские макросы и макросы {HOST.*}.<br>Это поле задаёт опцию cURL CURLOPT_HTTPHEADER [en]. |
| *Следовать перенаправлениям (Follow redirects)* | Отметьте для следования по HTTP перенаправлениям.<br>Этот параметр использует опцию cURL CURLOPT_FOLLOWLOCATION [en]. |

| Параметр | Описание |
|---|---|
| *Режим получения (Retrieve mode)* | Выберите режим получения:<br>**Тело** — получение только тела из HTTP ответа<br>**Заголовки** — получение только заголовков из HTTP ответа<br>**Тело и заголовки** — получение тела и заголовков из HTTP ответа |
| *Время ожидания (Timeout)* | Zabbix не будет тратить более указанного времени при обработке URL (от 1 секунды до максимум 1 часа). В действительности же этот параметр определяет максимальное время содания подключения к URL и максимальное время для выполнения HTTP запроса.<br>Следовательно, Zabbix не будет тратить более **2 x Время ожидания** секунд на один шаг.<br>Поддерживаются суффиксы времени, например, 30s, 1m, 1h.<br>Поддерживаются пользовательские макросы. |
| *Требуемая строка (Required string)* | Требуемый шаблон регулярных выражений.<br>Если полученное содержимое (HTML) не совпадает с требуемым шаблоном, то шаг будет считаться ошибочным. Если поле не заполнено, то проверка не производится.<br>Например:<br>Homepage of Zabbix<br>Welcome.*admin<br>*Обратите внимание*: Ссылки на регулярные выражения, созданные в веб-интерфейсе Zabbix, в этом поле не поддерживаются.<br>Поддерживаются пользовательские макросы и макросы {HOST.*}. |
| *Требуемые коды состояния (Required status codes)* | Список ожидаемых кодов состояния HTTP. Если Zabbix получает код не из списка, то шаг будет считаться ошибочным.<br>Если поле не заполнено, то проверка не производится.<br>Например: 200,201,210-299.<br>Поддердживаются пользовательские макросы. |

**Note:**
Любые изменения в шагах веб-сценариев будут сохранены, только если сам сценарий был также сохранён.

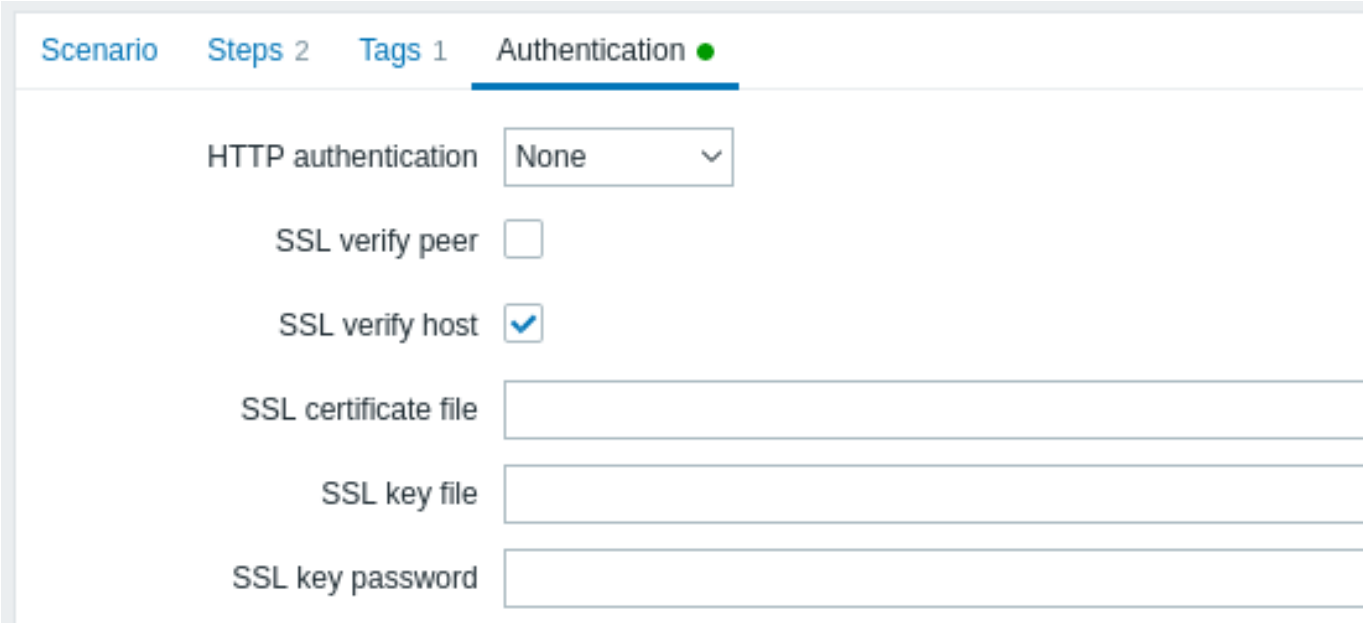
Смотрите также пример из реальной жизни о том, как можно настроить шаги веб-мониторинга.

**Настройка тегов** Вкладка **Теги (Tags)** позволяет задать теги уровня сценария.




Теги позволяют фильтровать веб-сценарии и элементы данных веб-мониторинга.

**Настройка аутентификации** Вкладка **Аутентификация (Authentication)** позволяет вам настроить опции аутентификации сценария. Зелёная точка в имени вкладки означает, что включён какой-то тип HTTP аутентификации.




Параметры аутентификации:

| Параметр | Описание |
|---|---|
| *HTTP аутентификация (HTTP authentication)* | Выберите опцию аутентификации.<br>**Нет** — без использования аутентификации.<br>**Простая аутентификация** — с использованием простой (basic) аутентификации.<br>**NTLM аутентификация** — с использованием NTLM (Windows NT LAN Manager) аутентификации.<br>**Kerberos** — с использованием Kerberos аутентификации. Смотрите также: Настройка Kerberos с Zabbix.<br>**Digest** — с использованием Digest аутентификации.<br>После выбора какого-либо метода аутентификации будут доступны доступны два дополнительных поля для ввода имени пользователя и пароля.<br>В полях имя пользователя и пароль можно использовать пользовательские макросы. |
| *Пользователь (User)* | Введите имя пользователя (до 255 символов).<br>Это поле доступно, если параметр *HTTP аутентификация* установлен в значение Basic, NTLM, Kerberos или Digest. Поддерживаются пользовательские макросы. |
| *Пароль (Password)* | Введите пароль пользователя (до 255 символов).<br>Это поле доступно, если параметр *HTTP аутентификация* установлен в значение Basic, NTLM, Kerberos или Digest. Поддерживаются пользовательские макросы. |
| *Проверка SSL узла (SSL verify peer)* | Отметьте для верификации SSL сертификата веб-сервера.<br>Сертификат сервера будет автоматически взят из местоположения центра сертификации (CA) всей системы. Вы можете переопределить расположение CA файлов, используя параметр конфигурации SSLCALocation Zabbix сервера или прокси.<br>Этот параметр использует опцию cURL CURLOPT_SSL_VERIFYPEER [en]. |
| *Проверка SSL хоста (SSL verify host)* | Отметьте для верификации, что поле *Common Name* или поле *Subject Alternate Name* сертификата веб-сервера совпадают.<br>Этот параметр использует cURL опцию CURLOPT_SSL_VERIFYHOST [en]. |
| *Файл SSL сертификата (SSL certificate file)* | Имя файла SSL сертификата для аутентификации клиента. Файл сертификата должен быть в формате PEM[1]. Если файл сертификата также содержит и закрытый ключ (private key), оставьте поле *Файл SSL ключа* пустым. Если ключ зашифрован, укажите пароль в поле *Пароль к SSL ключу*. Папка, содержащая этот файл, указывается в параметре конфигурации SSLCertLocation Zabbix сервера и прокси.<br>В этом поле можно использовать макросы `HOST.*` и пользовательские макросы.<br>Этот параметр использует опцию cURL CURLOPT_SSLCERT [en]. |
| *Файл SSL ключа (SSL key file)* | Имя файла закрытого ключа (private key) SSL, который используется для аутентификации клиента. Файл закрытого ключа должен быть в формате PEM[1]. Папка, содержащая этот файл, указывается в параметре конфигурации SSLKeyLocation Zabbix сервера и прокси.<br>В этом поле можно использовать макросы `HOST.*` и пользовательские макросы.<br>Этот параметр использует опцию cURL CURLOPT_SSLKEY [en]. |

| Параметр | Описание |
|---|---|
| *Пароль к SSL ключу (SSL key password)* | Пароль к файлу закрытого ключа.<br>В этом поле можно использовать пользовательские макросы.<br>Этот параметр использует cURL опцию CURLOPT_KEYPASSWD [en]. |

**Attention:**
[1] Zabbix поддерживает файлы сертификатов и закрытых ключей только в формате PEM. В случае, если у вас имеются данные сертификата и закрытого ключа в формате файла PKCS #12 (обычно используется с расширением *.p12 или *.pfx), вы можете сгенерировать из них PEM файл, используя следующие команды:

```
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:**
Zabbix сервер подхватывает изменения в сертификатах без перезапуска.

**Note:**
Если сертификат клиента и закрытый ключ (private key) находятся в одном файле, просто укажите его в поле «Файл SSL сертификата» и оставьте поле «Файл SSL ключа» пустым. Сертификат и ключ всё ещё должны быть в формате PEM. Очень просто объединить сертификат и ключ:

```
cat client.crt client.key > client.pem
```

**Отображение** Чтобы просмотреть веб-сценарии, настроенные для узла сети, перейдите в *Monitoring* → *Hosts*, найдите узел сети в списке и нажмите гиперссылку *Web* в последнем столбце. Нажмите на имя сценария, чтобы получить подробную информацию.

Обзор веб-сценариев также можно отобразить в *Dashboards* с помощью виджета мониторинга веба.

Последние результаты выполнения веб-сценария доступны в разделе *Monitoring* → *Latest data*.

**Расширенный мониторинг** Иногда необходимо записать в журнал полученное содержимое HTML страницы. Это особенно полезно, если некоторые шаги веб-сценария завершаются с ошибкой. Для этой цели служит уровень отладки 5 (трассировка). Этот уровень можно указать в файлах конфигурации сервера и прокси или использовать опции выполнения административных функций (`-R log_level_increase="http poller,N"`, где N является номером процесса). При условии, что уровень отладки 4 уже задан, расширенный мониторинг можно запустить так, как демонстрируется в следующих примерах:

```
#                          http      :
zabbix_server -R log_level_increase="http poller"

#                            http     :
zabbix_server -R log_level_increase="http poller,2"
```

Если расширенный мониторинг не требуется, его можно остановить, используя опцию `-R log_level_decrease`.

**1. Элементы данных веб-мониторинга**

Обзор

При создании веб-сценариев некоторые новые элементы данных для выполнения мониторинга добавляются автоматически.

Все элементы данных наследуют теги веб-сценария.

Элементы данных сценария

Как только сценарий будет создан, Zabbix автоматически добавит сопутствующие элементы данных для мониторинга.

| Элемент данных | Описание |
|---|---|
| *Download speed for scenario <Scenario>* (*Скорость загрузки сценария <Сценарий>*) | Этот элемент данных собирает информацию о скорости загрузки (байт в секунду) всего сценария, т.е. средний показатель для всех шагов.<br>Ключ элемента данных: web.test.in[Сценарий,,bps]<br>Тип: *Числовой (с плавающей точкой)* |
| *Failed step of scenario <Scenario>* (*Неуспешный шаг сценария <Сценарий>*) | Этот элемент данных отображает номер неуспешного шага в сценарии. Если все шаги в сценарии выполнены успешно, то возвращается 0.<br>Ключ элемента данных: web.test.fail[Сценарий]<br>Тип: *Числовой (целое)* |
| *Last error message of scenario <Scenario>* (*Последнее сообщение об ошибке для сценария <Сценарий>*) | Этот элемент данных возвращает текст последнего сообщения об ошибке в сценарии. Новое значение сохраняется только, если в сценарии есть неуспешный шаг. Если все шаги в сценарии выполнены успешно, то значение не сохраняется.<br>Ключ элемента данных: web.test.error[Сценарий]<br>Тип: *Символ* |

Вместо «Сценарий» будет использовано реальное имя сценария.

**Note:**
Если имя сценария содержит пользовательские макросы, эти макросы останутся нераскрытыми в именах элементов данных веб-мониторинга. <br><br> Если имя сценария начинается с двойной кавычки или содержит запятую либо квадратные скобки, оно будет корректно заключено в кавычки в ключах элементов данных. В других случаях дополнительные кавычки не добавляются.

**Note:**
Элементы данных веб-мониторинга добавляются с 30-дневным периодом хранения истории и с 90-дневным периодом хранения динамики изменений.

Эти элементы данных можно использовать для создания триггеров, а также при настройке условий оповещения.

Пример 1

Для создания триггера «Ошибка в веб-сценарии» вы можете указать следующее выражение триггера:

```
last(/  _  /web.test.fail[    ])<>0
```

Не забудьте заменить «Сценарий» на реальное название вашего сценария.

Пример 2

Для создания триггера «Веб-сценарий завершился с ошибкой» с полезным описанием проблемы в имени триггера вы можете указать триггер с именем:

```
        "Scenario"              : {ITEM.VALUE}
```

и выражением триггера:

```
length(last(/  _  /web.test.error[    ]))>0 and last(/  _  /web.test.fail[    ])>0
```

Не забудьте заменить «Сценарий» на реальное название вашего сценария.

Пример 3

Для создания триггера «Веб-приложение медленно отвечает» вы можете указать следующее выражение триггера:

```
last(/  _  /web.test.in[    ,,bps])<10000
```

Не забудьте заменить «Сценарий» на реальное название вашего сценария.

Элементы данных шага сценария

Как только шаг создан, Zabbix автоматически добавит сопутствующие элементы данных для мониторинга.

| Элемент данных | Описание |
|---|---|
| *Download speed for step <Step> of scenario <Scenario>* (*Скорость загрузки для шага <Шаг> сценария <Сценарий>*) | Этот элемент данных собирает информацию о скорости загрузки (байт в секунду) для указанного шага.<br>Ключ элемента данных: web.test.in[Сценарий,Шаг,bps]<br>Тип: *Числовой (с плавающей точкой)* |
| *Response time for step <Step> of scenario <Scenario>* (*Время ответа для шага <Шаг> сценария <Сценарий>*) | Этот элемент данных собирает информацию о времени ответа в секундах для указанного шага. Время ответа считается от начала отправки запроса до окончания получения всей передаваемой информации.<br>Ключ элемента данных: web.test.time[Сценарий,Шаг,resp]<br>Тип: *Числовой (с плавающей точкой)* |
| *Response code for step <Step> of scenario <Scenario>* (*Код ответа для шага <Шаг> сценария <Сценарий>*) | Этот элемент данных собирает информацию о возвращенных кодах ответа указанного шага.<br>Ключ элемента данных: web.test.rspcode[Сценарий,Шаг]<br>Тип: *Числовой (целое положительное)* |

Будут использоваться настоящие имена сценария и шага вместо «Сценарий» и «Шаг» соответственно.

**Note:**
Элементы данных веб-мониторинга добавляются с 30-дневным сроком хранения истории и 90-дневным сроком хранения динамики изменений.

**Note:**
Если имя сценария начинается с двойных кавычек или содержит запятые либо квадратные скобки, они будут корректно экранированы в ключах элементов данных. В других случаях дополнительное экранирование не выполняется.

Эти элементы данных можно использовать для создания триггеров и настройки условий оповещения. Например, для создания триггера «Вход в веб-интерфейс Zabbix слишком медленный» вы можете указать следующее выражение триггера:

```
last(/zabbix/web.test.time[ZABBIX GUI,Login,resp])>3
```

**2. Сценарий из реальной жизни**

Обзор

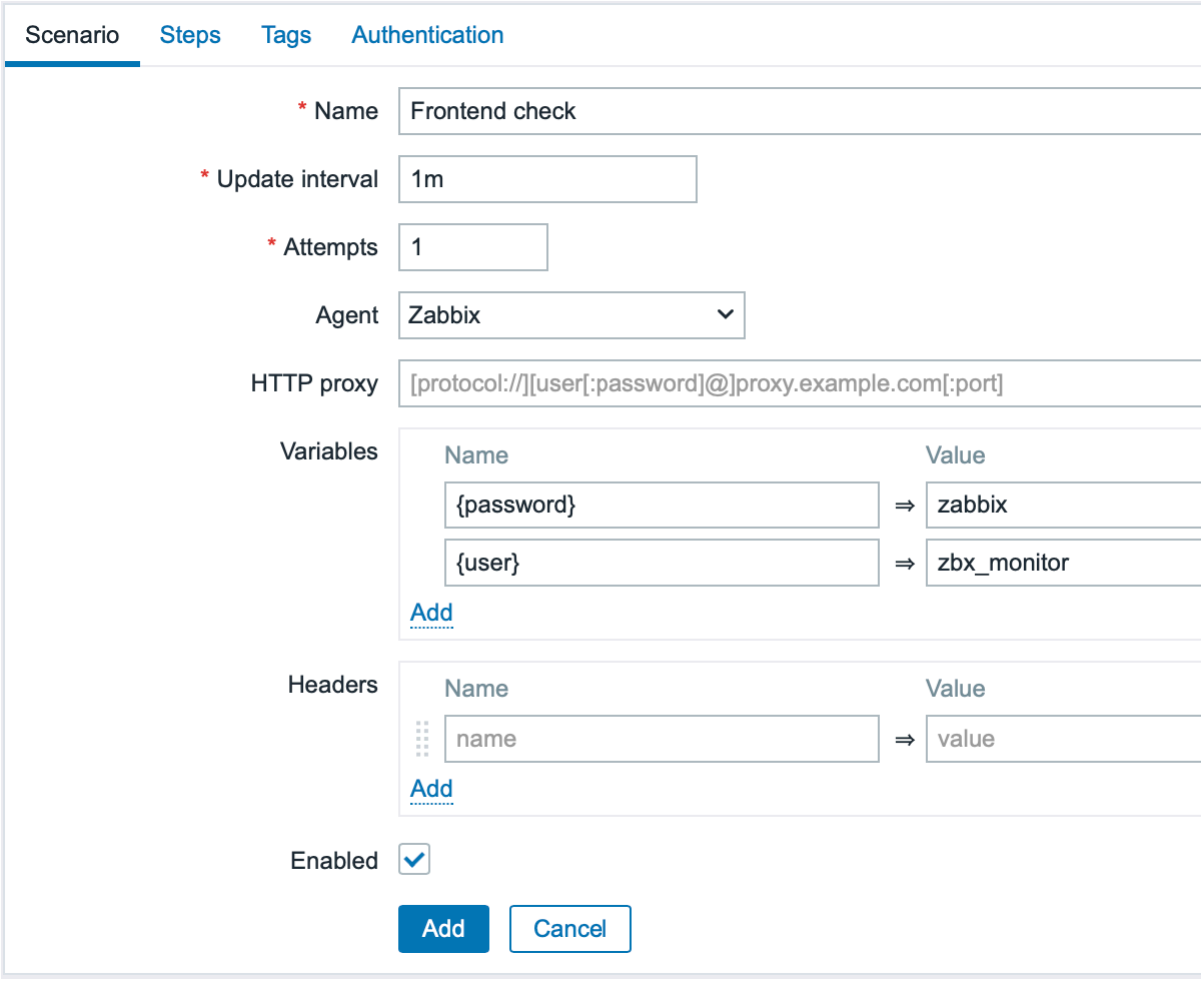
В этом разделе представлен пошаговый пример из реальной жизни того, как можно использовать веб-мониторинг.

Веб-мониторинг Zabbix будет использоваться для мониторинга веб-интерфейса Zabbix. Цель состоит в том, чтобы определить, доступен ли он, предоставляет ли нужный контент и насколько быстро он работает. Для этого требуется выполнить несколько шагов, включая проверку доступности первой страницы, вход в систему с использованием имени пользователя и пароля, проверку успешности входа, выход из системы и подтверждение выхода из системы.

Сценарий

Добавление нового веб-сценария

Перейдите в *Сбор данных* → *Узлы сети*, выберите узел сети и нажмите *Web* в строке этого узла сети.
Затем нажмите *Создать веб-сценарий*.




В новой форме веб-сценария заполните следующие поля:

- **Name** - Frontend check
- **Update interval** - 1m
- **Attempts** - 1
- **Agent** - Zabbix

В разделе *Variables* добавьте две переменные: *{password}* и *{user}*.
Введите в качестве значений свои существующие учетные данные пользователя Zabbix.

В целях безопасности рекомендуется создать отдельного пользователя с минимальными правами для использования в мониторинге.

При необходимости перейдите на вкладку *Tags* и добавьте теги веб-сценария.

**Note:**
После полной настройки этот веб-сценарий автоматически добавит на узел сети элемент данных Zabbix trapper.
Вы можете использовать теги веб-сценария, чтобы быстро идентифицировать связанные элементы данных и триггеры или выполнять поиск по собранным данным.
Например, подходящие теги для этого руководства: `component: web-scenario` и/или `target: frontend`.

Настройка шагов веб-сценария

Перейдите на вкладку *Steps* и задайте шаги для сценария. Нажмите кнопку *Add*, чтобы добавить отдельный шаг.

Общие поля

Для каждого шага, описанного ниже, заполните следующие поля в дополнение к полям, специфичным для шага:

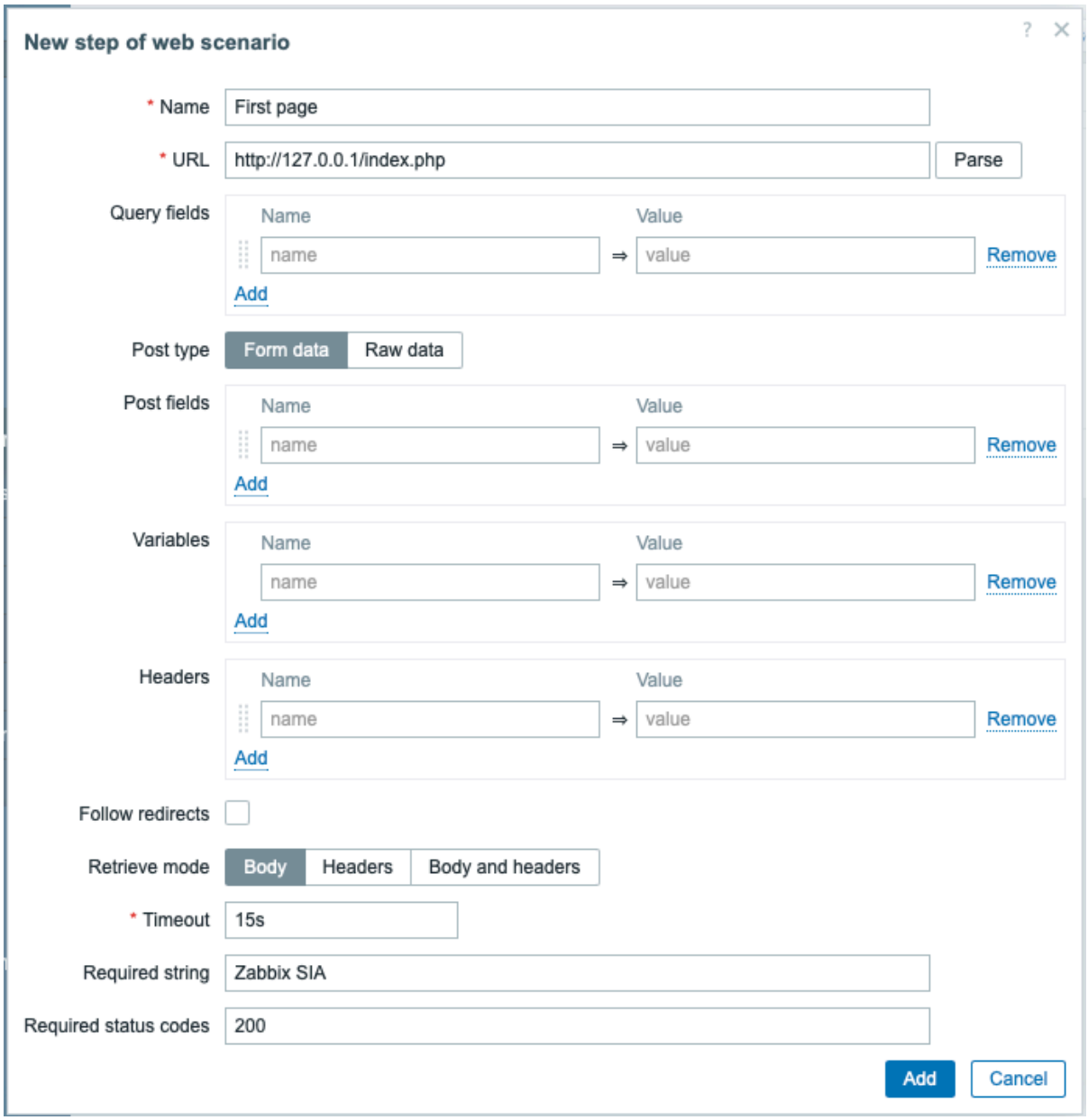
- **URL** - URL веб-интерфейса Zabbix
- **Timeout** - 15s
- **Required status codes** - 200

Шаг 1 веб-сценария

Проверьте, что первая страница отвечает корректно, возвращает код ответа HTTP 200 и содержит текст ”Zabbix SIA”.

- В поле **Name** введите *First page*.
- В поле **Required string** введите *Zabbix SIA*.
- Заполните общие поля.

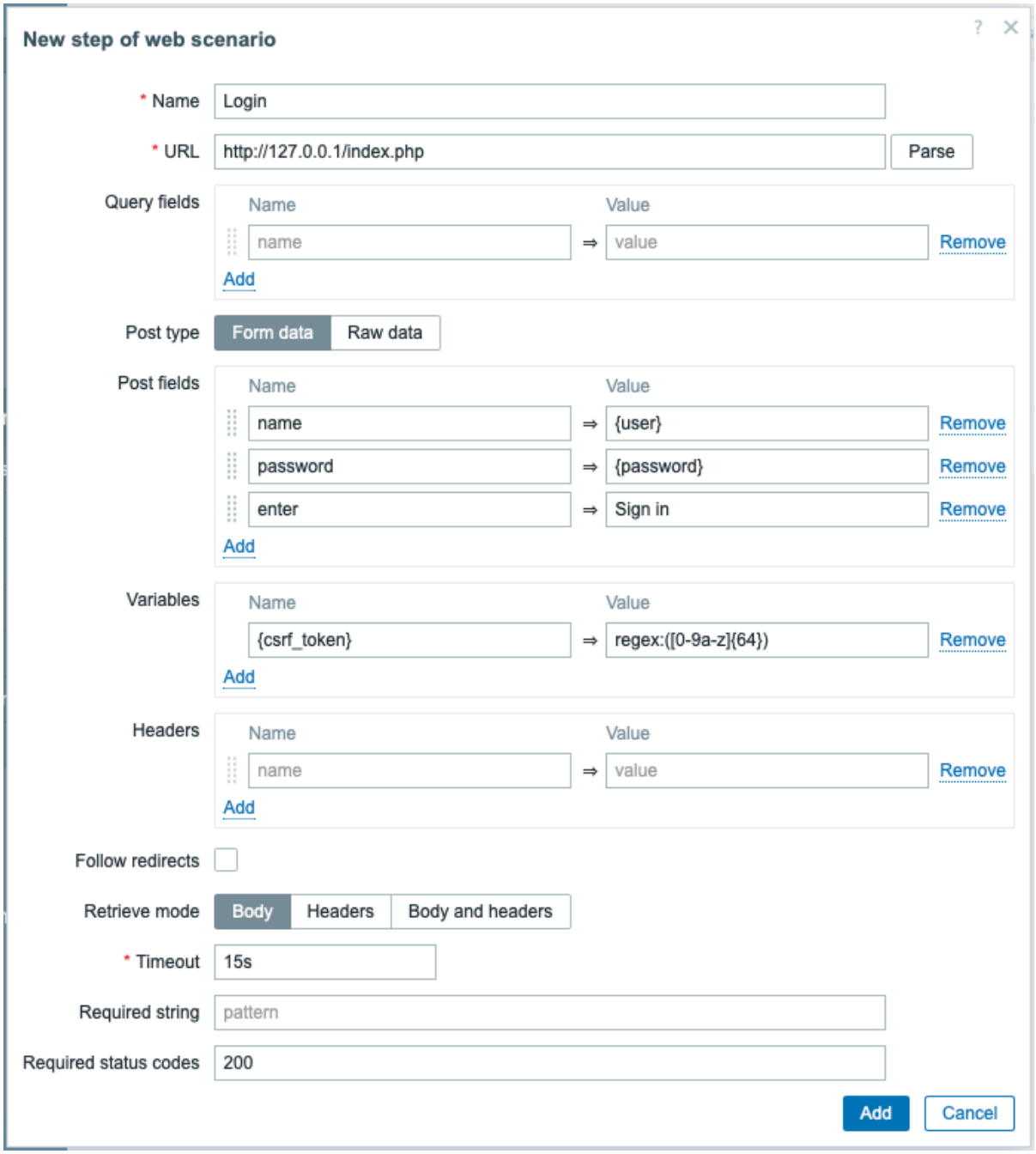
После завершения настройки шага нажмите кнопку *Add*.




Шаг 2 веб-сценария

Войдите в веб-интерфейс Zabbix, используя макросы (переменные), определенные на уровне сценария - *{user}* и *{password}*.

- В поле **Name** введите *Login*.
- В разделе **Post fields** добавьте три поля post:
  - *name* со значением *{user}*
  - *password* со значением *{password}*
  - *enter* со значением *Sign in*
- В разделе **Variables** добавьте новую переменную *{csrf_token}* со значением *regex:([0-9a-z]{64})*. Эта переменная будет перехватывать значение назначенного CSRF-токена для повторного использования в шаге 4.
- Заполните общие поля.

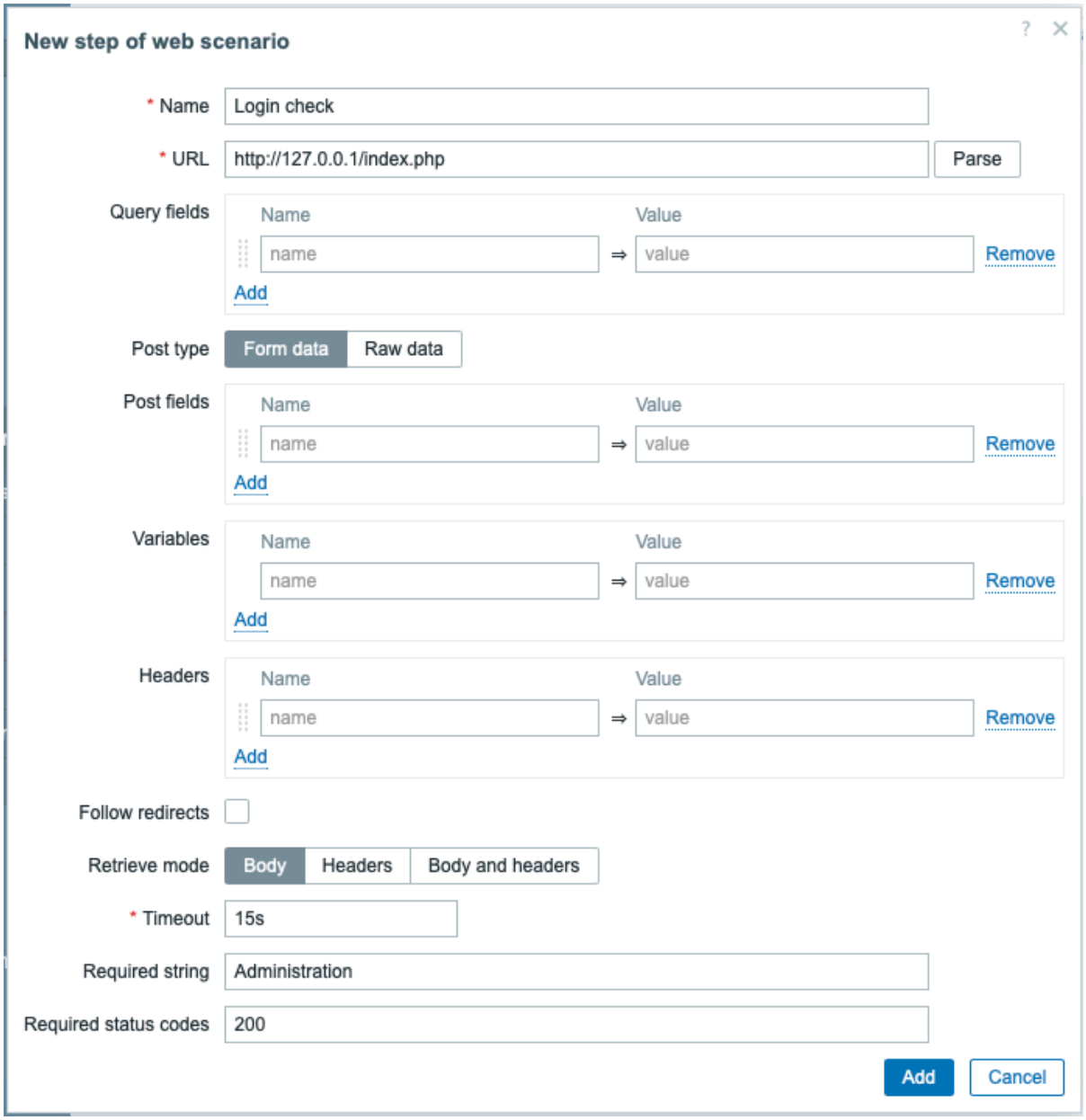



**Attention:**
Обратите внимание, что веб-интерфейс Zabbix использует перенаправление JavaScript при входе в систему, поэтому сначала должен выполняться вход, а функции для вошедшего в систему пользователя можно проверять только на последующих шагах. Кроме того, шаг входа должен использовать полный URL к файлу **index.php**.

Шаг 3 веб-сценария

После входа в систему проверьте успешность, убедившись, что отображается строка, видимая только после входа в систему, например *Administration*.

- В поле **Name** введите *Login check*.
- В поле **Required string** введите *Administration*.
- Заполните общие поля.

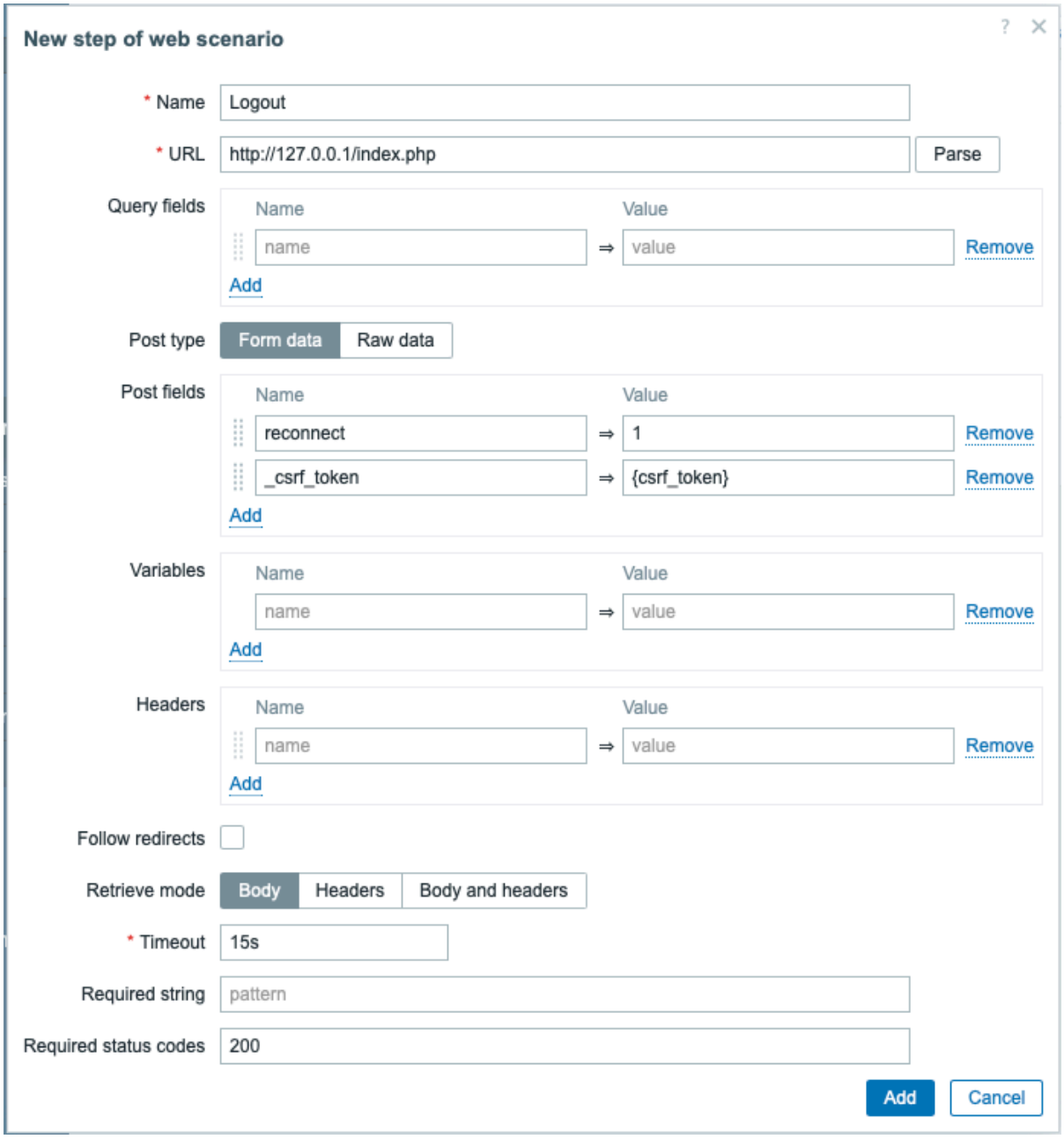



Шаг 4 сценария веб-мониторинга

После того как доступность веб-интерфейса и вход в систему будут проверены, добавьте шаг выхода из системы — иначе база данных Zabbix будет засоряться множеством открытых записей сеансов.

- В поле **Name** введите *Logout*.
- В разделе **Post fields** добавьте два поля post:
  - *reconnect* со значением *1*
  - *_csrf_token* со значением *{csrf_token}*.
- Заполните общие поля.

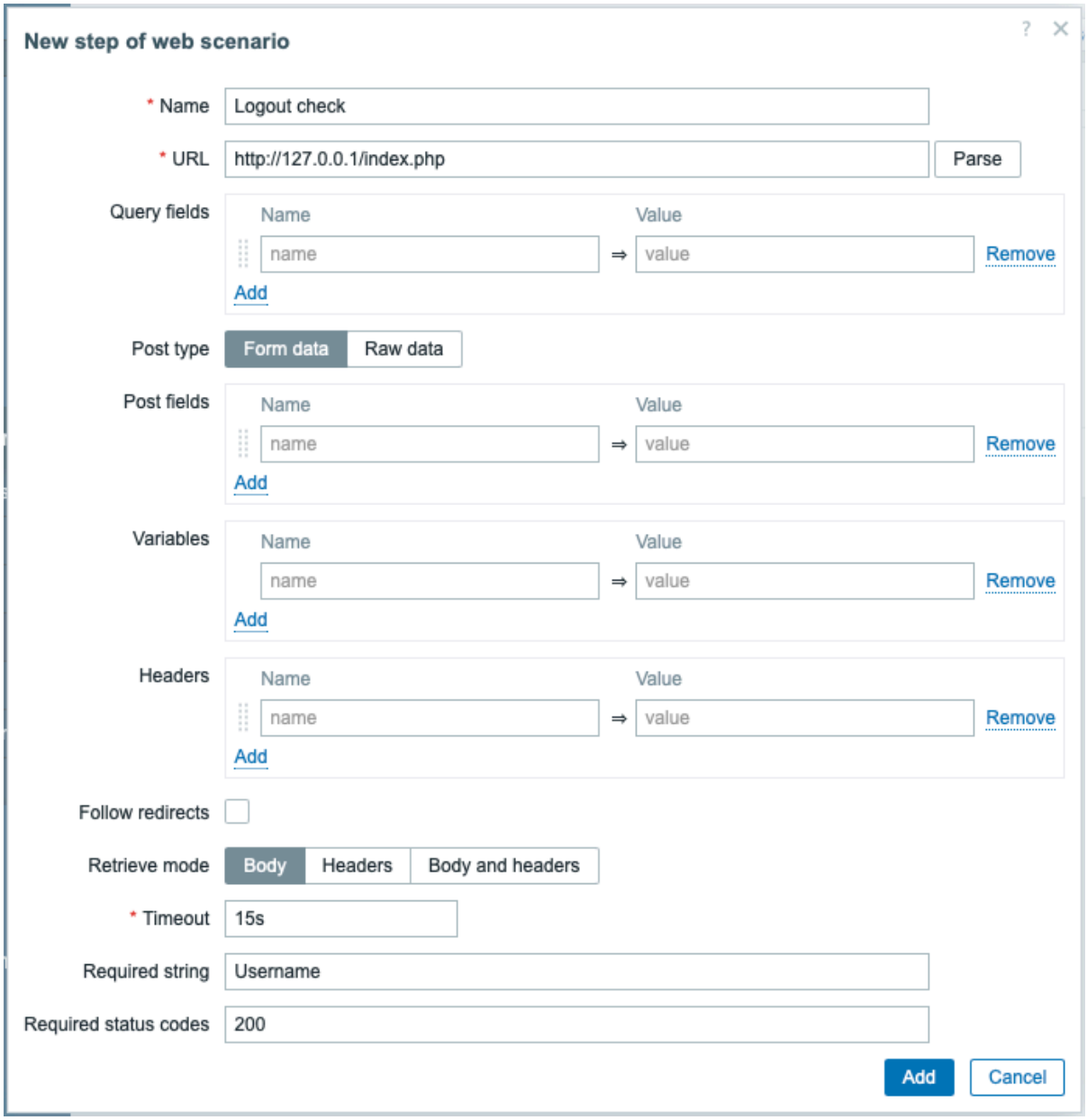
В этом шаге используется переменная {csrf_token}, полученная на шаге 2




Шаг 5 веб-сценария

Чтобы подтвердить выход из системы, проверьте наличие строки **Username**.

- В поле **Name** введите *Logout check*.
- В поле **Required string** введите *Username*.
- Заполните общие поля.




Полная конфигурация шагов

Полная конфигурация шагов веб-сценария должна выглядеть так:

| | Name | Timeout | URL | Required | Status codes | Action |
|---|---|---|---|---|---|---|
| 1: | First page | 15s | http://127.0.0.1/index.php | Zabbix SIA | 200 | Remove |
| 2: | Log in | 15s | http://127.0.0.1/index.php | | 200 | Remove |
| 3: | Log in Check | 15s | http://127.0.0.1/index.php | Administrati… | 200 | Remove |
| 4: | Log out | 15s | http://127.0.0.1/index.php | | 200 | Remove |
| 5: | Log out check | 15s | http://127.0.0.1/index.php | Username | 200 | Remove |

Add

Проверьте результаты

Сохраните завершенный сценарий веб-мониторинга.

Сценарий будет добавлен к узлу сети. Чтобы просмотреть информацию о веб-сценарии, перейдите в *Monitoring* → *Hosts*, найдите узел сети в списке и нажмите ссылку Web в последнем столбце.

Нажмите на имя сценария, чтобы увидеть более подробную статистику:

## 10. Мониторинг виртуальных машин

**Обзор** Zabbix может использовать правила низкоуровневого обнаружения для автоматического обнаружения гипервизоров и виртуальных машин VMware, а также создавать узлы сети для их мониторинга на основе предварительно определённых прототипов узлов сети.

Zabbix также включает готовые к использованию шаблоны для мониторинга гипервизоров VMware vCenter или ESXi.

Минимальная необходимая версия VMware vCenter или vSphere — 5.1.

**Сбор данных** Мониторинг виртуальных машин выполняется в два шага:

1. Процессы Zabbix *vmware collector* собирают данные виртуальных машин — процессы получают необходимую информацию от веб-сервисов VMware по протоколу SOAP, предварительно обрабатывают её и сохраняют в разделяемой памяти сервера Zabbix.
2. Процессы Zabbix *poller* извлекают данные при помощи ключей элементов данных мониторинга VMware простых проверок Zabbix.

Zabbix делит собранные данные на данные конфигурации VMware и данные счётчиков производительности VMware. Оба типа данных собираются независимо процессами *vmware collector*.

На основе данных счётчиков производительности VMware доступна статистика по следующим категориям:

- Хранилище данных
- Дисковое устройство
- Процессор
- Питание
- Сетевой интерфейс
- Элементы данных пользовательских счётчиков производительности

Для полного списка элементов данных, которые получают данные от счётчиков производительности VMware, обратитсесь к разделу *Ключи элементов данных мониторинга VMware*.

Обратите внимание, что частота получения событий VMware зависит от интервала опроса vmware.eventlog, но не может быть меньше 5 секунд.

**Настройка** Если Zabbix сервер собран из исходных кодов, для работы мониторинга виртуальных машин необходимо, чтобы он был скомпилирован с опциями сборки `--with-libcurl --with-libxml2`.

Для мониторинга виртуальных машин можно модифицировать следующие параметры файла конфигурации Zabbix сервера:

- `StartVMwareCollectors`

**Note:**
Рекомендуется включить больше коллекторов, чем число отслеживаемых служб VMware. В противном случае получение статистики счётчиков производительности VMware может быть задержано из-за получения данных конфигурации VMware (что занимает некоторое время для больших установок). <br><br> Как правило, значение `StartVMwareCollectors` не должно опускаться ниже 2 и не должно превышать двукратного количества наблюдаемых служб VMware: Количество служб < `StartVMwareCollectors` < (Количество служб * 2). Например, если у вас под наблюдением одна служба VMware, задайте `StartVMwareCollectors` равным 2; а при мониторинге трёх служб VMware, задайте параметр равным 5. <br><br> Имейте в виду, что требуемое количество коллекторов зависит также от размера вашей среды VMware, а также параметров конфигурации `VMwareFrequency` и `VMwarePerfFrequency`.

- `VMwareCacheSize`
- `VMwareFrequency`
- `VMwarePerfFrequency`
- `VMwareTimeout`

Для получения большей информации смотрите страницы файлов конфигурации Zabbix сервера и прокси.

**Attention:**
Для поддержки метрик объёмов хранилищ данных убедитесь, что значение ключа VMware `vpxd.stats.maxQueryMetrics` задано по крайней мере в 64. Для более подробной информации смотрите также статью [en] в *базе знаний VMware*.

Обнаружение

Zabbix может использовать правила низкоуровневого обнаружения (например, *vmware.hv.discovery[{$VMWARE.URL}]*) для автоматического обнаружения гипервизоров и виртуальных машин VMware. Более того, Zabbix может использовать прототипы узлов сети, чтобы автоматически генерировать реальные узлы сети для обнаруженных объектов. Для дополнительной информации обратитесь к странице *Прототипы узлов сети*.

Configuration examples

For a basic example of how to set up Zabbix for monitoring VMware using the *VMware FQDN* template, see *Monitor VMware with Zabbix*.

For a more detailed example of how to create a host, a low-level discovery rule, and a host prototype for monitoring VMware, see *Setup example*.

**Расширенное журналирование** Данные, собираемые процессами *vmware collector*, можно для подробной отладки записывать в файл журнала, используя уровень отладки 5. Этот уровень можно задать в файлах конфигурации сервера и прокси или используя опцию выполнения административных функций `-R log_level_increase="vmware collector,N"`, где «N» является номером процесса.

Например, для повышения уровня отладки с 4 до 5 для всех процессов *vmware collector* выполните следующую команду:

```
zabbix_server -R log_level_increase="vmware collector"
```

Для повышения уровня отладки с 4 до 5 для второго процесса *vmware collector* выполните следующую команду:

```
zabbix_server -R log_level_increase="vmware collector,2"
```

Когда расширенное журналирование данных коллекторов VMware больше не требуется, рекомендуется уровень отладки понизить до значения по умолчанию (3) командой `-R log_level_decrease`.

**Решение проблем**

- В случае недоступности метрик, пожалуйста, убедитесь, что они не стали недоступными или отключёнными по умолчанию в последних версиях VMware vSphere, или не наложены ли какие-либо ограничения на запросы метрик производительности к базе данных. Для получения более подробных сведений обратитесь к ZBX-12094 [en].
- В случае ошибки *`config.vpxd.stats.maxQueryMetrics` is invalid or exceeds the maximum number of characters permitted (`config.vpxd.stats.maxQueryMetrics`ошибочен или превышает максимальное количество разрешённых символов)*, добавьте параметр `config.vpxd.stats.maxQueryMetrics` в настройки сервера vCenter. Значение этого параметра должно быть таким же, как и `maxQuerysize` в файле VMware *web.xml*. Для получения более подробных сведений смотрите эту статью базы знаний VMware [en].
- Если вы подозреваете, что ваша установка Zabbix использует слишком много памяти, смотрите Профилирование чрезмерного использования памяти с помощью tcmalloc.

### 1. Ключи элементов данных для мониторинга VMware

**Обзор** На этой странице представлены подробные сведения о простых проверках, которые можно использовать для мониторинга сред виртуализации VMware.

Метрики сгруппированы по целям мониторинга.

**Поддерживаемые ключи элементов данных** Ключи элементов данных перечислены без параметров и дополнительной информации. Щелкните ключ элемента данных, чтобы увидеть полные сведения.

| Ключ элемента данных | Описание | Группа элементов данных |
|---|---|---|
| vmware.eventlog | Журнал событий VMware. | Общая служба |
| vmware.fullname | Полное имя службы VMware. | |
| vmware.version | Версия службы VMware. | |
| vmware.cl.perfcounter | Метрики счетчика производительности кластера VMware. | Cluster |
| vmware.cluster.alarms.get | Данные тревог кластера VMware. | |
| vmware.cluster.discovery | Обнаружение кластеров VMware. | |
| vmware.cluster.property | Свойство кластера VMware. | |
| vmware.cluster.status | Состояние кластера VMware. | |
| vmware.cluster.tags.get | Массив тегов кластера VMware. | |
| vmware.datastore.alarms.get | Данные тревог хранилища данных VMware. | Хранилище данных |
| vmware.datastore.discovery | Обнаружение хранилищ данных VMware. | |
| vmware.datastore.hv.list | Список гипервизоров хранилища данных. | |
| vmware.datastore.perfcounter | Значение счетчика производительности хранилища данных VMware. | |
| vmware.datastore.property | Свойство хранилища данных VMware. | |
| vmware.datastore.read | Время операции чтения из хранилища данных. | |

| Ключ элемента данных | Описание | Группа элементов данных |
|---|---|---|
| vmware.datastore.size | Объем хранилища данных VMware в байтах или в процентах от общего объема. | |
| vmware.datastore.tags.get | Массив тегов хранилища данных VMware. | |
| vmware.datastore.write | Время операции записи в хранилище данных. | |
| vmware.dc.alarms.get | Данные тревог центра обработки данных VMware. | Центр обработки данных |
| vmware.dc.discovery | Обнаружение центров обработки данных VMware. | |
| vmware.dc.tags.get | Массив тегов центра обработки данных VMware. | |
| vmware.dvswitch.discovery | Обнаружение распределенных коммутаторов VMware vSphere Distributed Switch. | vSphere Distributed Switch |
| vmware.dvswitch.fetchportdata | Данные портов VMware vSphere Distributed Switch. | |
| vmware.hv.alarms.get | Данные тревог гипервизора VMware. | Гипервизор |
| vmware.hv.cluster.name | Имя кластера гипервизора VMware. | |
| vmware.hv.connectionstate | Состояние подключения гипервизора VMware. | |
| vmware.hv.cpu.usage | Использование процессора гипервизора VMware (Hz). | |
| vmware.hv.cpu.usage.perf | Использование процессора гипервизора VMware в процентах за интервал. | |
| vmware.hv.cpu.utilization | Использование процессора гипервизора VMware в процентах за интервал, зависит от управления питанием или HT. | |
| vmware.hv.datacenter.name | Имя центра обработки данных гипервизора VMware. | |
| vmware.hv.datastore.discovery | Обнаружение хранилищ данных гипервизора VMware. | |
| vmware.hv.datastore.list | Список хранилищ данных гипервизора VMware. | |
| vmware.hv.datastore.multipath | Количество доступных путей к хранилищу данных. | |
| vmware.hv.datastore.read | Среднее время операции чтения из хранилища данных. | |
| vmware.hv.datastore.size | Объем хранилища данных VMware в байтах или в процентах от общего объема. | |
| vmware.hv.datastore.write | Среднее время операции записи в хранилище данных. | |
| vmware.hv.discovery | Обнаружение гипервизоров VMware. | |
| vmware.hv.diskinfo.get | Данные дисков гипервизора VMware. | |
| vmware.hv.fullname | Полное имя продукта, включая сведения о версии. | |
| vmware.hv.hw.cpu.freq | Частота процессора гипервизора VMware. | |
| vmware.hv.hw.cpu.model | Модель процессора гипервизора VMware. | |
| vmware.hv.hw.cpu.num | Количество ядер процессора на гипервизоре VMware. | |
| vmware.hv.hw.cpu.threads | Количество потоков процессора на гипервизоре VMware. | |
| vmware.hv.hw.memory | Общий объем памяти гипервизора VMware. | |
| vmware.hv.hw.model | Модель гипервизора VMware. | |
| vmware.hv.hw.sensors.get | Значение аппаратных датчиков гипервизора VMware. | |
| vmware.hv.hw.serialnumber | Серийный номер гипервизора VMware. | |
| vmware.hv.hw.uuid | UUID BIOS гипервизора VMware. | |
| vmware.hv.hw.vendor | Имя производителя гипервизора VMware. | |
| vmware.hv.maintenance | Состояние обслуживания гипервизора VMware. | |
| vmware.hv.memory.size.ballooned | Объем balloon-памяти гипервизора VMware. | |
| vmware.hv.memory.used | Объем используемой памяти гипервизора VMware. | |
| vmware.hv.net.if.discovery | Обнаружение сетевых интерфейсов гипервизора VMware. | |
| vmware.hv.network.in | Статистика входящего сетевого трафика гипервизора VMware. | |
| vmware.hv.network.linkspeed | Скорость сетевого интерфейса гипервизора VMware. | |
| vmware.hv.network.out | Статистика исходящего сетевого трафика гипервизора VMware. | |
| vmware.hv.perfcounter | Значение счетчика производительности гипервизора VMware. | |
| vmware.hv.property | Свойство гипервизора VMware. | |
| vmware.hv.power | Потребление энергии гипервизором VMware. | |
| vmware.hv.sensor.health.state | Сводный датчик состояния здоровья гипервизора VMware. | |
| vmware.hv.sensors.get | Датчики состояния производителя оборудования гипервизора VMware. | |
| vmware.hv.status | Состояние гипервизора VMware. | |
| vmware.hv.tags.get | Массив тегов гипервизора VMware. | |
| vmware.hv.uptime | Время непрерывной работы гипервизора VMware. | |
| vmware.hv.version | Версия гипервизора VMware. | |
| vmware.hv.vm.num | Количество виртуальных машин на гипервизоре VMware. | |
| vmware.rp.cpu.usage | Использование CPU в герцах за интервал в VMware Resource Pool. | Пул ресурсов |

| Ключ элемента данных | Описание | Группа элементов данных |
|---|---|---|
| vmware.rp.memory | Показатели памяти пула ресурсов VMware. | |
| vmware.alarms.get | Данные тревог виртуального центра VMware. | Виртуальный центр |
| vmware.vm.alarms.get | Данные тревог виртуальной машины VMware. | Виртуальная машина |
| vmware.vm.attribute | Значение пользовательского атрибута виртуальной машины VMware. | |
| vmware.vm.cluster.name | Имя виртуальной машины VMware. | |
| vmware.vm.consolidationneeded | Диск виртуальной машины VMware требует консолидации. | |
| vmware.vm.cpu.latency | Процент времени, в течение которого виртуальная машина не может выполняться из-за конкуренции за доступ к физическим CPU. | |
| vmware.vm.cpu.num | Количество процессоров на виртуальной машине VMware. | |
| vmware.vm.cpu.readiness | Процент времени, в течение которого виртуальная машина была готова, но не могла быть запланирована для выполнения на физическом CPU. | |
| vmware.vm.cpu.ready | Время, в течение которого виртуальная машина была готова, но не могла быть запланирована для выполнения на физическом CPU. | |
| vmware.vm.cpu.swapwait | Процент времени CPU, затраченного на ожидание подкачки в память. | |
| vmware.vm.cpu.usage | Использование процессора виртуальной машины VMware (Hz). | |
| vmware.vm.cpu.usage.perf | Использование процессора виртуальной машины VMware в процентах за интервал. | |
| vmware.vm.datacenter.name | Имя центра обработки данных виртуальной машины VMware. | |
| vmware.vm.discovery | Обнаружение виртуальных машин VMware. | |
| vmware.vm.guest.memory.size.swapped | Объем физической памяти гостевой ОС, выгруженной в пространство подкачки. | |
| vmware.vm.guest.osuptime | Общее время, прошедшее с момента последней загрузки операционной системы. | |
| vmware.vm.hv.maintenance | Состояние обслуживания гипервизора виртуальной машины VMware. | |
| vmware.vm.hv.name | Имя гипервизора виртуальной машины VMware. | |
| vmware.vm.memory.size | Общий объем памяти виртуальной машины VMware. | |
| vmware.vm.memory.size.ballooned | Объем balloon-памяти виртуальной машины VMware. | |
| vmware.vm.memory.size.compressed | Объем сжатой памяти виртуальной машины VMware. | |
| vmware.vm.memory.size.consumed | Объем физической памяти узла сети, использованной для резервирования страниц физической памяти гостевой ОС. | |
| vmware.vm.memory.size.private | Объем частной памяти виртуальной машины VMware. | |
| vmware.vm.memory.size.shared | Объем общей памяти виртуальной машины VMware. | |
| vmware.vm.memory.size.swapped | Объем памяти виртуальной машины VMware, выгруженной в пространство подкачки. | |
| vmware.vm.memory.size.usage.guest | Использование памяти гостевой ОС виртуальной машины VMware. | |
| vmware.vm.memory.size.usage.host | Использование памяти узла сети виртуальной машины VMware. | |
| vmware.vm.memory.usage | Процент использованной физической памяти узла сети. | |
| vmware.vm.net.if.discovery | Обнаружение сетевых интерфейсов виртуальной машины VMware. | |
| vmware.vm.net.if.in | Статистика входящего трафика сетевого интерфейса виртуальной машины VMware. | |
| vmware.vm.net.if.out | Статистика исходящего трафика сетевого интерфейса виртуальной машины VMware. | |
| vmware.vm.net.if.usage | Использование сети виртуальной машиной VMware за интервал. | |
| vmware.vm.perfcounter | Значение счетчика производительности виртуальной машины VMware. | |
| vmware.vm.powerstate | Состояние питания виртуальной машины VMware. | |
| vmware.vm.property | Свойство виртуальной машины VMware. | |
| vmware.vm.snapshot.get | Состояние снимка виртуальной машины VMware. | |
| vmware.vm.state | Состояние виртуальной машины VMware. | |
| vmware.vm.storage.committed | Зафиксированный объем хранилища виртуальной машины VMware. | |
| vmware.vm.storage.readoio | Среднее количество ожидающих запросов на чтение к виртуальному диску за интервал сбора. | |
| vmware.vm.storage.totalreadlatency | Среднее время чтения с виртуального диска. | |
| vmware.vm.storage.totalwritelatency | Среднее время записи на виртуальный диск. | |
| vmware.vm.storage.uncommitted | Незафиксированный объем хранилища виртуальной машины VMware. | |
| vmware.vm.storage.unshared | Неразделяемый объем хранилища виртуальной машины VMware. | |
| vmware.vm.storage.writeoio | Среднее количество ожидающих запросов на запись к виртуальному диску за интервал сбора. | |
| vmware.vm.tags.get | Массив тегов виртуальной машины VMware. | |

| Ключ элемента данных | Описание | Группа элементов данных |
|---|---|---|
| vmware.vm.tools | Состояние или версия guest tools виртуальной машины VMware. | |
| vmware.vm.uptime | Время непрерывной работы виртуальной машины VMware. | |
| vmware.vm.vfs.dev.discovery | Обнаружение дисковых устройств виртуальной машины VMware. | |
| vmware.vm.vfs.dev.read | Статистика чтения дискового устройства виртуальной машины VMware. | |
| vmware.vm.vfs.dev.write | Статистика записи дискового устройства виртуальной машины VMware. | |
| vmware.vm.vfs.fs.discovery | Обнаружение файловых систем виртуальной машины VMware. | |
| vmware.vm.vfs.fs.size | Статистика файловой системы виртуальной машины VMware. | |

**Подробности ключей элементов данных** Параметры без угловых скобок обязательны. Параметры, обозначенные угловыми скобками **< >**, опциональны.

vmware.eventlog[url,<mode>,<severity>]

<br> Журнал событий VMware.<br> Возвращаемое значение: *Log*.

Параметры:

- **url** - URL службы VMware;
- **mode** - *all* (по умолчанию) или *skip* - пропуск обработки более старых данных;
- **severity** - фильтрация по уровню серьезности: *error*, *warning*, *info* или *user*. Этот параметр должен быть заключен в кавычки, если указано более одного уровня серьезности в списке, разделенном запятыми (например,"`error,warning,info,u`
По умолчанию отключено.

Комментарии:

- Для одного URL может быть только один ключ элемента данных `vmware.eventlog`;
- См. также пример фильтрации записей журнала событий VMware.

vmware.fullnameurl

<br> Полное имя службы VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware.

vmware.versionurl

<br> Версия службы VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware.

vmware.cl.perfcounter[url,id,путь,<экземпляр>]

<br> Метрики счётчиков производительности VMware кластера.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **id** — ID кластера VMware. `id` может быть получено из `vmware.cluster.discovery[]` как {#CLUSTER.ID}.
- **путь** — путь к счётчику производительности[1];
- **экземпляр** — экземпляр счётчика производительности.

vmware.cluster.alarms.get[url,id]

<br> Данные alarm'ов кластера VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **id** — ID кластера VMware.

vmware.cluster.discoveryurl

<br> Обнаружение кластеров VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware.

vmware.cluster.property[url,id,prop]

<br> Свойство кластера VMware.<br> Возвращаемое значение: *String*.

Параметры:

- **url** - URL службы VMware;
- **id** - ID кластера VMware;
- **prop** - путь к свойству, которое является именем свойства объекта VM, как определено в VMware SDK.

Примеры:

```
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},overallStatus]
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},runtime.powerState]
```

vmware.cluster.status[url,имя]

<br> Статус кластера VMware.<br> Возвращаемое значение: *0* — серый; *1* — зелёный; *2* — жёлтый; *3* — красный.

Параметры:

- **url** — URL службы VMware;
- **имя** — имя кластера VMware.

vmware.cluster.tags.get[url,id]

<br> Массив тегов кластера VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **id** — ID кластера VMware.

Этот элемент данных работает с vSphere 6.5 и новее.

vmware.datastore.alarms.get[url,uuid]

<br> Данные alarm'ов хранилищ данных VMware.<br> Возвращаемое значение: Объект JSON.

Parameters:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор хранилища данных VMware.

vmware.datastore.discovery[url,<filter_uuid>]

<br> Обнаружение хранилищ данных VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **filter_uuid** — один уникальный идентификатор HV или VM (по умолчанию: пусто). Если указано, будут обнаружены только присоединённые хранилища данных.

vmware.datastore.hv.list[url,хранилище]

<br> Список гипервизоров по хранилищу.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **хранилище** — UUID или имя хранилища.

Пример вывода:

```
esx7-01-host.zabbix.sandbox
esx7-02-host.zabbix.sandbox
```

vmware.datastore.perfcounter[url,uuid,путь,<экземпляр>]

<br> Значение счётчика производительности хранилища данных VMware.<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — уникальный глобальный идентификатор хранилища данных VMware;
- **путь** — путь к счётчику производительности[1];

- **экземпляр** — экземпляр счётчика производительности. Оставьте пустым для получения агрегированных значений (по умолчанию). может быть получен из `vmware.datastore.discovery[]` как часть массива {#DATASTORE.EXTENT}.

vmware.datastore.property[url,uuid,prop]

<br> Свойство хранилища данных VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор хранилища данных VMware;
- **prop** — путь свойства.

vmware.datastore.read[url,хранилище,<режим>]

<br> Время операций чтения из хранилища (в миллисекундах).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **хранилище** — UUID или имя хранилища;
- **режим** — *latency* (среднее значение, по умолчанию) или *maxlatency* (максимальное значение).

vmware.datastore.size[url,хранилище,<режим>]

<br> Размер хранилища данных VMware в байтах или в процентах от общего размера.<br> Возвращаемое значение: *Целое число* — для байт; *Дробное число* — для процентов.

Параметры:

- **url** — URL службы VMware;
- **хранилище** — UUID или имя хранилища;
- **режим** — возможные значения: *total* (по умолчанию), *free*, *pfree* (свободно, в процентах), *uncommitted*.

vmware.datastore.tags.get[url,uuid]

<br> Массив тегов хранилища данных VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор хранилища данных VMware.

Этот элемент данных работает с vSphere 6.5 и новее.

vmware.datastore.write[url,хранилище,<режим>]

<br> Время операций записи в хранилище (в миллисекундах).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **хранилище** — UUID или имя хранилища;
- **режим** — *latency* (среднее значение, по умолчанию) или *maxlatency* (максимальное значение).

vmware.dc.alarms.get[url,id]

<br> Данные alarm'ов датацентра VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **id** — ID датацентра VMware.

vmware.dc.discoveryurl

<br> Обнаружение датацентров VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware.

vmware.dc.tags.get[url,id]

<br> Массив тегов датацентра VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **id** — ID датацентра VMware.

Этот элемент данных работает с vSphere 6.5 и новее.

vmware.dvswitch.discoveryurl

<br> Обнаружение распределённых свитчей VMware vSphere.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware.

vmware.dvswitch.fetchports.get[url,uuid,<фильтр>,<режим>]

<br> Данные портов распределённого свитча VMware vSphere.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор распределённого свитча VMware.
- **фильтр** — одна строка с критериями выбора портов, перечисленными через запятую;
- **режим** — *state* (все XML без узлов XML "config", по умолчанию) или *full*.

Параметр **фильтр** поддерживает критерии [en], доступные в объекте данных VMware DistributedVirtualSwitchPortCriteria.

Пример:

```
vmware.dvswitch.fetchports.get[{$VMWARE.URL},{$VMWARE.DVS.UUID},"connected:true,active:true,uplinkPort:fal
```

vmware.hv.alarms.get[url,uuid]

<br> Данные alarm'ов гипервизора VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.cluster.name[url,uuid]

<br> Имя кластера гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.connectionstate[url,uuid]

<br> Состояние соединения гипервизора VMware.<br> Возвращаемое значение: *Строка*: *connected*, *disconnected* или *notResponding*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.cpu.usage[url,uuid]

<br> Использование процессора гипервизором VMware (Гц).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.cpu.usage.perf[url,uuid]

<br> Использование процессора гипервизора VMware в процентном отношении за интервал времени.<br> Возвращаемое значение: *Дробное число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.cpu.utilization[url,uuid]

<br> Использование процессора гипервизора VMware в процентном отношении за интервал времени, зависит от управления питанием или HT.<br> Возвращаемое значение: *Дробное число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.datacenter.name[url,uuid]

<br> Имя датацентра гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.datastore.discovery[url,uuid]

<br> Обнаружение хранилищ у гипервизора VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.datastore.list[url,uuid]

<br> Список хранилищ гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Пример вывода:

```
SSD-RAID1-VAULT1
SSD-RAID1-VAULT2
SSD-RAID10
```

vmware.hv.datastore.multipath[url,uuid,<хранилище>,<partitionid>]

<br> Количество доступных путей до хранилища.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **хранилище** — UUID или имя хранилища;
- **partitionid** — внутренний ID физического устройства из `vmware.hv.datastore.discovery`.

vmware.hv.datastore.read[url,uuid,хранилище,<режим>]

<br> Среднее время операций чтения из хранилища данных (миллисекунд).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **хранилище** — UUID или имя хранилища;
- **режим** — *latency* (по умолчанию).

vmware.hv.datastore.size[url,uuid,хранилище,<режим>]

<br> Размер пространства VMware хранилища в байтах или в процентах от общего размера.<br> Возвращаемое значение: *Целое число* — для байт; *Число с плавающей точкой* — для процентов.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **хранилище** — UUID или имя хранилища;
- **режим** — возможные значения: *total* (по умолчанию), *free*, *pfree* (свободно, в процентах), *uncommitted*.

vmware.hv.datastore.write[url,uuid,хранилище,<режим>]

<br> Среднее время операций записи в хранилище данных (миллисекунд).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **хранилище** — UUID или имя хранилища;
- **режим** — *latency* (по умолчанию).

vmware.hv.discoveryurl

<br> Обнаружение гипервизоров VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware.

vmware.hv.diskinfo.get[url,uuid]

<br> Данные о дисках гипервизора VMware.<br> Возвращаемое значение: объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.fullname[url,uuid]

<br> Полное наименование продукта, включая информацию о версии.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.cpu.freq[url,uuid]

<br> Частота процессора гипервизора VMware (Гц).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.cpu.model[url,uuid]

<br> Модель процессора гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.cpu.num[url,uuid]

<br> Количество ядер процессора у гипервизора VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.cpu.threads[url,uuid]

<br> Количество потоков процессора у гипервизора VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.memory[url,uuid]

<br> Общее количество памяти гипервизора VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;

- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.model[url,uuid]

<br> Модель гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.sensors.get[url,uuid]

<br> Значения датчиков оборудования гипервизора VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.hw.serialnumber[url,uuid]

<br> Серийный номер гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Этот элемент данных работает с vSphere API 6.7 и новее.

vmware.hv.hw.uuid[url,uuid]

<br> UUID BIOS гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Этот элемент данных работает с vSphere API 6.7 и новее.

vmware.hv.hw.vendor[url,uuid]

<br> Имя поставщика гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Этот элемент данных работает с vSphere API 6.7 и новее.

vmware.hv.maintenance[url,uuid]

<br> Состояние обслуживания гипервизора VMware.<br> Возвращаемое значение: *0* — не в обслуживании; *1* — в обслуживании.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.memory.size.ballooned[url,uuid]

<br> Размер раздутой памяти гипервизора VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.memory.used[url,uuid]

<br> Размер используемой гипервизором VMware памяти (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url**— URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.net.if.discovery[url,uuid]

<br> Обнаружение сетевых интерфейсов гипервизора VMware.<br> Возвращаемое значение: *Объект JSON*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.network.in[url,uuid,<режим>]

<br> Статистика по входящему трафику на сетевом интерфейсе гипервизора VMware (в байтах в секунду).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **режим** — *bps* (по умолчанию), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.network.linkspeed[url,uuid,ifname]

<br> Скорость сетевого интерфейса гипервизора VMware.<br> Возвращаемое значение: *Целое число*. Если сетевой интерфейс не поднят, возвращает *0*, в противном случае — значение скорости интерфейса.

Parameters:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **ifname** — имя интерфейса.

vmware.hv.network.out[url,uuid,<режим>]

<br> Статистика по исходящему трафику на сетевом интерфейсе гипервизора VMware (в байтах в секунду).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **режим** — *bps* (по умолчанию), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.perfcounter[url,uuid,путь,<экземпляр>]

<br> Значение счётчика производительности гипервизора VMware.<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — уникальный глобальный идентификатор хранилища данных VMware;
- **путь** — путь к счётчику производительности[1];
- **экземпляр** — экземпляр счётчика производительности. Оставьте пустым для получения агрегированных значений (по умолчанию).

vmware.hv.property[url,uuid,prop]

<br> Свойство гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **prop** — путь свойства.

vmware.hv.power[url,uuid,<max>]

<br> Энергопотребление гипервизора VMware (Вт).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware;
- **max** — максимально допустимое энергопотребление .

vmware.hv.sensor.health.state[url,uuid]

<br> Сводный датчик состояния работоспособности гипервизора VMware.<br> Возвращаемое значение: *Целое число*: *0* — серый; *1* — зелёный; *2* — жёлтый; *3* — красный.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Обратите внимание, что этот элемент данных может не работать в VMware vSphere 6.5 и более поздних версиях, поскольку VMware прекратила поддержку датчика *VMware Rollup Health State*.

vmware.hv.sensors.get[url,uuid]

<br> Датчики состояний поставщика оборудования гипервизора VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.status[url,uuid]

<br> Состояние гипервизора VMware.<br> Возвращаемое значение: *Целое число*: *0* — серый; *1* — зелёный; *2* — жёлтый; *3* — красный.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Этот элемент данных использует свойство общего статуса хост-системы.

vmware.hv.tags.get[url,uuid]

<br> Массив тегов гипервизора VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Этот элемент данных работает с vSphere 6.5 и новее.

vmware.hv.uptime[url,uuid,<max>]

<br> Время работы гипервизора VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

Этот элемент данных использует свойство общего статуса хост-системы.

vmware.hv.version[url,uuid]

<br> Версия гипервизора VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.hv.vm.num[url,uuid]

<br> Количество виртуальных машин на гипервизоре VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор гипервизора VMware.

vmware.rp.cpu.usage[url,rpid]

<br> Использование CPU (в герцах) за интервал в пуле ресурсов VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **rpid** — ID пула ресурсов VMware.

vmware.rp.memory[url,rpid,<режим>]

<br> Метрики памяти пула ресурсов VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **rpid** — ID пула ресурсов VMware;
- **режим** — возможные значения:<br>*consumed* (по умолчанию) — объем физической памяти хоста, потребляемый для поддержания страниц гостевой физической памяти<br> *ballooned* — объем гостевой физической памяти, восстановленной из виртуальной машины драйвером balloon в гостевой системе<br>*overhead* — физическая память хоста, потребляемая структурами данных ESXi для работы виртуальных машин

vmware.alarms.geturl

<br> Данные alarm'ов виртуального центра VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware

vmware.vm.alarms.get[url,uuid]

<br> Данные alarm'ов виртуальной машины VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.attribute[url,uuid,name]

<br> Значение пользовательского атрибута виртуальной машины VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор хранилища данных VMware;
- **name** — имя пользовательского атрибута.

vmware.vm.cluster.name[url,uuid]

<br> Имя виртуальной машины VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.consolidationneeded[url,uuid]

<br> Диск виртуальной машины VMware требует консолидации.<br> Возвращаемое значение: *Строка*: *true* — требуется консолидация; *false* — консолидация не требуется.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware

vmware.vm.cpu.latency[url,uuid]

<br> Процент времени, в течение которого виртуальная машина не может работать из-за того, что борется за доступ к физическому(-им) CPU.<br> Возвращаемое значение: *Дробное число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.cpu.num[url,uuid]

<br> Количество процессоров на виртуальной машине VMware.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.cpu.readiness[url,uuid,<экземпляр>]

<br> Процент времени, в течение которого виртуальная машина была готова, но не получилось ее запланировать на запуск на физическом CPU.<br> Возвращаемое значение: *Дробное число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр CPU.

vmware.vm.cpu.ready[url,uuid]

<br> Время (в миллисекундах), когда виртуальная машина была готова, но не запланирована на выполнение на физическом CPU. Это время зависит от количества виртуальных машин на хосте и загрузки их CPU (%).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.cpu.swapwait[url,uuid,<экземпляр>]

<br> Процент времени CPU, потраченного на ожидание замены (swap-in).<br> Возвращаемое значение: *Дробное число*.

Parameters:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр CPU.

vmware.vm.cpu.usage[url,uuid]

<br> Использование процессора виртуальной машины VMware (Гц).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.cpu.usage.perf[url,uuid]

<br> Использование процессора виртуальной машины VMware в процентах в течение интервала.<br> Возвращаемое значение: *Дробное число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.datacenter.name[url,uuid]

<br> Имя центра обработки данных виртуальной машины VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.discoveryurl

<br> Обнаружение виртуальных машин VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware.

vmware.vm.guest.memory.size.swapped[url,uuid]

<br> Объём гостевой физической памяти, которая выгружается в файл подкачки (КБ).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.guest.osuptime[url,uuid]

<br> Общее количество времени с момента последней загрузки операционной системы (в секундах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.hv.maintenance[url,uuid]

<br> Статус обслуживания гипервизора виртуальной машины VMware.<br> Возвращаемое значение: *0* - не в режиме обслуживания; *1* - в режиме обслуживания.

Параметры:

- **url** - URL службы VMware;
- **uuid** - глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.hv.name[url,uuid]

<br> Имя гипервизора виртуальной машины VMware.<br> Возвращаемое значение: *Строка*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size[url,uuid]

<br> Общий размер памяти виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.ballooned[url,uuid]

<br> Размер раздутой памяти виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.compressed[url,uuid]

<br> Размер сжатой памяти виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.consumed[url,uuid]

<br> Объём физической памяти гипервизора (хоста), потребляемый для поддержания страниц физической памяти гостевой ОС (КБ).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.private[url,uuid]

<br> Размер приватной памяти виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.shared[url,uuid]

<br> Размер разделяемой памяти виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.swapped[url,uuid]

<br> Размер памяти, выгруженной в файл подкачки виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.usage.guest[url,uuid]

<br> Размер памяти, используемой виртуальной машиной VMware (статистика гостевой виртуальной машины, в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.size.usage.host[url,uuid]

<br> Размер памяти, используемой виртуальной машиной VMware (статистика гипервизора, в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.memory.usage[url,uuid]

<br> Процент потреблённой физической памяти гипервизора.<br> Возвращаемое значение: *Дробное число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.net.if.discovery[url,uuid]

<br> Обнаружение сетевых интерфейсов виртуальной машины VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.net.if.in[url,uuid,экземпляр,<режим>]

<br> Статистика по входящему трафику на сетевом интерфейсе виртуальной машины VMware (в байтах/пакетах в секунду).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр сетевого интерфейса;
- **режим** — *bps* (по умолчанию) или *pps* — байт либо пакетов в секунду.

vmware.vm.net.if.out[url,uuid,экземпляр,<режим>]

<br> Статистика по исходящему трафику на сетевом интерфейсе виртуальной машины VMware (в байтах/пакетах в секунду).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр сетевого интерфейса;
- **режим** — *bps* (по умолчанию) или *pps* — байт либо пакетов в секунду.

vmware.vm.net.if.usage[url,uuid,<экземпляр>]

<br> Утилизация сети виртуальной машиной VMware (суммарная статистика по входящему и исходящему трафику) в течение интервала (КБ/с).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр сетевого интерфейса.

vmware.vm.perfcounter[url,uuid,путь,<экземпляр>]

<br> Значение счётчика производительности у виртуальной машины VMware.<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — уникальный глобальный идентификатор виртуальной машины VMware;
- **путь** — путь к счётчику производительности[1];
- **экземпляр** — экземпляр счётчика производительности. Оставьте пустым для получения агрегированных значений (по умолчанию).

vmware.vm.powerstate[url,uuid]

<br> Состояние питания виртуальной машины VMware.<br> Возвращаемое значение: *0* — выключено (poweredOff); *1* — включено (poweredOn); *2* — приостановлено (suspended).

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.property[url,uuid,prop]

<br> Свойство виртуальной машины VMware.<br> Возвращаемое значение: *String*.

Параметры:

- **url** - URL службы VMware;
- **uuid** - глобальный уникальный идентификатор виртуальной машины VMware;
- **prop** - путь к свойству.

vmware.vm.snapshot.get[url,uuid]

<br> Состояние снапшота виртуальной машины VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.state[url,uuid]

<br> Состояние виртуальной машины VMware.<br> Возвращаемое значение: *Строка*: *notRunning*, *resetting*, *running*, *shuttingDown*, *standby* или *unknown*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.storage.committed[url,uuid]

<br> Объём записанных данных виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.storage.readoio[url,uuid,экземпляр]

<br> Среднее количество невыполненных запросов на чтение к виртуальному диску в течение интервала сбора данных.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр дискового устройства.

vmware.vm.storage.totalreadlatency[url,uuid,instance]

<br> Среднее время чтения с виртуального диска (в миллисекундах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр дискового устройства.

vmware.vm.storage.totalwritelatency[url,uuid,instance]

<br> Среднее время записи на виртуальный диск (в миллисекундах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр дискового устройства.

vmware.vm.storage.uncommitted[url,uuid]

<br> Объём незаписанных данных виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.storage.unshared[url,uuid]

<br> Размер неразделяемого пространства хранилища виртуальной машины VMware (в байтах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.storage.writeoio[url,uuid,экземпляр]

<br> Среднее количество невыполненных запросов на запись на виртуальный диск в течение интервала сбора данных.<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр дискового устройства.

vmware.vm.tags.get[url,uuid]

<br> Массив тегов виртуальной машины VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

Этот элемент данных работает с vSphere 6.5 и новее.

vmware.vm.tools[url,uuid,<режим>]

<br> Состояние гостевых инструментов виртуальной машины VMware.<br> Возвращаемое значение: *Строка*. <br> В режиме *status*: *guestToolsExecutingScripts* — VMware Tools запускаются; *guestToolsNotRunning* — VMware Tools не работают; *guestToolsRunning* — VMware Tools работают.<br> В режиме *version*: версия.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **режим** — *version*, *status* (по умолчанию).

vmware.vm.uptime[url,uuid]

<br> Время непрерывной работы виртуальной машины VMware (в секундах).<br> Возвращаемое значение: *Целое число*.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.vfs.dev.discovery[url,uuid]

<br> Обнаружение дисковых устройств виртуальной машины VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

vmware.vm.vfs.dev.read[url,uuid,экземпляр,<режим>]

<br> Статистика чтения с дискового устройства виртуальной машины VMware (в байтах/операциях в секунду).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр дискового устройства;
- **режим** — *bps* (по умолчанию) или *ops* — байт или операций в секунду.

vmware.vm.vfs.dev.write[url,uuid,экземпляр,<режим>]

<br> Статистика записи на дисковое устройство виртуальной машины VMware (в байтах/операциях в секунду).<br> Возвращаемое значение: *Целое число* [2].

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **экземпляр** — экземпляр дискового устройства;
- **режим** — *bps* (по умолчанию) или *ops* — байт или операций в секунду.

vmware.vm.vfs.fs.discovery[url,uuid]

<br> Обнаружение файловых систем виртуальной машины VMware.<br> Возвращаемое значение: Объект JSON.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware.

Для работы этого элемента данных необходимо, чтобы на гостевой виртуальной машине были установлены VMware Tools.

vmware.vm.vfs.fs.size[url,uuid,fsname,<режим>]

<br> Статистика файловой системы виртуальной машины VMware (в байтах/процентах).<br> Возвращаемое значение: *Целое число* — для байт; *Дробное число* — для процентов.

Параметры:

- **url** — URL службы VMware;
- **uuid** — глобальный уникальный идентификатор виртуальной машины VMware;
- **fsname** — имя файловой системы;
- **режим** — *total*, *free*, *used*, *pfree* или *pused*.

Для работы этого элемента данных необходимо, чтобы на гостевой виртуальной машине были установлены VMware Tools.

Примечания

[1] Смотрите Создание пользовательских имён счётчиков производительности для VMware.

[2] Значения по этим элементам данных извлекаются из счётчиков производительности VMware, и параметр VMwarePerfFrequency используется для обновления данных по этим элементам данных в Zabbix VMware кэше:

- 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

Дополнительная информация

Смотрите раздел мониторинга виртуальных машин для получения более подробной информации о настройке Zabbix для мониторинга сред VMware.

**2. Поля ключей обнаружения виртуальных машин**

В следующей таблице перечислены поля, возвращаемые ключами обнаружения, относящимися к виртуальным машинам.

| **Ключ элемента данных** | | |
|---|---|---|
| **Описание** | **Поле** | **Возвращаемое содержимое** |
| **vmware.cluster.discovery** | | |
| Выполнение обнаружения кластеров. | {#CLUSTER.ID} | Идентификатор кластера. |
| | {#CLUSTER.NAME} | Имя кластера. |
| | ”resource_pool” | Массив, содержащий данные пула ресурсов, в том числе ID группы ресурсов, массив тегов, путь пула ресурсов, количество виртуальных машин.<br><br>Структура массива:<br>`[{`<br>`"rpid":"id ",`<br>`"tags":[{}],`<br>`"rpath":" ",`<br>`"vm_count":0`<br>`}]` |
| | ”tags” | Относительно структуры массива `"tags"` смотрите поле «tags».<br>Массив, содержащий теги с именем тега, описанием и категорией.<br><br>Структура ассива:<br>`[{`<br>`"tag":" ",`<br>`"tag_description":" ",`<br>`"category":" "`<br>`}]` |
| **vmware.datastore.discovery** | | |
| Выполнение обнаружения хранилищ данных. | {#DATASTORE} | Имя хранилища. |
| | {#DATASTORE.EXTENT} | Массив, содержащий ID раздела и имя экземпляра экстента хранилища.<br><br>Структура массива:<br>`[{`<br>`"partitionid":1,`<br>`"instance":" "`<br>`}]` |

| Ключ элемента данных | | |
|---|---|---|
| | {#DATASTORE.TYPE} | Тип хранилища.<br><br>Примеры значений: VMFS, NFS, vsan и т.д. |
| | {#DATASTORE.UUID} | Идентификатор хранилища. |
| | "tags" | Массив, содержащий теги с именем, описанием и категорией.<br><br>Структура массива:<br>[{<br>"tag":" ",<br>"tag_description":" ",<br>"category":" "<br>}] |
| **vmware.dc.discovery** | | |
| Выполнение обнаружения центров хранения данных. | {#DATACENTER} | Имя центра хранения данных. |
| | {#DATACENTERID} | Идентификатор центра хранения данных. |
| | "tags" | Массив, содержащий теги с именем, описанием и категорией.<br><br>Структура массива:<br>[{<br>"tag":" ",<br>"tag_description":" ",<br>"category":" "<br>}] |
| **vmware.dvswitch.discovery** | | |
| Выполнение обнаружения рапределённых свитчей vSphere. | {#DVS.NAME} | Имя свитча. |
| | {#DVS.UUID} | Идентификатор свитча. |
| **vmware.hv.discovery** | | |
| Выполнение обнаружения гипервизоров. | {#HV.UUID} | Уникальный идентификатор гипервизора. |
| | {#HV.ID} | Идентификатор гипервизора (Имя управляемого объекта HostSystem). |
| | {#HV.NAME} | Имя гипервизора. |
| | {#HV.NETNAME} | Сетевое имя хоста гипервизора. |
| | {#HV.IP} | IP адрес гипервизора, может быть пустым.<br><br>В случае конфигурации с HA с несколькими сетевыми интерфейсами соблюдается следующий приоритет выбора интерфейса:<br>- предпочтение IP, который входит в IP подсеть с vCenter IP;<br>- предпочтение IP с IP подсетью с со шлюзом по умолчанию;<br>- предпочтение IP с интерфейса с наименьшим ID. |
| | {#CLUSTER.NAME} | Имя кластера, может быть пустым. |
| | {#DATACENTER.NAME} | Имя центра хранения данных. |
| | {#PARENT.NAME} | Имя контейнера, в котором хранится гипервизор. |
| | {#PARENT.TYPE} | Тип контейнера, в котором хранится гипервизор.<br>Значения могут быть: Datacenter, Folder, ClusterComputeResource, VMware, где «VMware» означает неизвестный тип контейнера. |

| Ключ элемента данных | | |
|---|---|---|
| | ”resource_pool” | Массив, содержащий данные пула ресурсов, в том числе ID группы ресурсов, массив тегов, путь пула ресурсов, количество виртуальных машин.<br><br>Структура массива:<br>[{<br>"rpid":"id ",<br>"tags":[{}],<br>"rpath":" ",<br>"vm_count":0<br>}]<br><br>Относительно структуры массива "tags" смотрите поле «tags». |
| | ”tags” | Массив, содержащий теги с именем, описанием и категорией.<br><br>Структура массива:<br>[{<br>"tag":" ",<br>"tag_description":" ",<br>"category":" "<br>}] |
| **vmware.hv.datastore.discovery** | | |
| Выполнение обнаружения хранилищ данных гипервизора. Обратите внимание, что несколько гипервизоров могут использовать одно и то же хранилище данных. | {#DATASTORE} | Имя хранилища данных. |
| | {#DATASTORE.TYPE} | Тип хранилища данных.<br><br>Примеры значений: VMFS, NFS, vsan и т.д. |
| | {#DATASTORE.UUID} | Идентификатор хранилища данных. |
| | {#MULTIPATH.COUNT} | Зарегистрированное количество путей хранилища данных. |
| | {#MULTIPATH.PARTITION.COUNT} | Количество доступных разделов диска. |
| | ”datastore_extent” | Массив, содержащий имя экземпляра и ID раздела экстента хранилища.<br><br>Структура массива:<br>[{<br>"partitionid":1,<br>"instance":" "<br>}] |
| | ”tags” | Массив, содержащий теги с именем, описанием и категорией.<br><br>Структура массива:<br>[{<br>"tag":" ",<br>"tag_description":" ",<br>"category":" "<br>}] |
| **vmware.hv.net.if.discovery** | | |
| Выполнение обнаружения сетевых интерфейсов гипервизоров. | {#IFNAME} | Имя интерфейса. |
| | {#IFDRIVER} | Драйвер интерфейса. |
| | {#IFDUPLEX} | Настройки дуплекса интерфейса. |

| Ключ элемента данных | | |
|---|---|---|
| | {#IFSPEED} | Скорость интерфейса. |
| | {#IFMAC} | MAC адрес интерфейса. |
| **vmware.vm.discovery** | | |
| Выполнение обнаружения виртуальных машин. | {#VM.UUID} | Уникальный идентификатор виртуальной машины. |
| | {#VM.ID} | Идентификатор виртуальной машины (Имя управляемого объекта VirtualMachine). |
| | {#VM.NAME} | Имя виртуальной машины. |
| | {#HV.NAME} | Имя гипервизора. |
| | {#HV.UUID} | Уникальный идентификатор гипервизора. |
| | {#HV.ID} | Идентификатор гипервизора (Имя управляемого объекта HostSystem). |
| | {#CLUSTER.NAME} | Имя кластера, может быть пустым. |
| | {#DATACENTER.NAME} | Имя центра хранения данных. |
| | {#DATASTORE.NAME} | Имя хранилища данных. |
| | {#DATASTORE.UUID} | Идентификатор хранилища данных. |
| | {#VM.IP} | IP адрес виртуальной машины, может быть пустым. |
| | {#VM.DNS} | DNS имя виртуальной машины, может быть пустым. |
| | {#VM.GUESTFAMILY} | Семейство ОС гостевой виртуальной машины, может быть пустым. |
| | {#VM.GUESTFULLNAME} | Полное имя ОС гостевой виртуальной машины, может быть пустым. |
| | {#VM.FOLDER} | Цепочка родительских папок виртуальной машины, можно использовать как значение для вложенных групп; имена папок комбинируются с «/». Может быть пустым. |
| | {#VM.TOOLS.STATUS} | Состояние инструментов виртуальной машины VMware. |
| | {#VM.POWERSTATE} | Состояние питания виртуальной машины VMware (`poweredOff`, `poweredOn` или `suspended`). |
| | {#VM.RPOOL.ID} | Идентификатор пула ресурсов. |
| | {#VM.RPOOL.PATH} | Полный путь пула ресурсов, за исключением «корневого» имени ”Resources”. Имена папок комбинируются с «/». |
| | {#VM.SNAPSHOT.COUNT} | Количество снапшотов виртуальной машины. |
| | ”tags” | Массив, содержащий теги с именем, описанием и категорией.<br><br>Структура массива:<br>`[{`<br>`"tag":" ",`<br>`"tag_description":" ",`<br>`"category":" "`<br>`}]` |
| | ”vm_customattribute” | Массив пользовательских атрибутов виртуальной машины (если задано).<br><br>Структура массива:<br>`[{`<br>`"name":" ",`<br>`"value":" "`<br>`}]` |

| Ключ элемента данных | | |
|---|---|---|
| | ”net_if” | Массив сетевых интерфейсов виртуальной машины.<br><br>Структура массива:<br>[{<br>"ifname": "  ",<br>"ifdesc": "  ",<br>"ifmac": "00:00:00:00:00:00",<br>"ifconnected": true,<br>"iftype": "  ",<br>"ifbackingdevice": "  ",<br>"ifdvswitch_uuid": "uuid  ",<br>"ifdvswitch_portgroup": "  ",<br>"ifdvswitch_port": "  ",<br>"ifip": ["ip  "]<br>}]<br><br>Относительно описания возвращаемых данных смотрите ключ элемента данных «vmware.vm.net.if.discovery». |
| **vmware.vm.net.if.discovery** | | |
| Выполнение обнаружения сетевых интерфейсов виртуальной машины. | {#IFNAME} | Имя сетевого интерфейса. |
| | {#IFDESC} | Описание сетевого интерфейса. |
| | {#IFMAC} | MAC адрес сетевого интерфейса. |
| | {#IFCONNECTED} | Состояние соединения сетевого интерфейса. (`false` — отключён; `true` — подключён). |
| | {#IFTYPE} | Тип сетевого интерфейса. |
| | {#IFBACKINGDEVICE} | Имя нижележащего устройства. |
| | {#IFDVSWITCH.UUID} | Уникальный идентификатор распределённого свитча vSphere. |
| | {#IFDVSWITCH.PORTGROUP} | Распределённая группа портов. |
| | {#IFDVSWITCH.PORT} | Порт распределённого свитча vSphere. |
| | ”ifip” | Массив адресов интерфейса. |
| **vmware.vm.vfs.dev.discovery** | | |
| Выполнение обнаружения дисковых устройств виртуальной машины. | {#DISKNAME} | Имя дискового устройства. |
| **vmware.vm.vfs.fs.discovery** | | |
| Выполнение обнаружения файловых систем виртуальной машины. | {#FSNAME} | Имя файловой системы. |

### 3. Примеры JSON для элементов данных VMware

**Обзор** В этом разделе приводится дополнительная информация о JSON-объектах, возвращаемых различными элементами данных VMware.

**vmware.*.alarms.get** Элементы данных **vmware.alarms.get[]**, **vmware.cluster.alarms.get[]**, **vmware.datastore.alarms.get[]**, **vmware.dc.alarms.get[]**, **vmware.hv.alarms.get[]**, **vmware.vm.alarms.get[]** возвращают объекты JSON со следующей структурой (значения приведены в качестве примера; объект `entity` возвращается начиная с Zabbix 7.4.10):

```
{
    "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** Элементы данных **vmware.cluster.tags.get[]**, **vmware.datastore.tags.get[]**, **vmware.dc.tags.get[]**, **vmware.hv.tags.get[]**, **vmware.vm.tags.get[]** возвращают объекты JSON следующей структуры (значения предоставлены для примера):

```
{
    "tags": [
        {
            "name": "Windows",
        "description": "tag for cat OS type",
        "category": "OS type"
    },
    {
            "name": "SQL Server",
        "description": "tag for cat application name",
        "category": "application name"
        },
    {
            //
        }
    ]
}
```

**vmware.hv.diskinfo.get** Элемент данных **vmware.hv.diskinfo.get[]** возвращает объекты JSON следующей структуры (значения предоставлены для примера):

```
[
  {
    "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": {}
  },
  {
    //
  }
]
```

**vmware.dvswitch.fetchports.get** Элемент данных **vmware.dvswitch.fetchports.get[]** возвращает объекты JSON следующей структуры (значения предоставлены для примера):

```
{
    "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"
            },
            {
                //
            }
        ]
    }
}
```

**vmware.hv.hw.sensors.get** Элемент данных **vmware.hv.hw.sensors.get[]** возвращает объекты JSON следующей структуры (значения предоставлены для примера):

```
{
    "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"
                 }
            },
            {
                    //
                }
        ]
        }
}
```

**vmware.hv.sensors.get** Элемент данных **vmware.hv.sensors.get[]** возвращает объекты JSON следующей структуры (значения предоставлены для примера):

```
{
    "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"
        },
            {
                    //
                }
        ]
        }
}
```

**vmware.vm.snapshot.get** Если существуют снапшоты, то элемент данных **vmware.snapshot.get[]** возвращает объекты JSON следующей структуры (значения предоставлены для примера):

```
{
  "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
}
```


При отсутствии снапшотов элемент данных **vmware.snapshot.get[]** возвращает объект JSON с пустыми значениями:


```
{
  "snapshot": [],
  "count": 0,
  "latestdate": null,
  "latestage": 0,
  "oldestdate": null,
  "oldestage": 0,
  "size": 0,
  "uniquesize": 0
}
```


#### 4. Пример настройки мониторинга VMware

Обзор

Следующий пример описывает, как настроить Zabbix для мониторинга виртуальных машин VMware. Это включает:

- создание узла сети, который представляет вашу среду VMware;
- создание правила низкоуровневого обнаружения, которое обнаруживает виртуальные машины в вашей среде VMware;
- создание прототипа узла сети, на основе которого Zabbix будет генерировать реальные узлы сети для виртуальных машин, обнаруженных правилом низкоуровневого обнаружения.

Предварительные условия

> **Note:**
> В этом примере не рассматриваются настройки VMware. Предполагается, что VMware уже настроено.

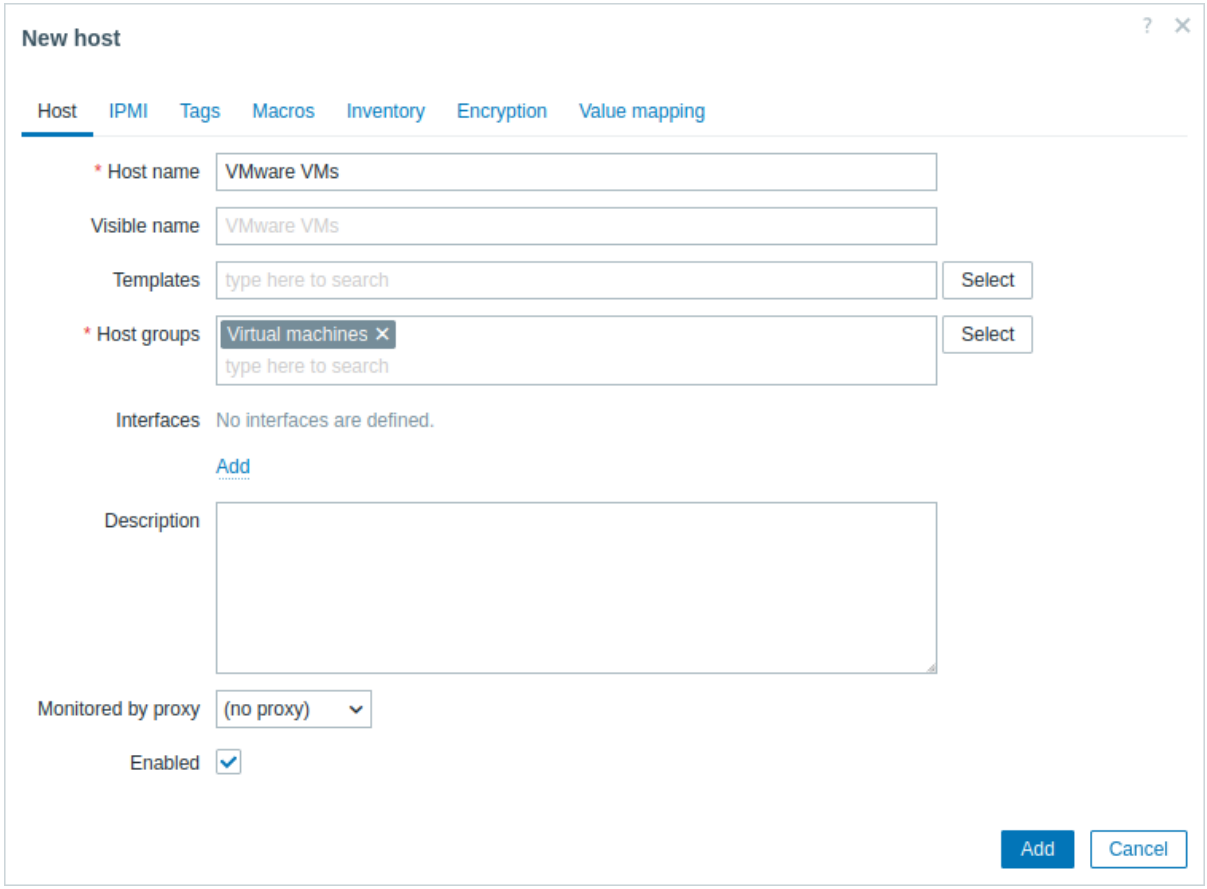
Прежде чем продолжить, установите параметр `StartVMwareCollectors` в файле конфигурации Zabbix сервера в значение 2 или более (значение по умолчанию — 0).

Создание узла сети

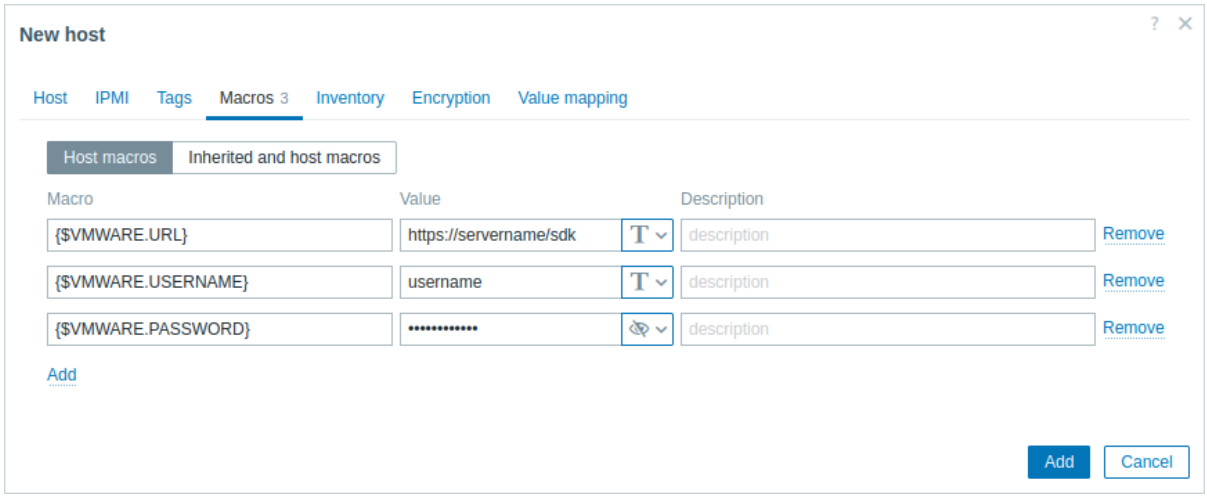
1. Перейдите к *Сбор данных (Data collection)* → *Узлы сети (Hosts)*.

2. Создайте узел сети:

- В поле *Имя узла сети (Host name)* введите имя узла сети (например, «VMware VMs»).
- В поле *Группы узлов сети (Host groups)* введите либо выберите группу узлов сети (например, «Virtual machines»).




- На вкладке *Макросы (Macros)* задайте следующие макросы узла сети:
  - {$VMWARE.URL} — URL SDK службы VMware (vCenter или ESXi hypervisor)
  - {$VMWARE.USERNAME} — имя пользователя службы VMware
  - {$VMWARE.PASSWORD} — пароль пользователя {$VMWARE.USERNAME} службы VMware



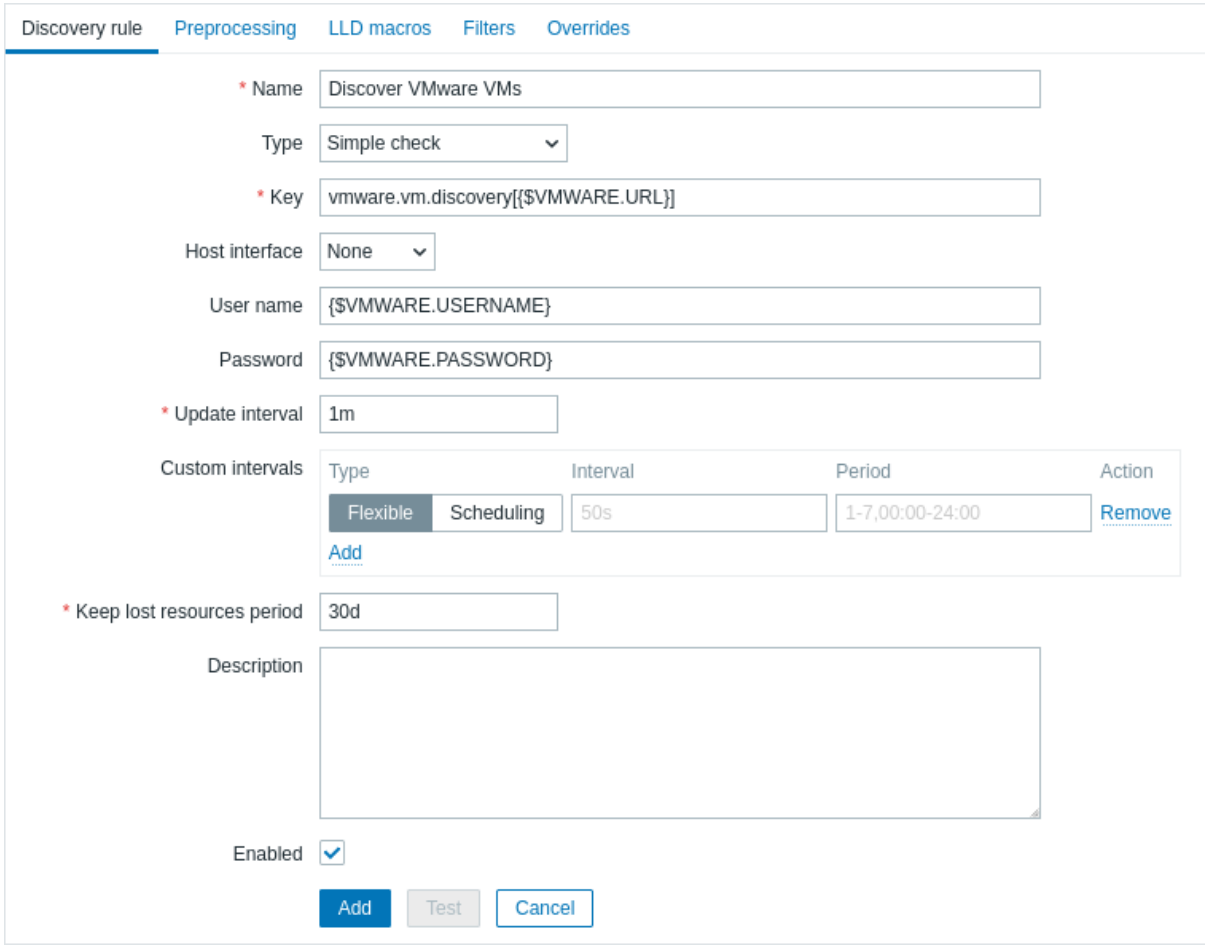

3. Нажмите кнопку *Добавить (Add)* для создания узла сети. Этот узел сети будет представлять вашу среду виртуализации VMware.

Создание правила низкоуровневого обнаружения

1. Нажмите *Обнаружение (Discovery)* напротив созданного узла сети, чтобы перейти к списку правил низкоуровневого обнаружения этого узла сети.

2. Создайте правило низкоуровневого обнаружения:

- В поле *Имя (Name)* введите имя правила низкоуровневого обнаружения (например, «Discover VMware VMs»).
- В поле *Тип (Type)* выберите «Простая проверка (Simple check)».
- В поле *Ключ (Key)* введите встроенный ключ элемента данных для обнаружения виртуальных машин VMware: `vmware.vm.discovery[{$VMWARE.URL}]`
- В полях *Имя пользователя (User name)* и *Пароль (Password)* введите соответствующие макросы, ранее настроенные на узле сети.



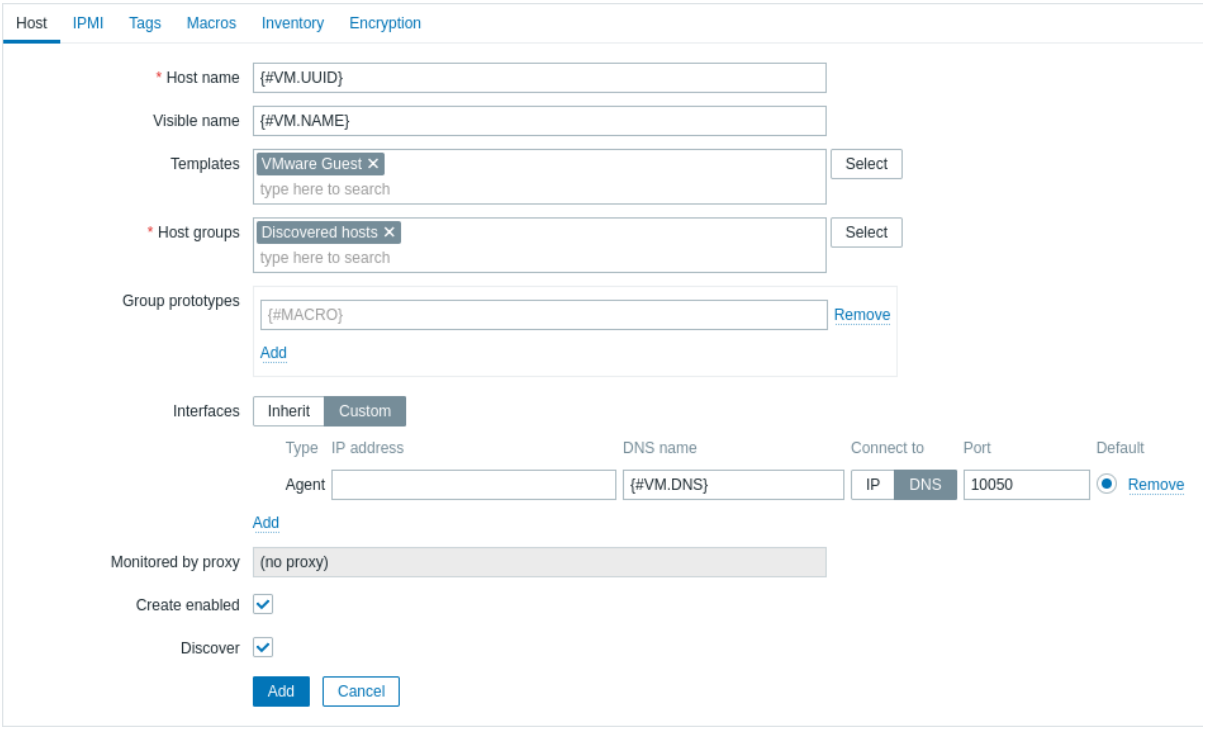

3. Нажмите кнопку *Добавить (Add)* для добавления правила низкоуровневого обнаружения. Это правило низкоуровневого обнаружения будет обнаруживать виртуальные машины в вашей среде виртуализации VMware.

Создание прототипа узла сети

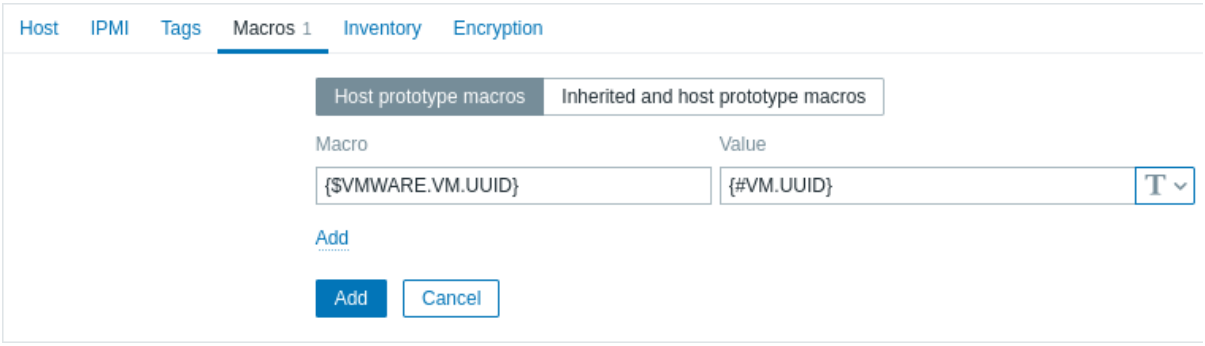
1. В списке правил низкоуровневого обнаружения напротив ранее созданного правила низкоуровневого обнаружения нажмите *Прототипы узлов сети (Host prototypes)*.

2. Создайте прототип узла сети. Поскольку прототипы узла сети — это заготовки для создания узлов сети посредством правил низкоуровневого обнаружения, большинство полей будут содержать макросы низкоуровневого обнаружения. Это гарантирует, что узлы сети будут созданы со свойствами на основе содержимого, полученного ранее созданным правилом низкоуровневого обнаружения.

- В поле *Имя узла сети (Host name)* введите макрос {#VM.UUID}.
- В поле *Видимое имя (Visible name)* введите макрос {#VM.NAME}.
- В поле *Шаблоны (Templates)* введите или выберите шаблон «VMware Guest». Это шаблон содержит элементы данных VMware и правила обнаружения для мониторинга состояния питания виртуальной машины, использования CPU, использования памяти, сетевые устройства и т.д.
- В поле *Группы узлов сети (Host groups)* введите или выберите группу узлов сети (например, «Discovered hosts»).
- В поле *Интерфейсы (Interfaces)* добавьте пользовательский интерфейс узла сети. Затем введите макрос {#VM.DNS} в поле *DNS имя (DNS name)* или введите макрос {#VM.IP} в поле *IP адрес (IP address)*. В качестве альтернативы, если виртуальные машины вашей среды VMware имеют несколько интерфейсов, перейдите к разделу *Расширенная настройка интерфейса узла сети*. Настройка пользовательского интерфейса узла сети нужна для правильного функционирования шаблона *VMware Guest*.




- На вкладке *Макросы (Macros)* укажите макрос {$VMWARE.VM.UUID} со значением {#VM.UUID}. Это необходимо для правильного функционирования шаблона *VMware Guest*, который использует данный макрос как пользовательский макрос уровня узла сети в параметрах элементов данных (например, *vmware.vm.net.if.discovery[{$VMWARE.URL}, {$VMWARE.VM.UUID}]*).

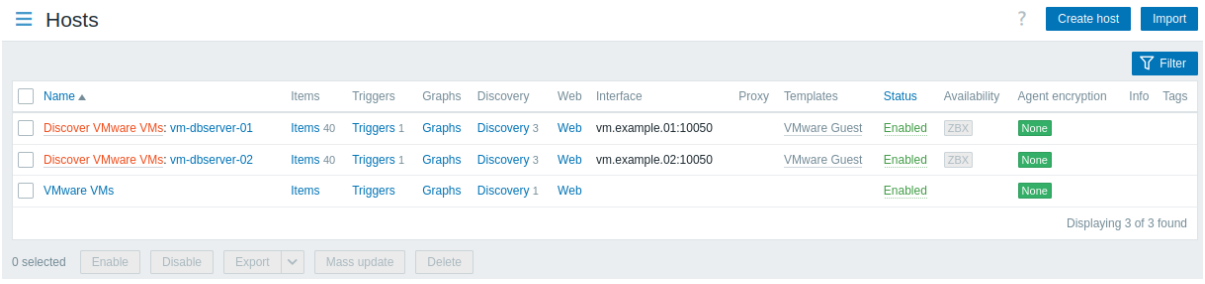



3. Нажмите кнопку *Добавить (Add)*, чтобы создать прототип узла сети. Этот прототип узла сети будет использоваться для создания узлов сети для виртуальных машин, обнаруженных ранее созданным правилом низкоуровневого обнаружения.

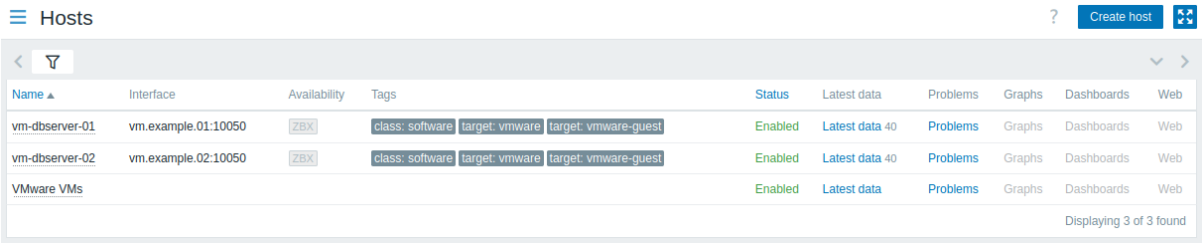
Просмотр узлов сети и метрик

После того, как прототип узла сети создан, правило низкоуровневого обнаружения будет создавать узлы сети для обнаруженных виртуальных машин VMware, а Zabbix приступит к их мониторингу. Обратите внимание, что обнаружение и создание узлов сети можно при необходимости также выполнить вручную.

Для просмотра созданных узлов сети перейдите к разделу меню *Сбор данных (Data collection) → Узлы сети (Hosts)*.




Для просмотра собранных метрик перейдите к разделу меню *Мониторинг (Monitoring) → Узлы сети (Hosts)* и нажмите *Последние данные (Latest data)* напротив одного из узлов сети.




Расширенная настройка интерфейса узла сети

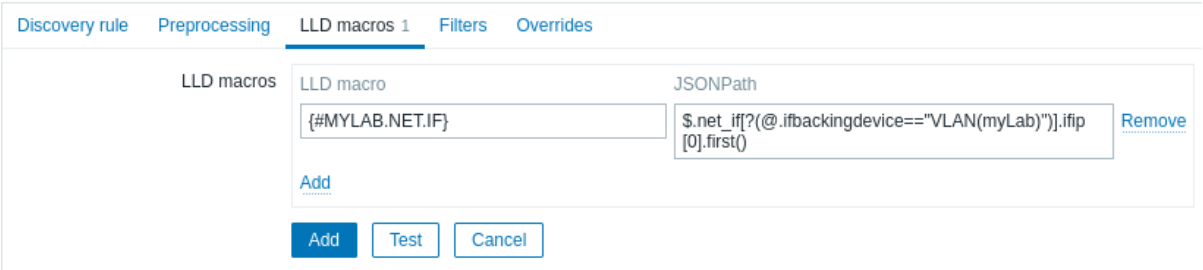
Ключ элемента данных `vmware.vm.discovery[{$VMWARE.URL}]`, настроенный в разделе *Создание правила низкоуровневого обнаружения*, возвращает данные интерфейсов узлов сети в поле "net_if":

```
"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": []
      }
    ]
```

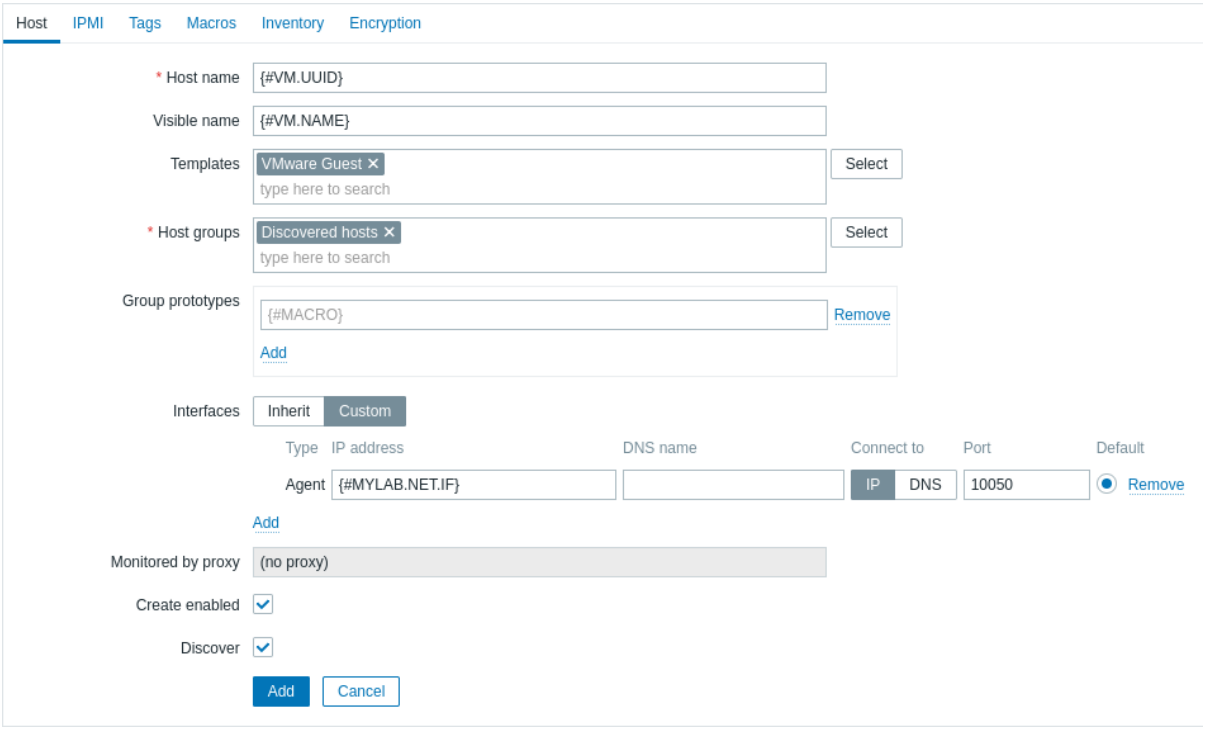
Эти данные могут быть использованы, чтобы настроить пользовательский интерфейс узла сети.

1. При создании правила низкоуровневого обнаружения дополнительно настройте макрос низкоуровневого обнаружения. На вкладке *LLD макросы (LLD macros)* создайте пользовательский LLD макрос со значением JSONPath. Например:

- `{#MYLAB.NET.IF}` — `$.net_if[?(@.ifbackingdevice=="VLAN(myLab)")].ifip[0].first()`




2. При создании прототипа узла сети добавьте пользовательский интерфейс узла сети и введите LLD макрос в поле *DNS имя (DNS name)* или *IP адрес (IP address)*.




## 11. Обслуживание

**Обзор** Режим обслуживания используется, чтобы подавить проблемы в течение заранее заданных временных периодов.

В Zabbix вы можете указать периоды обслуживания для узлов сети и групп узлов сети.

Более того, можно определить обслуживание только для одного триггера (или подмножества триггеров), указав теги триггера. В этом случае обслуживание будет активировано только для этих триггеров; все остальные триггеры узла сети или группы узлов сети не будут находиться в режиме обслуживания.

Имеется два типа обслуживания: *Со сбором данных* и *Без сбора данных*.

В течение обслуживания *Со сбором данных* триггеры обрабатываются как обычно и, если потребуется, создаются события. Однако, если в настройках действия выбрана опция *Приостановить операции для подавленных проблем (Pause operations for suppressed problems)*, то эскалации о проблемах будут поставлены на паузу для узлов сети/триггеров в обслуживании. В этом случае шаги эскалаций, которые могут включать отправку оповещений или удалённые команды, будут игнорироваться до тех пор, пока не завершится период обслуживания. Обратите внимание, что операции восстановления и обновления не подавляются в процессе обслуживания, только эскалации.

Например, если шаги эскалаций запланированы на 0, 30 и 60 минуты после начала проблемы и имеется получасовое обслуживание, которое начинается с 10 минуты по 40 минуту после возникновения реальной проблемы, то шаги второй и третий будут выполнены на полчаса позже, или спустя 60 минут и 90 минут (при условии, что проблема всё ещё имеется). Аналогично, если проблема произойдёт в течение обслуживания, эскалация начнется после завершения обслуживания.

Чтобы получать оповещения о проблемах в процессе обслуживания как обычно (без задержки), вам необходимо убрать опцию *Приостановить операции для подавленных проблем* в настройках действия.

**Note:**
Если хотя бы один узел сети (используемый в выражении триггера) не находится в режиме обслуживания, Zabbix отправит оповещение о проблеме.

Zabbix сервер должен работать во время обслуживания. Обслуживания пересчитываются каждую минуту или сразу после обновления кэша конфигурации, если есть изменения в периоде обслуживания.

Процессы таймера проверяют, нужно ли изменить состояния узлов сети в/без обслуживания в 0 секунду каждой минуты. Кроме того, каждую секунду процесс таймера проверяет, нужно ли запустить/остановить какие-либо обслуживания, основываясь на том, есть ли изменения в периодах обслуживания после обновления конфигурации. Таким образом, скорость запуска/остановки периодов обслуживания зависит от интервала обновления конфигурации (по умолчанию 10

секунд). Обратите внимание, что изменения периода обслуживания не включают настройки *Активно с/Активно до (Active since/Active till)*. Кроме того, если узел/группа узлов добавляется в существующий активный период обслуживания, изменения будут активированы процессом таймера только в начале следующей минуты.

Обратите внимание, когда узел сети входит в режим обслуживания, процессы таймера Zabbix сервера перечитают все открытые проблемы для проверки, требуется ли их подавить. Такое поведение может повлиять на производительность, если имеется большое количество открытых проблем. Zabbix сервер будет также читать все открытые проблемы при запуске процесса, даже если на это время отсутствуют настроенные обслуживания.

Обратите внимание, что Zabbix сервер (или прокси) всегда собирает данные, независимо от типа обслуживания (включая обслуживание *Без сбора данных*). Эти данные затем игнорируются сервером, если указан тип *Без сбора данных*.

После завершения обслуживания *Без сбора данных* триггеры, использующие функцию `nodata()`, не сработают до момента следующей проверки в течение периода времени обновления этих элементов данных.

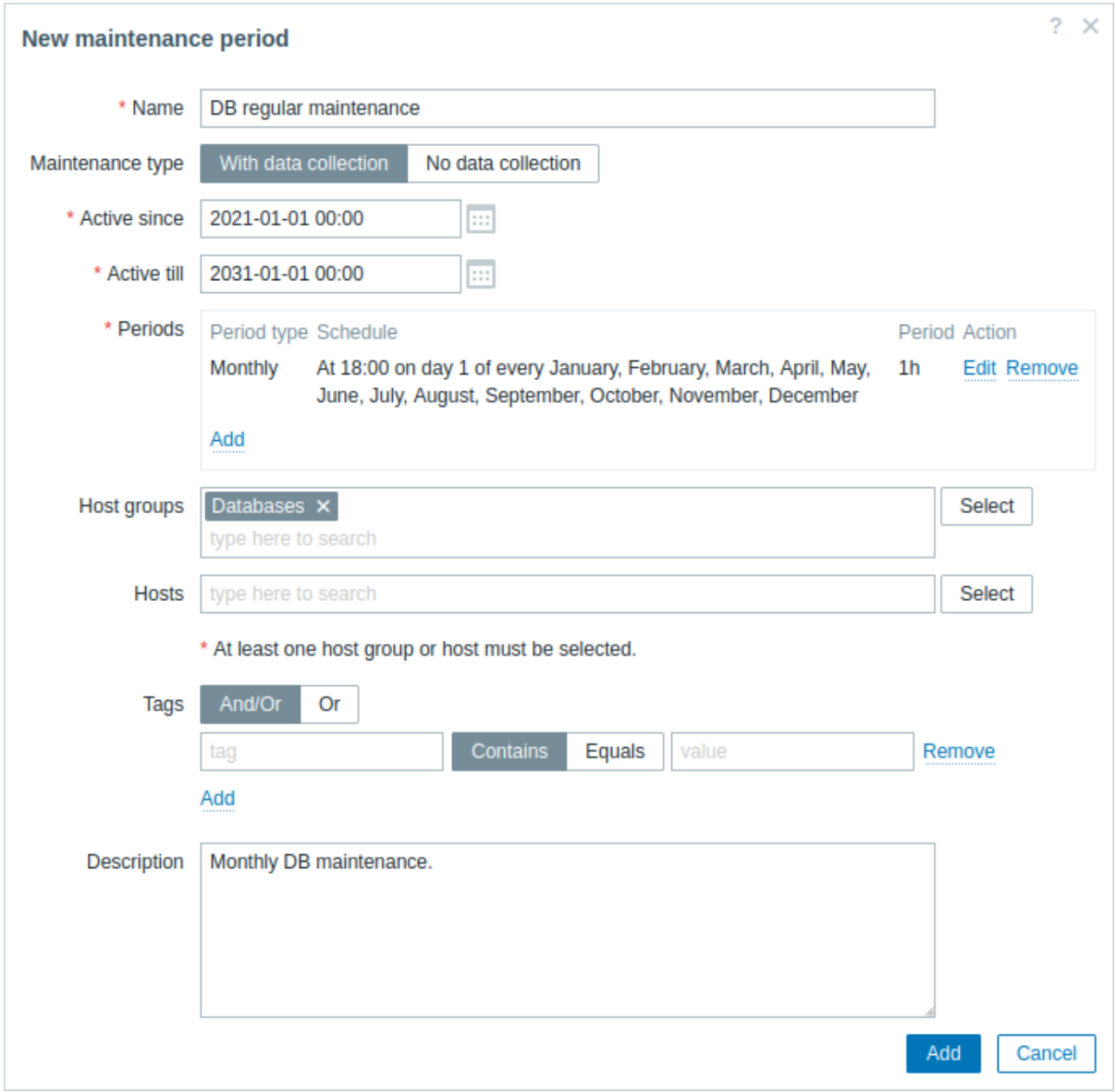
Если элемент данных журнала добавляется к узлу сети, который находится в обслуживании, и обслуживание заканчивается, то будут обработаны только новые записи в файле журнала, начиная с момента завершения обслуживания.

Если на узел сети, который находится в режиме обслуживания *Без сбора данных*, отсылается значение со штампом времени (например при использовании Zabbix sender), тогда это значение будет отброшено; однако, имеется возможность отправить значение со штампом времени за истекший период обслуживания, и оно будет принято.

Если период обслуживания, узлы сети, группы или теги изменяются пользователем, то такие изменения вступают в силу только после синхронизации кэша конфигурации.

**Конфигурация** Чтобы настроить период обслуживания:

1. Перейдите в: *Сбор данных* > *Обслуживание*.
2. Нажмите *Создать период обслуживания* (или на имя существующего периода обслуживания).
3. Введите параметры обслуживания в форме.



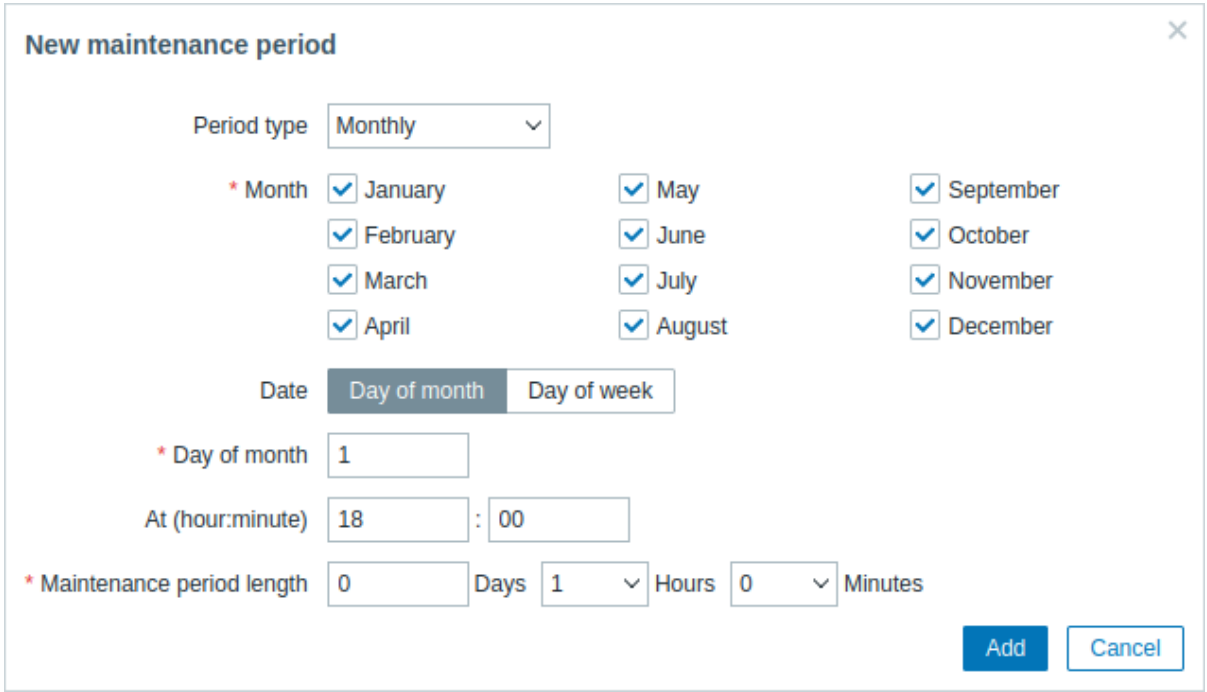

Все обязательные поля ввода отмечены красной звездочкой.

| Параметр | Описание |
|---|---|
| *Name* | Имя периода обслуживания. |
| *Maintenance type* | Можно задать два типа обслуживания:<br>**With data collection** - данные собираются сервером во время обслуживания, и триггеры обрабатываются;<br>**No data collection** - данные по-прежнему могут собираться, но во время обслуживания они не сохраняются в базу данных, и триггеры (включая функцию `nodata()`) не срабатывают.<br>См. Влияние периодов обслуживания, чтобы узнать, как каждый тип влияет на отчет о доступности. |
| *Active since* | Дата и время, с которых выполнение периодов обслуживания становится активным.<br>*Примечание:* Указание только этого времени не активирует период обслуживания; периоды обслуживания должны быть настроены в *Periods* (см. ниже). |
| *Active till* | Дата и время, до которых выполнение периодов обслуживания остается активным. |
| *Periods* | Этот блок позволяет задать точные дни и часы, когда выполняется обслуживание.<br>Нажатие Add открывает всплывающее окно с гибкой формой *Maintenance period*, где можно задать расписание обслуживания. Подробное описание см. в разделе Периоды обслуживания. |

| Параметр | Описание |
|---|---|
| *Host groups* | Выберите группы узлов сети, для которых будет активировано обслуживание.<br>Обслуживание будет активировано для всех узлов сети из указанных групп узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе в нем будет отображаться выпадающий список всех доступных групп узлов сети.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети. Таким образом, обслуживание также будет активировано для узлов сети из вложенных групп. |
| *Hosts* | Выберите узлы сети, для которых будет активировано обслуживание. Это поле поддерживает автодополнение, поэтому при вводе в нем будет отображаться выпадающий список всех доступных узлов сети. |
| *Tags* | Укажите теги для подавления проблем с совпадающими тегами на узлах сети в обслуживании.<br>Можно задать несколько условий. Сопоставление имен тегов всегда чувствительно к регистру.<br><br>Для каждого условия доступны два оператора:<br>**Contains** - включать указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, чувствителен к регистру);<br>**Equals** - включать указанные имена и значения тегов (чувствительно к регистру).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия.<br><br>Теги можно указать только если выбран тип обслуживания *With data collection*. |
| *Description* | Описание периода обслуживания. |

Периоды обслуживания

Диалог периода обслуживания предназначен для планирования времени периодического или однократного обслуживания. Доступные поля диалога меняются динамически на основе выбранного *Тип периода*.



| Тип периода | Описание |
|---|---|
| *Единовременно (One time only)* | Настройте одноразовый период обслуживания:<br>*Дата (Date)* — дата и время периода обслуживания;<br>*Длительность периода обслуживания (Maintenance period length)* — как долго будет активно обслуживание. |
| *Ежедневно (Daily)* | Настройте ежедневный период обслуживания:<br>*Каждый день(и) (Every day(s))* — частота обслуживания (1 *(по умолчанию)* — каждый день, 2 — каждые два дня и т.д.);<br>*В (час:минута) (At (hour:minute))* — время суток начала обслуживания;<br>*Длительность периода обслуживания (Maintenance period length)* — как долго обслуживание будет активно.<br><br>Если параметр *Каждый день(дни)* больше «1», то начальным днём будет день, на который приходится время *Активно с*. Примеры:<br>- если *Активно с* установлено на «2021-01-01 12:00», *Каждый день(дни)* установлено на «2», а *В (час:минута)* установлено на «23:00», то первый период обслуживания начнётся 1 января в 23:00, а второй период обслуживания начнётся 3 января в 23:00;<br>- если *Активно с* установлено на «2021-01-01 12:00», *Каждый день(дни)* установлено на «2», а *В (час:минута)* установлено на «01:00», то первый период обслуживания начнётся 3 января в 01:00, а второй период обслуживания начнётся 5 января в 01:00. |
| *Еженедельно (Weekly)* | Настройте еженедельный период обслуживания:<br>*Каждую неделю(и) (Every week(s))* — частота обслуживания (1 *(по умолчанию)* — каждую неделю, 2 — каждую вторую неделю и т.д.);<br>*День недели (Day of week)* — в какой день недели обслуживание должно выполняться;<br>*В (час:минута) (At (hour:minute))* — время суток начала обслуживания;<br>*Длительность периода обслуживания (Maintenance period length)* — как долго обслуживание будет активно.<br><br>Если параметр *Каждую неделю(и)* больше «1», то начальной неделей будет неделя, на которую приходится время *Активно с*. Примеры смотрите в описании параметра *Ежедневно* выше. |
| *Ежемесячно (Monthly)* | Настройте ежемесячный период обслуживания:<br>*Месяц (Month)* — выберите все месяцы, в течение которых будет выполняться регулярное обслуживание;<br>*Дата (Date)*: **День месяца (Day of month)** — выберите эту опцию, если обслуживание проводится в одно и то же число месяца (например, каждый 1-й день месяца), а затем выберите требуемое число в появившемся поле *День месяца (Day of month)*.<br>*Дата (Date)*: **День недели (Day of week)** — выберите эту опцию, если обслуживание проводится только в определенные дни недели (например, каждый первый понедельник месяца), а затем в выпадающем списке выберите требуемую неделю месяца (первая, вторая, третья, четвертая или последняя) и отметьте дни недели обслуживания;<br>*В (час:минута) (At (hour:minute))* — время суток начала обслуживания.<br>*Длительность периода обслуживания (Maintenance period length)* — как долго обслуживание будет активно. |

**Attention:**
При создании периода обслуживания используется часовой пояс пользователя, который его создает. Однако при планировании повторяющихся периодов обслуживания (*Daily*, *Weekly*, *Monthly*) используется часовой пояс сервера Zabbix. Чтобы обеспечить предсказуемое поведение повторяющихся периодов обслуживания, необходимо использовать единый часовой пояс для всех компонентов Zabbix.

Когда закончите, нажмите *Add*, чтобы добавить период обслуживания в блок *Periods*.

Обратите внимание, что изменения времени на летнее/зимнее время (DST) не влияют на длительность обслуживания. Например, предположим, что настроено двухчасовое обслуживание, которое обычно начинается в 01:00 и заканчивается в 03:00:

- если после одного часа обслуживания (в 02:00) происходит переход на летнее/зимнее время и текущее время меняется с 02:00 на 03:00, обслуживание продолжится еще один час (до 04:00);
- если после двух часов обслуживания (в 03:00) происходит переход на летнее/зимнее время и текущее время меняется с 03:00 на 02:00, обслуживание остановится, поскольку прошло два часа;

- если период обслуживания начинается в час, который пропускается из-за перехода на летнее/зимнее время, то обслуживание не начнется.

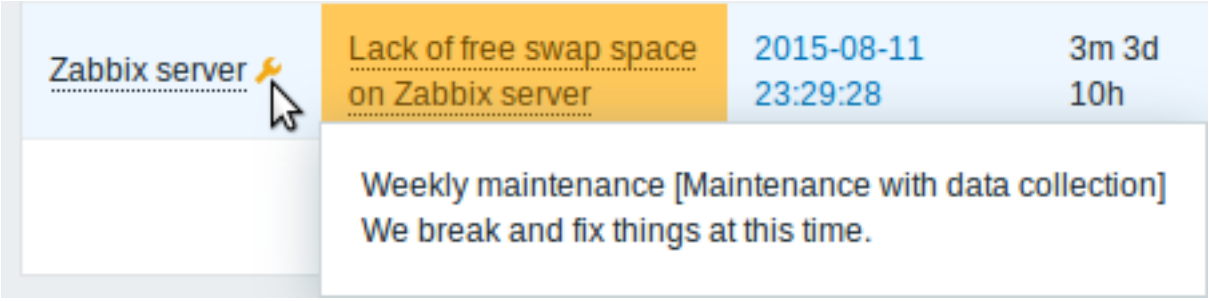
Если период обслуживания задан как "1 day" (фактическая длительность обслуживания составляет 24 часа, поскольку Zabbix рассчитывает дни в часах), начинается в 00:00 и заканчивается в 00:00 следующего дня:

- обслуживание остановится в 01:00 следующего дня, если текущее время сдвинется вперед на один час;
- обслуживание остановится в 23:00 этого дня, если текущее время сдвинется назад на один час.

**Просмотр** Отображение узлов сети в обслуживании

Значок с оранжевым гаечным ключом вблизи имени узла сети означает, что этот узел сети находится в обслуживании в следующих разделах:

- *Панели (Dashboards)*
- *Мониторинг → Проблемы (Monitoring → Problems)*
- *Инвентаризация → Узлы сети → Детали инвентарных данных узла сети (Inventory → Hosts → Host inventory details)*
- *Сбор данных → Узлы сети (Data collection → Hosts)* (Смотрите столбец «Состояние (Status)»)




Детали обслуживания отображаются при наведении на значок курсора мыши.

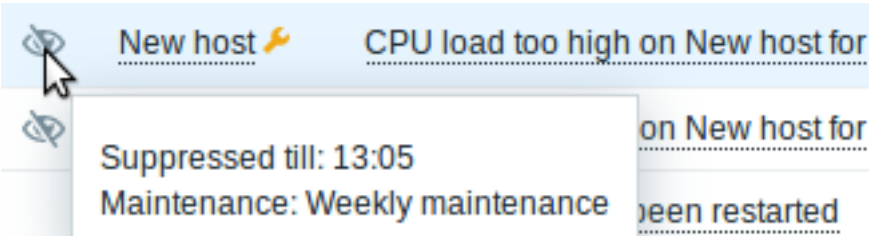
Кроме этого, у узлов сети в обслуживании появляется оранжевый фон в *Мониторинг → Карты сети (Monitoring → Maps)*.

Отображение подавленных проблем

Обычно проблемы с узлов сети в обслуживании подавляются, то есть не отображаются в веб-интерфейсе. Однако, можно настроить и так, чтобы при выборе опции *Отображение подавленных проблем (Show suppressed problems)* подавленные проблемы отображались в следующих местах:

- *Панели (Dashboards)* (при настройке виджетов *Узлы сети с проблемами*, *Проблемы*, *Проблемы по важности*, *Обзор триггеров*)
- *Мониторинг → Проблемы (Monitoring → Problems)* (в фильтре)
- *Мониторинг → Карты сети (Monitoring → Maps)* (в настройках карты)
- Глобальные оповещения (в настройках профиля пользователя)

При отображении подавленных проблем отображается следующий значок: . При наведении курсора мыши отобразятся более подробные сведения:




**Расчёт очередей во время обслуживания** Очереди, отображаемые в веб-интерфейсе Zabbix (Администрирование → Очередь (Administration → Queue)), рассчитываются Zabbix сервером. Они не включают элементы данных, находящиеся в режиме обслуживания *Без сбора данных* — для этих элементов данных длина очереди всегда равна нулю, даже если их значения задерживаются. Задержанные элементы данных в режиме обслуживания *Со сбором данных* по-прежнему учитываются в очереди.

Zabbix прокси не знает о периодах обслуживания, поскольку конфигурация обслуживания не синхронизируется между Zabbix сервером и прокси. Внутренние проверки, посчитанные на Zabbix прокси (например, `zabbix[queue,,]` и `zabbix[stats,,,queue,,]`), сообщают о задержанных элементах данных независимо от состояния обслуживания на Zabbix сервере.

В результате для одних и тех же элементов данных в режиме обслуживания *Без сбора данных* веб-интерфейсом Zabbix и внутренними проверками на Zabbix прокси могут сообщаться разные длины очередей.

## 12. Регулярные выражения

**Обзор** Perl Compatible Regular Expressions (PCRE, PCRE2) поддерживаются в Zabbix.

В Zabbix есть два способа использования регулярных выражений:

- вручную ввести регулярное выражение
- использовать глобальное регулярное выражение, созданное в Zabbix

**Регулярные выражения** Вы можете вручную ввести регулярное выражение в поддерживаемых местах. Обратите внимание, что выражение не может начинаться с @, поскольку этот символ используется в Zabbix для ссылки на глобальные регулярные выражения.

> **Attention:**
> При использовании регулярных выражений возможно исчерпание памяти стека. Дополнительные сведения см. в PCRE PERFORMANCE.

Обратите внимание, что при сопоставлении в многострочном режиме якоря ^ и $ соответствуют началу и концу каждой строки соответственно, а не началу и концу всей строки.

См. также примеры правильного экранирования в различных контекстах.

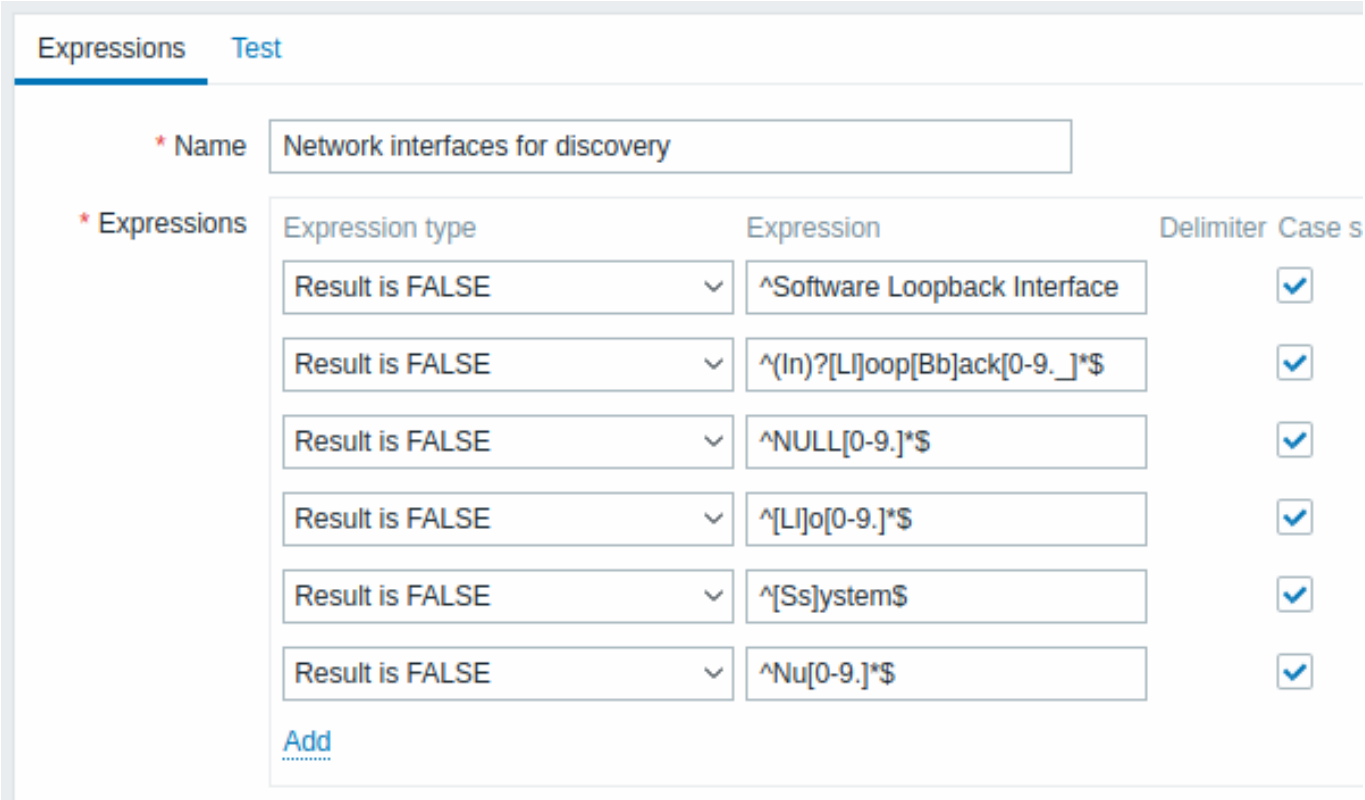
**Глобальные регулярные выражения** В веб-интерфейсе Zabbix есть расширенный редактор для создания и тестирования сложных регулярных выражений.

После того как регулярное выражение создано таким способом, его можно использовать в нескольких местах веб-интерфейса, обращаясь к его имени с префиксом @, например, *@mycustomregexp*.

Чтобы создать глобальное регулярное выражение:

- Перейдите в: *Administration* > *General*
- Выберите *Regular expressions* в выпадающем списке
- Нажмите *New regular expression*

Вкладка **Expressions** позволяет задать имя регулярного выражения и добавить подвыражения.



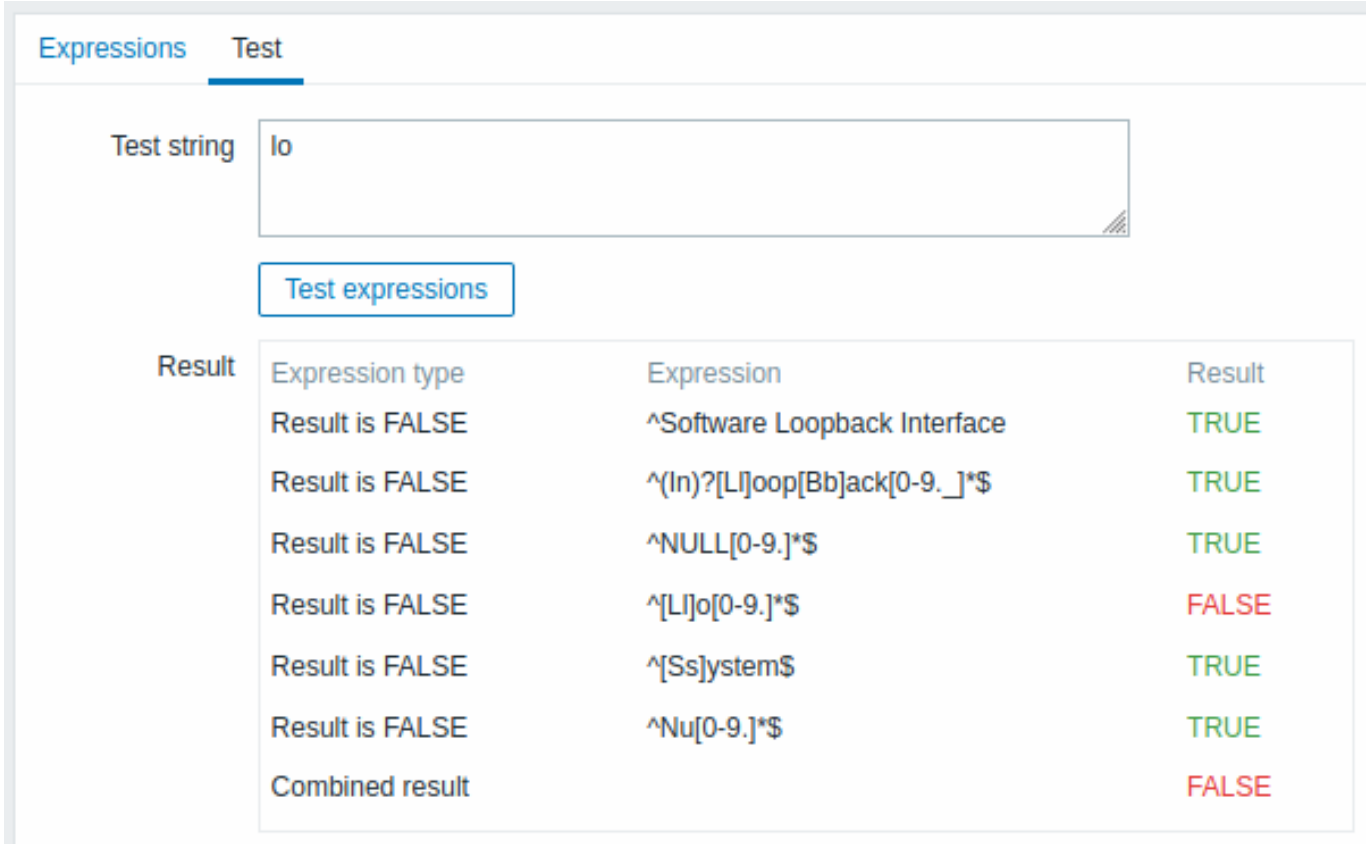

Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Name* | Задайте имя регулярного выражения. Допускаются любые символы Unicode. |
| *Expressions* | Нажмите *Add* в блоке Expressions, чтобы добавить новое подвыражение. |
| *Expression type* | Выберите тип выражения:<br>**Character string included** - совпадение с подстрокой<br>**Any character string included** - совпадение с любой подстрокой из списка, разделенного разделителями. В качестве разделителей используются запятая (,), точка (.) или косая черта (/).<br>**Character string not included** - совпадение с любой строкой, кроме подстроки<br>**Result is TRUE** - совпадение с регулярным выражением<br>**Result is FALSE** - не совпадать с регулярным выражением |
| *Expression* | Введите подстроку/регулярное выражение. |
| *Delimiter* | Запятая (,), точка (.) или косая черта (/) для разделения текстовых строк в регулярном выражении. Этот параметр активен только при выборе типа выражения *"Any character string included"*. |
| *Case sensitive* | Флажок, указывающий, чувствительно ли регулярное выражение к регистру букв. |

Косая черта (/) в выражении воспринимается буквально, а не как разделитель. Таким образом можно сохранять выражения, содержащие косую черту, без ошибок.

**Attention:**
Пользовательское имя регулярного выражения в Zabbix может содержать запятые, пробелы и т. д. В тех случаях, когда это может привести к неверной интерпретации при ссылке на него (например, запятая в параметре ключа элемента данных), всю ссылку можно заключить в кавычки, например: *"@My custom regexp for purpose1, purpose2"*.<br> В других местах имена регулярных выражений не должны заключаться в кавычки (например, в свойствах правила LLD).

На вкладке **Test** регулярное выражение и его подвыражения можно проверить, указав тестовую строку.




Результаты показывают состояние каждого подвыражения и общее состояние пользовательского выражения.

Общее состояние пользовательского выражения определяется как *Combined result*. Если определено несколько подвыражений, Zabbix использует логический оператор AND для вычисления *Combined result*. Это означает, что если хотя бы один Result имеет значение False, то *Combined result* также будет иметь статус False.

**Регулярные выражения по умолчанию для глобальных правил** Zabbix поставляется с несколькими глобальными регулярными выражениями в наборе данных по умолчанию.

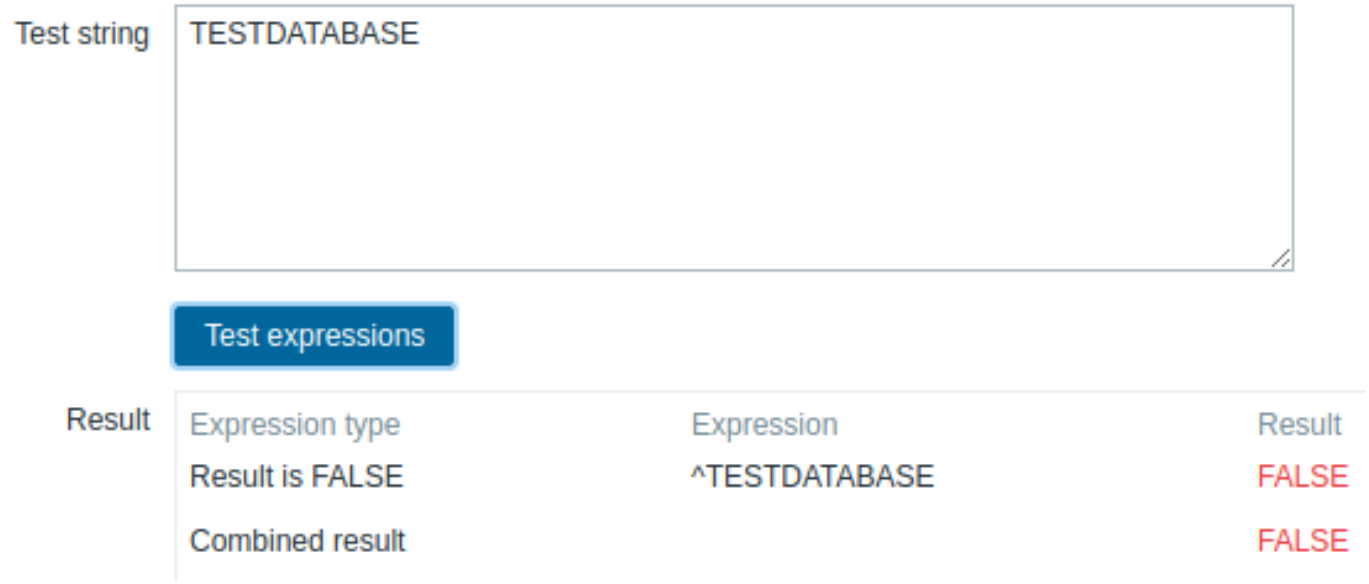
| Name | Expression | Matches |
|---|---|---|
| *Файловые системы для обнаружения* | `^(btrfs\|ext2\|ext3\|ext4\|jfs\|reiser\|xfs\|ffs\|ufs\|jfs\|jfs2\|vxfs\|hfs\|refs\|apfs\|ntfs\|fat32\|zfs)$` | ”btrfs” или ”ext2” или ”ext3” или ”ext4” или ”jfs” или ”reiser” или ”xfs” или ”ffs” или ”ufs” или ”jfs” или ”jfs2” или ”vxfs” или ”hfs” или ”refs” или ”apfs” или ”ntfs” или ”fat32” или ”zfs” |
| *Сетевые интерфейсы для обнаружения* | `^Software Loopback Interface` | Строки, начинающиеся с ”Software Loopback Interface”. |
| | `^lo$` | ”lo” |
| | `^(In)?[Ll]oop[Bb]ack[0-9._]*$` | Строки, которые необязательно начинаются с ”In”, затем содержат ”L” или ”l”, затем ”oop”, затем ”B” или ”b”, затем ”ack”, после чего может необязательно следовать любое количество цифр, точек или символов подчеркивания. |
| | `^NULL[0-9.]*$` | Строки, начинающиеся с ”NULL”, за которыми необязательно следует любое количество цифр или точек. |
| | `^[Ll]o[0-9.]*$` | Строки, начинающиеся с ”Lo” или ”lo” и за которыми необязательно следует любое количество цифр или точек. |
| | `^[Ss]ystem$` | ”System” или ”system” |
| | `^Nu[0-9.]*$` | Строки, начинающиеся с ”Nu”, за которыми необязательно следует любое количество цифр или точек. |

| Name | Expression | Matches |
|---|---|---|
| *Устройства хранения для обнаружения по SNMP* | `^(Physical memory\|Virtual memory\|Memory buffers\|Cached memory\|Swap space)$` | ”Physical memory” или ”Virtual memory” или ”Memory buffers” или ”Cached memory” или ”Swap space” |
| *Имена служб Windows для обнаружения* | `^(MMCSS\|gupdate\|SysmonLog\|clr_optimization_v4.0.30319_32)$` | ”MMCSS”, ”gupdate”, ”SysmonLog” и строки вида ”clr_optimization_v2.0.50727_32” и ”clr_optimization_v4.0.30319_32”, где вместо точек можно использовать любой символ, кроме символа новой строки. |
| *Состояния запуска служб Windows для обнаружения* | `^(automatic\|automatic delayed)$` | ”automatic” или ”automatic delayed” |

**Примеры** Пример 1

Использование следующего выражения в низкоуровневом обнаружении для обнаружения баз данных, кроме базы данных с определенным именем:

```
^TESTDATABASE$
```

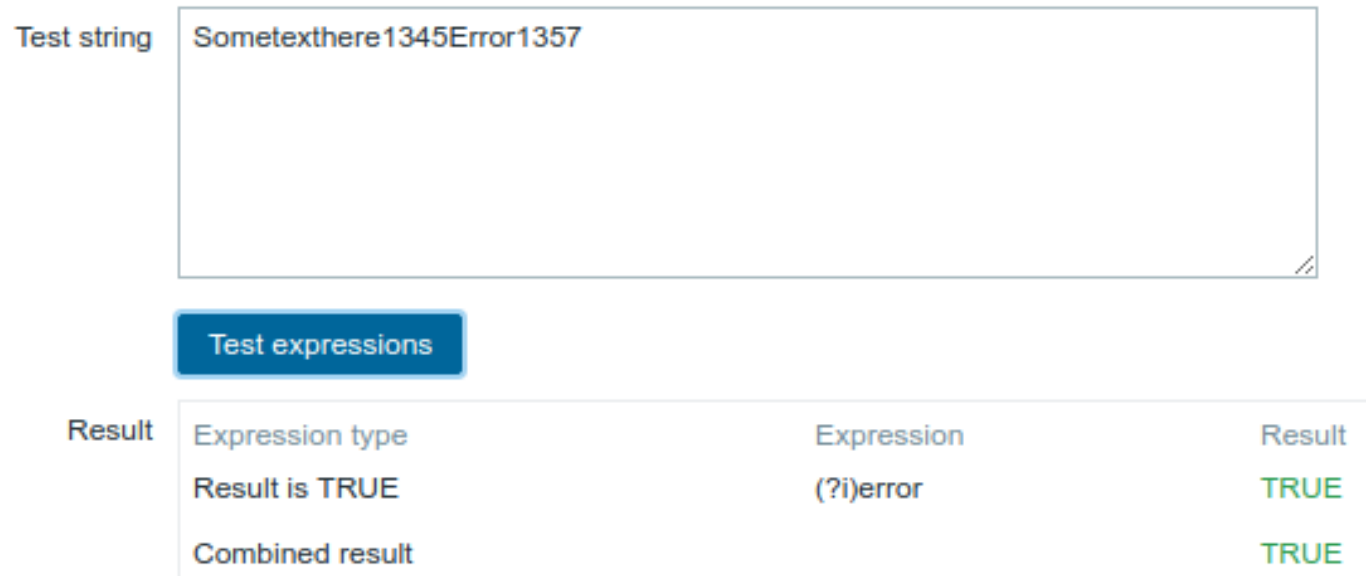



Выбранный *тип выражения*: ”Результат — FALSE”. Не соответствует имени, содержащему строку ”*TESTDATABASE*”.

Пример с модификатором регулярного выражения в строке

Использование следующего регулярного выражения, включая встроенный модификатор (?i), для сопоставления символов ”error”:

```
(?i)error
```




Выбранный *тип выражения*: ”Результат — TRUE”. Символы ”error” сопоставляются.

Еще один пример с модификатором регулярного выражения в строке

Использование следующего регулярного выражения, включая несколько встроенных модификаторов, для сопоставления символов после определенной строки:

```
(?<=match (?i)everything(?-i) after this line\n)(?sx).*# we add s modifier to allow . match newline charac
```

Выбранный тип выражения: ”Result is TRUE”. Символы после определенной строки сопоставляются.

**Attention:**
Модификатор **g** нельзя указать в строке. Список доступных модификаторов можно найти на странице man pcresyntax. Дополнительные сведения о синтаксисе PCRE см. в HTML-документации PCRE.

**Поддержка регулярных выражений по расположению**

| Расположение | Регулярное выражение | Глобальное регулярное выражение | Многострочное сопоставление | Комментарии |
|---|---|---|---|---|
| **Элементы данных агента** | | | | |
| eventlog[] | Yes | Yes | Yes | параметры `regexp`, `severity`, `source`, `eventid` |
| eventlog.count[] | | | | параметры `regexp`, `severity`, `source`, `eventid` |
| log[] | | | | параметр `regexp` |
| log.count[] | | | | |

| Расположение | Регулярное выражение | Глобальное регулярное выражение | Многострочное сопоставление | Комментарии |
|---|---|---|---|---|
| logrt[] | | Yes/No | | параметр `regexp` поддерживает оба варианта, параметр `file_regexp` поддерживает только неглобальные выражения |
| logrt.count[] | | | | |
| proc.cpu.util[] | | No | No | параметр `cmdline` |
| proc.get[] | | | | |
| proc.mem[] | | | | |
| proc.num[] | | | | |
| sensor[] | | | | параметры `device` и `sensor` в Linux 2.4 |
| system.hw.macaddr[] | | | | параметр `interface` |
| system.sw.packages[] | | | | параметр `regexp` |
| system.sw.packages.get[] | | | | параметр `regexp` |
| vfs.dir.count[] | | | | параметры `regex_incl`, `regex_excl`, `regex_excl_dir` |
| vfs.dir.get[] | | | | параметры `regex_incl`, `regex_excl`, `regex_excl_dir` |
| vfs.dir.size[] | | | | параметры `regex_incl`, `regex_excl`, `regex_excl_dir` |
| vfs.file.regexp[] | | | Yes | параметр `regexp` |
| vfs.file.regmatch[] | | | | |
| web.page.regexp[] | | | | |
| **SNMP traps** | | | | |
| snmptrap[] | Yes | Yes | No | параметр `regexp` |
| **Предварительная обработка значения элемента данных** | Yes | No | No | параметр `pattern` |
| **Функции для триггеров/вычисляемых элементов данных** | | | | |
| count() | Yes | Yes | Yes | параметр `pattern`, если параметр `operator` имеет значение *regexp* или *iregexp* |
| countunique() | Yes | Yes | | |
| find() | Yes | Yes | | |
| logeventid() | Yes | Yes | No | параметр `pattern` |
| logsource() | | | | |
| **Низкоуровневое обнаружение** | | | | |
| Filters | Yes | Yes | No | поле *Regular expression* |
| Overrides | Yes | No | | в условиях *matches*, *does not match* для *Operation* |
| **Условия действий** | Yes | No | No | в условиях автосоздания *matches*, *does not match* для *Host name* и *Host metadata* |
| **Скрипты** | Yes | Yes | No | поле *Input validation rule* |
| **Веб-мониторинг** | Yes | No | Yes | *Variables* с префиксом **regex:** поле *Required string* |
| **Контекст пользовательского макроса** | Yes | No | No | в контексте макроса с префиксом **regex:** |
| **Функции макросов** | | | | |
| regsub() | Yes | No | No | параметр `pattern` |

| Расположение | Регулярное выражение | Глобальное регулярное выражение | Многострочное сопоставление | Комментарии |
|---|---|---|---|---|
| iregsub() | | | | |
| **Индикаторы ссылок на картах** | Yes | No | No | поле *Pattern* (для текстовых элементов данных) |
| **Сопоставление значков** | Yes | Yes | No | поле *Expression* |
| **Сопоставление значений** | Yes | No | No | поле *Value*, если тип сопоставления — `regexp` |

## 13. Подтверждение проблем

**Обзор** Проблемные события в Zabbix могут быть подтверждены пользователями.

Если пользователь получает уведомление о проблемном событии, он может перейти в веб-интерфейс Zabbix, открыть всплывающее окно обновления проблемы для этого события одним из способов, перечисленных ниже, и подтвердить проблему. При подтверждении он может добавить комментарий, например указать, что работает над ней, или написать что-либо еще по своему усмотрению.

Таким образом, если другой пользователь системы обнаружит ту же проблему, он сразу увидит, была ли она подтверждена, а также все комментарии на данный момент.

Таким образом, процесс устранения проблем, в котором участвует более одного пользователя системы, может выполняться согласованно.

Статус подтверждения также используется при определении операций действия. Например, можно задать, чтобы уведомление отправлялось менеджеру более высокого уровня только в том случае, если событие не было подтверждено в течение некоторого времени.

Чтобы подтверждать события и оставлять к ним комментарии, пользователь должен иметь как минимум права на чтение соответствующих триггеров. Чтобы изменить серьезность проблемы или закрыть проблему, пользователь должен иметь права на чтение и запись для соответствующих триггеров.

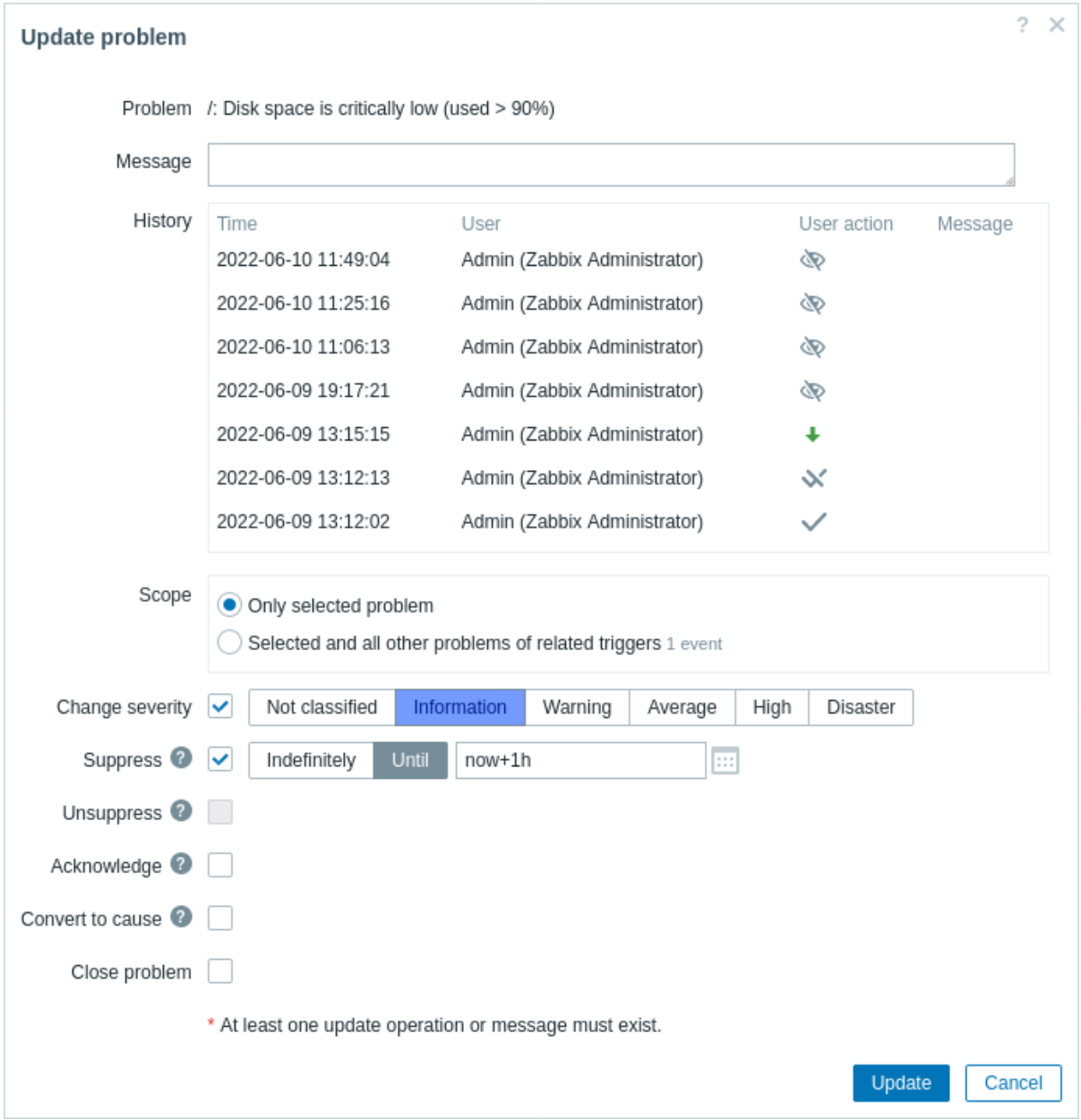
Существует **несколько** способов открыть всплывающее окно обновления проблемы, которое позволяет подтвердить проблему.

- Можно выбрать проблемы в *Monitoring* > *Problems*, а затем нажать *Mass update* под списком
- Можно нажать *Update* в столбце *Update* у проблемы в:
  - *Dashboards* (виджеты *Problems* и *Problems by severity*)
  - *Monitoring* > *Problems*
  - *Monitoring* > *Problems* > *Event details*
- Можно нажать на ячейку с нерешенной проблемой в:
  - *Dashboards* (виджет *Trigger overview*)

Всплывающее меню содержит пункт *Update*, который перенаправит вас в окно обновления проблемы.

**Обновление проблем** Всплывающее окно обновления проблемы позволяет:

- добавить комментарий к проблеме
- просмотреть комментарии и действия на данный момент
- изменить серьезность проблемы
- подавить/снять подавление проблемы
- подтвердить/снять подтверждение проблемы
- изменить проблему-симптом на проблему-причину
- закрыть проблему вручную




Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Problem* | Если выбрана только одна проблема, отображается ее название.<br>Если выбрано несколько проблем, отображается *выбрано N проблем*. |
| *Message* | Введите текст комментария к проблеме (максимум 2048 символов). |
| *History* | Здесь перечислены предыдущие действия и комментарии по проблеме, а также время и сведения о пользователе.<br>Значение значков, используемых для обозначения действий пользователя, см. на странице деталей события.<br>Обратите внимание, что история отображается, только если для обновления выбрана одна проблема. |
| *Scope* | Определите область действия таких операций, как изменение серьезности, подтверждение или ручное закрытие проблем:<br>**Только выбранная проблема** - повлияет только на это событие<br>**Выбранная и все другие проблемы связанных триггеров** - в случае подтверждения/закрытия проблемы повлияет на это событие и на все другие проблемы, которые еще не подтверждены/не закрыты. Если область содержит проблемы, которые уже подтверждены или закрыты, повторно подтверждать/закрывать их не будут. С другой стороны, количество операций по изменению сообщения и серьезности не ограничено. |

| Parameter | Description |
|---|---|
| *Change severity* | Установите флажок и нажмите кнопку серьезности, чтобы обновить серьезность проблемы.<br>Флажок для изменения серьезности доступен, если для хотя бы одной из выбранных проблем есть права на чтение и запись. При нажатии *Update* будут обновлены только те проблемы, для которых разрешены чтение и запись.<br>Если права на чтение и запись отсутствуют ни для одного из выбранных триггеров, флажок недоступен. |
| *Suppress* | Установите флажок, чтобы подавить проблему:<br>**Indefinitely** - подавить на неопределенный срок<br>**Until** - подавить до указанного времени. Поддерживаются оба формата времени: абсолютный и относительный, например:<br>`now+1d` - на один день от текущего момента (по умолчанию)<br>`now/w` - до конца текущей недели<br>`2022-05-28 12:00:00` - до указанной даты/времени<br>Обратите внимание, что простой период (например, `1d`, `1w`) не поддерживается.<br>Доступность этой опции зависит от настроек роли пользователя "Suppress problems".<br>См. также: Подавление проблем |
| *Unsuppress* | Установите флажок, чтобы снять подавление проблемы. Этот флажок активен только если хотя бы одна из выбранных проблем подавлена.<br>Доступность этой опции зависит от настроек роли пользователя "Suppress problems". |
| *Acknowledge* | Установите флажок, чтобы подтвердить проблему.<br>Этот флажок доступен, если среди выбранных есть хотя бы одна неподтвержденная проблема.<br>Невозможно добавить еще одно подтверждение для уже подтвержденной проблемы (однако можно добавить еще один комментарий). |
| *Unacknowledge* | Установите флажок, чтобы снять подтверждение проблемы.<br>Этот флажок доступен, если среди выбранных есть хотя бы одна подтвержденная проблема. |
| *Convert to cause* | Установите флажок, чтобы преобразовать проблему-симптом(ы) в проблему(ы)-причину. |
| *Close problem* | Установите флажок, чтобы вручную закрыть выбранную(ые) проблему(ы).<br>Флажок для закрытия проблемы доступен, если в конфигурации триггера для хотя бы одной из выбранных проблем включена опция *Allow manual close*. При нажатии *Update* будут закрыты только те проблемы, для которых разрешено закрытие.<br>Если ни одну проблему нельзя закрыть вручную, флажок недоступен.<br>Уже закрытые проблемы повторно закрываться не будут. |

**Отображение** На основе информации о подтверждении можно настроить отображение количества проблем на панели или картах. Чтобы сделать это, вам необходимо выбрать в опции *Отображение проблем (Problem display)*, доступной как в настройках карты, так и в виджете панели *Проблемы по степени важности (Problems by severity)*. Можно отобразить все проблемы, количество неподтверждённых проблем отдельно от общего количества или только количество неподтверждённых проблем.

На основе информации об обновлении проблемы (подтверждение и т.д.) можно настроить операции обновления — отправить сообщение или выполнить удалённые команды.

#### 1. Подавление проблем

Обзор

Подавление проблем предлагает способ временно скрыть проблему, с которой можно разобраться позже. Это полезно для очистки списка проблем, чтобы придать наивысший приоритет наиболее срочным проблемам. Например, иногда на выходных может возникнуть проблема, которая недостаточно срочная, чтобы ее решать незамедлительно, поэтому ее можно «отложить» до утра понедельника.

Подавление проблем позволяет скрыть *единственную* проблему, в отличие от подавления проблем посредством обслуживания узла сети, когда скрываются все проблемы находящегося в обслуживании узла сети.

Операции по триггерным действиям будут приостановлены для подавленных проблем так же, как это делается с помощью обслуживания узла сети.

Настройка

Проблему можно подавить через окно **обновление проблемы (problem update)**, где подавление является одним из вариантов обновления проблемы наряду с комментированием, изменением важности, подтверждением и т. д.

Подавление проблемы также можно отменить через то же окно обновления проблемы.

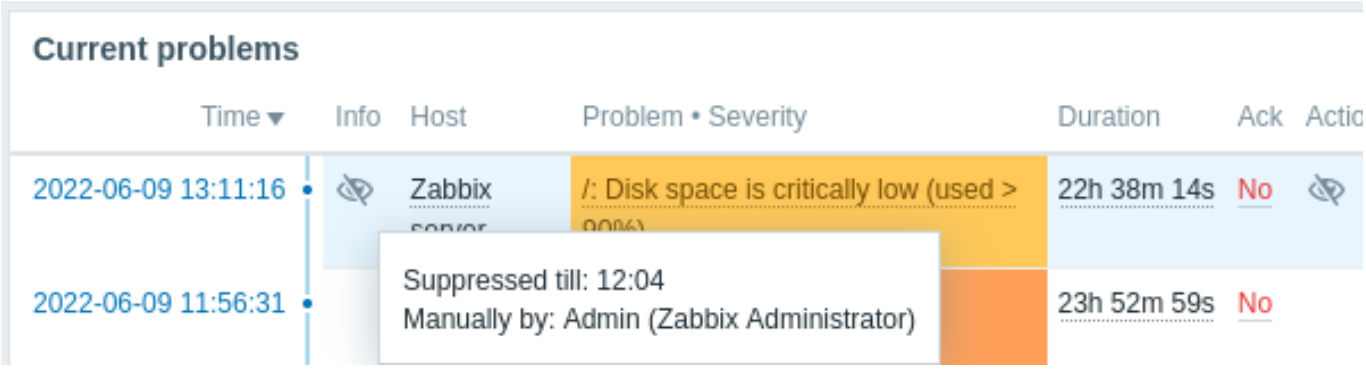
Отображение

После подавления проблема помечается мигающим значком подавления в столбце *Инфо (Info)*, прежде чем будет скрыта.

Значок подавления мигает, пока задача подавления находится в списке ожидания. Как только диспетчер задач подавит проблему, значок перестанет мигать. Если значок подавления продолжает мигать в течение длительного времени, это может указывать на проблему с сервером, например, если сервер не работает и диспетчер задач не может выполнить задачу. Та же логика применима и к отмене подавления. В течение короткого периода времени после отправки задачи, пока сервер её не выполнил, значок отмены подавления мигает.

Подавленная проблема может быть как скрытой, так и отображённой, в зависимости от настроек фильтра/виджета проблем.

При отображении в списке проблем подавленная проблема отмечается значком подавления, а подробности подавления отображаются при наведении указателя мыши:




Подробности о подавлении также отображаются во всплывающем окне при наведении указателя мыши на значок подавления в столбце *Действия (Actions)*.

## 14. Экспорт/импорт настроек

**Обзор** Функционал экспорта/импорта настроек в Zabbix делает возможным выполнять обмен объектами конфигурации между одной Zabbix системой и другой.

Типичные случаи применения этого функционала:

- обмен шаблонами и картами сети — пользователи Zabbix могут обмениваться своими параметрами конфигурации
- размещение шаблона на странице Zabbix Community templates [en]. Другие люди затем могут загрузить этот шаблон себе и импортировать в Zabbix.
- интеграция с прикладными инструментами — универсальные форматы YAML, XML и JSON делают возможным интеграцию и импорт/экспорт данных со сторонними инструментами и приложениями.

Что можно экспортировать/импортировать

Объекты, которые можно экспортировать/импортировать:

- Группы узлов сети (*только через Zabbix API*)
- Группы шаблонов (*только через Zabbix API*)
- Шаблоны
- Узлы сети
- Карты сети
- Способы оповещения
- Изображения

Формат экспорта

Данные можно экспортировать при помощи веб-интерфейса Zabbix или Zabbix API. Поддерживаемые форматы экспорта: YAML, XML и JSON.

**Дополнительная информация об экспорте**

- Все поддерживаемые элементы экспортируются в один файл.
- Объекты на узлах сети и в шаблонах (элементы данных, триггеры, графики, правила обнаружения), которые унаследованы из присоединённых шаблонов, не экспортируются. Любые изменения, выполненные над этими объектами на уровне узла сети (например, изменённый интервал обновления, изменённое регулярное выражение или прототипы, добавленные к правилу низкоуровневого обнаружения), при экспорте будут потеряны; при импорте все объекты из присоединённых шаблонов будут пересозданы такими, как они есть в исходном присоединённом шаблоне.
- Объекты, созданные низкоуровневым обнаружением, и любые зависимые от них объекты не экспортируются. Например, если триггер создан для элемента данных, сгенерированного правилом обнаружения, то такой триггер не будет экспортирован.
- Если экспортируемый узел сети/шаблон содержит объекты, поддерживающие тайм-ауты, значения тайм-аутов будут экспортированы, если для этих объектов настроены собственные тайм-ауты.

**Дополнительная информация об импорте**

- Импорт останавливается при первой ошибке.
- При обновлении существующих изображений при импорте изображений поле «imagetype» игнорируется, т.е. изменить тип изображения посредством импорта невозможно.
- При импорте хостов/шаблонов с использованием опции «Удалить отсутствующие» макросы хоста/шаблона, отсутствующие в файле импорта, будут удалены из хоста/шаблона после импорта.
- Пустые теги элементов данных, триггеров, графиков, правил обнаружения, прототипов элементов данных, прототипов триггеров, прототипов графиков бессмысленны, т.к. они имеют такой же смысл, как если бы были не указаны вовсе.
- Если для объектов импортированного хоста/шаблона настроены собственные тайм-ауты, они будут применены; в противном случае будут применены тайм-ауты прокси/глобальные тайм-ауты.
- Импорт поддерживает YAML, XML и JSON, импортируемый файл должен иметь корректное расширение файла: .yaml и .yml для YAML, .xml для XML и .json для JSON. Смотрите информацию о совместимости касательно поддерживаемых версий XML.
- Импорт поддерживает конфигурационные файлы только в кодировке UTF-8 (как с, так и без маркера последовательности байтов (BOM)); другие кодировки (UTF16LE, UTF16BE, UTF32LE, UTF32BE, etc.) приведут к ошибке конвертации при импорте.

**Базовый формат YAML** Формат экспорта YAML содержит следующие узлы:

- Корневой узел для экспорта Zabbix в YAML
- Версия экспорта

```
zabbix_export:
  version: '7.4'
```

Другие узлы зависят от экспортируемых объектов.

**XML-формат** Формат экспорта XML содержит следующие теги:

- Стандартный заголовок для XML-документов
- Корневой тег для XML-экспорта Zabbix
- Версия экспорта

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>7.4</version>
</zabbix_export>
```

Другие теги зависят от экспортируемых объектов.

**Формат JSON** Формат экспорта JSON содержит следующие объекты:

- Корневой объект для экспорта Zabbix JSON
- Версия экспорта

```
{
    "zabbix_export": {
        "version": "7.4"
    }
}
```

Другие объекты зависят от экспортируемых объектов.

**1. Группы шаблонов**

Обзор

В веб-интерфейсе группы шаблонов могут быть экспортированы только при экспорте шаблона. При экспорте шаблона все группы, к которым он принадлежит, автоматически экспортируются вместе с ним.

API позволяет экспортировать группы шаблонов независимо от шаблонов.

Формат экспорта

```
template_groups:
  - uuid: 36bff6c29af64692839d077febfc7079
    name: 'Network devices'
```

Экспортируемые элементы

| Элемент | Тип | Описание |
|---|---|---|
| uuid | *строка* | Уникальный идентификатор этой группы шаблонов. |
| name | *строка* | Имя группы. |

**2. Группы узлов сети**

Обзор

В веб-интерфейсе Zabbix группы узлов сети можно экспортировать только с экспортом узла сети. При экспорте узла сети все группы, к которым он принадлежит, автоматически экспортируются вместе с ним.

API позволяет экспортировать группы узлов сети независимо от узлов сети.

Формат экспорта

```
host_groups:
  - uuid: 6f6799aa69e844b4b3918f779f2abf08
    name: 'Zabbix servers'
```

Экспортируемые элементы

| Элемент | Тип | Описание |
|---|---|---|
| uuid | *строка* | Уникальный идентификатор этой группы узлов сети. |
| name | *строка* | Имя группы. |

**3. Шаблоны**

Обзор

Шаблоны экспортируются вместе со множеством связанных объектов и со связями между объектами.

Экспорт шаблона содержит:

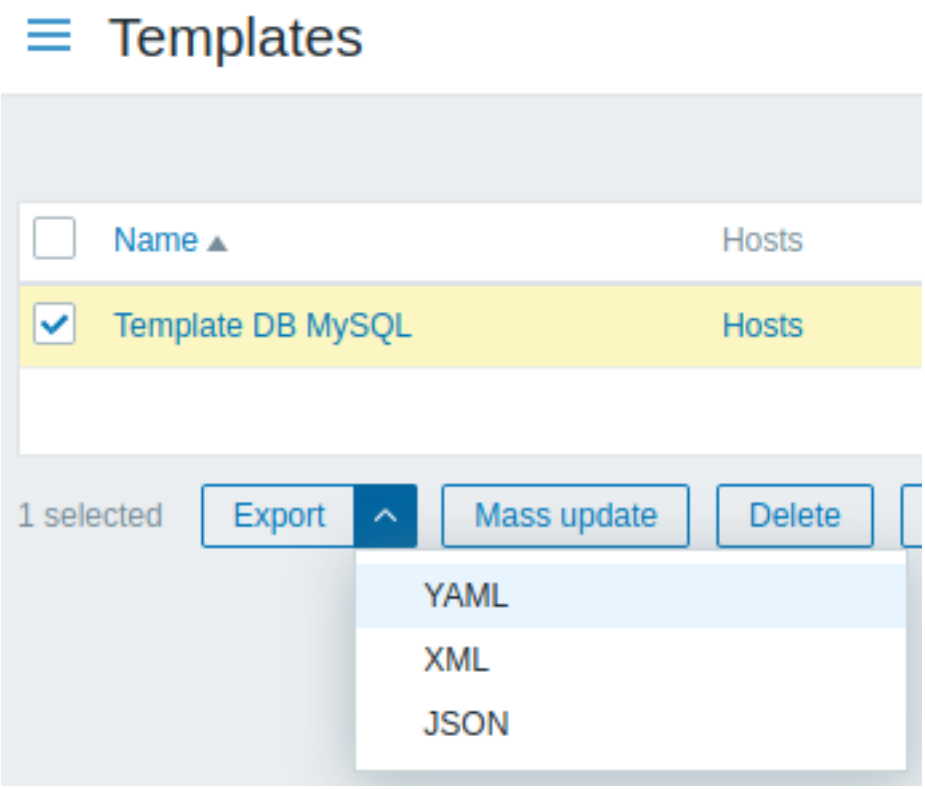
- Присоединённые группы шаблонов
- Присоединённые группы узлов сети (если используются в настройках прототипа узла сети)
- Данные шаблона
- Связи с другими шаблонами
- Связи с группами шаблонов
- Напрямую присоединённые элементы данных
- Напрямую присоединённые триггеры
- Напрямую присоединённые графики

- Напрямую присоединённые панели
- Напрямую присоединённые правила обнаружения со всеми прототипами
- Напрямую присоединённые веб-сценарии
- Преобразования значений

Экспорт

Для экспорта шаблонов выполните следующее:

1. Перейдите к: *Сбор данных* → *Шаблоны (Data collection* → *Templates)*
2. Отметьте шаблоны, которые нужно экспортировать
3. Нажмите на *Экспорт (Export)* ниже списка



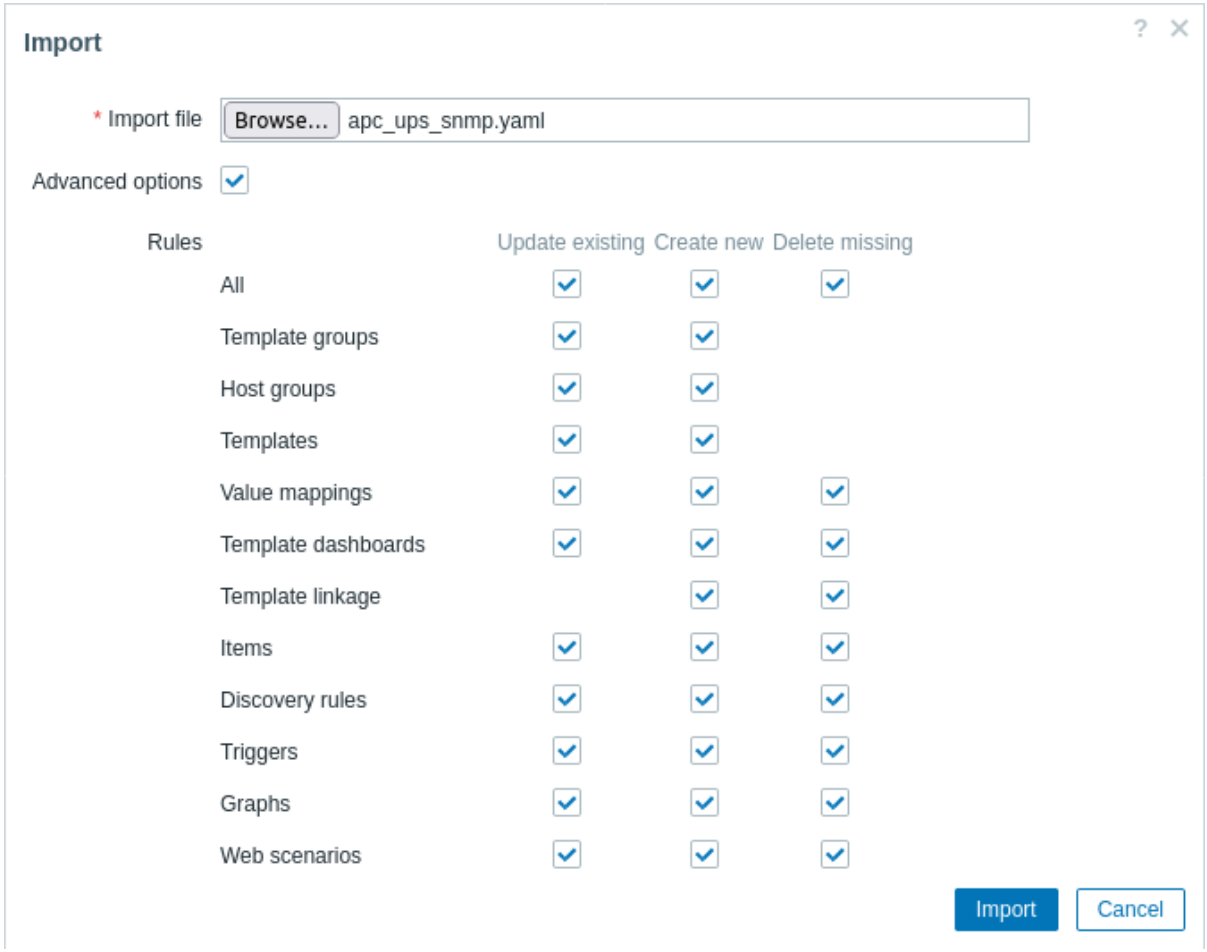

В зависимости от выбранного формата шаблоны будут экспортированы в локальный файл со следующим именем по умолчанию:

- *zabbix_export_templates.yaml* — при экспорте в YAML (опция экспорта по умолчанию);
- *zabbix_export_templates.xml* — при экспорте в XML;
- *zabbix_export_templates.json* — при экспорте в JSON.

Импорт

Чтобы импортировать шаблоны, выполните следующие действия:

1. Перейдите в *Сбор данных* → *Шаблоны*.
2. Нажмите *Импорт* в правом верхнем углу.
3. Выберите файл импорта.
4. Нажмите *Импорт* в правом нижнем углу формы настройки.




Если установить флажок *Дополнительные параметры*, будет отображен подробный список всех элементов, доступных для импорта — отметьте или снимите отметку для каждого правила импорта по необходимости.

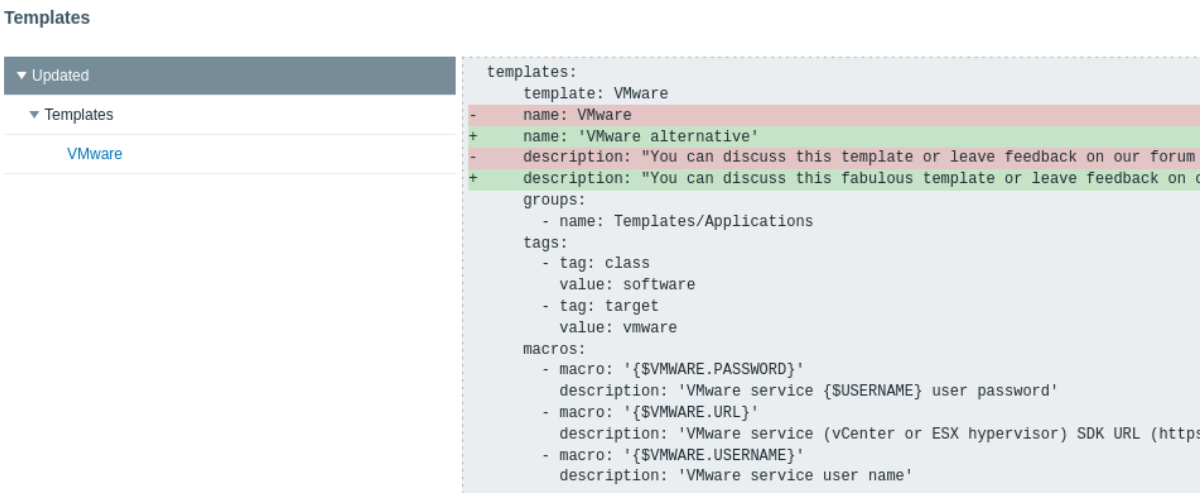
Если щелкнуть флажок в строке *Все*, все элементы ниже будут отмечены или сняты.

Правила импорта:

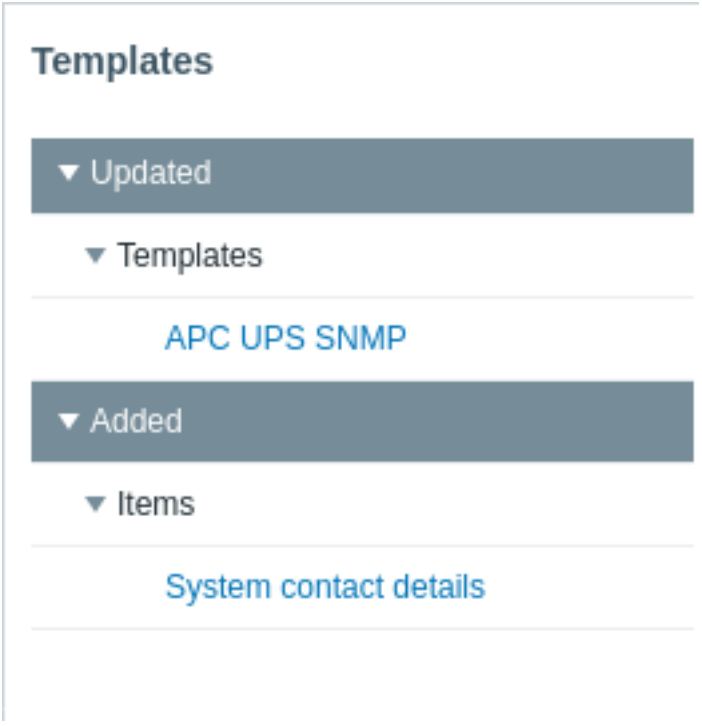
| Правило | Описание |
|---|---|
| *Обновить существующие* | Существующие элементы будут обновлены с использованием данных из файла импорта. В противном случае они не будут обновлены. |
| *Создать новые* | Новые элементы будут созданы с использованием данных из файла импорта. В противном случае они не будут созданы. |
| *Удалить отсутствующие* | Существующие элементы, отсутствующие в файле импорта, будут удалены. В противном случае они не будут удалены.<br>Если для *Связь шаблонов* отмечено *Удалить отсутствующие*, текущая связь шаблонов, отсутствующая в файле импорта, будет разорвана. Наследуемые от отвязанных шаблонов сущности (элементы данных, триггеры, графики и т. д.) не будут удалены (если только для каждой такой сущности также не выбрана опция *Удалить отсутствующие*). |

**Attention:**
Если шаблоны с теми же именами уже существуют, при импорте следует отметить параметры *Удалить отсутствующие*, чтобы выполнить корректный импорт. Так будут удалены старые элементы, которые больше не присутствуют в обновленном шаблоне (учтите, что это приведет к потере истории этих старых элементов).

На следующем экране вы сможете просмотреть содержимое импортируемого шаблона. Если это новый шаблон, все элементы будут выделены зеленым цветом. При обновлении существующего шаблона новые элементы шаблона будут выделены зеленым цветом; удаленные элементы шаблона будут выделены красным; элементы, которые не изменились, будут отображаться на сером фоне.




Меню слева можно использовать для навигации по списку изменений. Раздел *Обновлено* выделяет все изменения, внесенные в существующие элементы шаблона. Раздел *Добавлено* содержит список новых элементов шаблона. Элементы в каждом разделе сгруппированы по типу элемента; нажмите серую стрелку, чтобы развернуть или свернуть группу элементов.




Проверьте изменения шаблона, а затем нажмите *Импорт*, чтобы выполнить импорт шаблона. В веб-интерфейсе будет отображено сообщение об успешном завершении или ошибке импорта.

**Формат экспорта** Когда шаблон экспортируется, Zabbix генерирует структурированный формат YAML, JSON или XML. Экспорт включает элементы шаблона, такие как метаданные шаблона, элементы данных, макросы, триггеры, дашборды и многое другое.

Каждый элемент выполняет определенную функцию и может содержать вложенные элементы.

В следующих разделах описан каждый элемент в формате экспорта. В примерах используется шаблон Linux by Zabbix agent.

Многоточие `(...)` обозначает элементы, опущенные для краткости. Примечание `(see table below)` используется, когда элемент объясняется более подробно в последующих разделах.

```
zabbix_export:
  version: '7.4'
  template_groups:
    - uuid: 846977d1dfed4968bc5f8bdb363285bc
```

```
    name: 'Templates/Operating systems'
  templates: ( .        )
  triggers: ( .        )
  graphs: ( .        )
```

**Метаданные шаблона**

| Element | Type | Description |
|---|---|---|
| version | string | (обязательно) Версия экспорта. |
| template_groups | | (обязательно) Корневой элемент для групп шаблонов. |
| uuid | string | (обязательно) Уникальный идентификатор для этой группы шаблонов. |
| name | string | (обязательно) Имя группы шаблонов. |
| host_groups | | Корневой элемент для групп узлов сети, которые используются прототипами узлов сети. |
| uuid | string | (обязательно) Уникальный идентификатор для этой группы узлов сети. |
| name | string | (обязательно) Имя группы узлов сети. |
| templates | | Корневой элемент для шаблонов. |
| triggers | | Корневой элемент для элементов триггеров шаблона, которые совпадают с триггерами элементов данных шаблона. |
| graphs | | Корневой элемент для элементов графиков шаблона, которые совпадают с графиками узлов сети. |

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      template: 'Linux by Zabbix agent'
      name: 'Linux by Zabbix agent'
      description: |
                    Linux.            Zabbix agent 7.4       . (...)
      wizard_ready: 'YES'
      readme: |
        ##

                    Linux.            Zabbix agent 7.4       . (...)
      vendor:
        name: Zabbix
        version: 7.4-2
      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: '{$AGENT.TIMEOUT}'
          value: 3m
          description: '  -  ,                          .                ,            /    Zabbix (
          config: (see table below)
        - macro: '{$CPU.UTIL.CRIT}'
          value: '90'
          description: '               CPU,         %.'
          config: (see table below)
        (...)
      dashboards: (see table below)
      valuemaps: (see table below)
    (...)
```

**Шаблоны**

| Element | Type | Description |
|---|---|---|
| uuid | string | (required) Уникальный идентификатор этого шаблона. |
| template | string | (required) Уникальное имя шаблона. |
| name | string | Отображаемое имя шаблона. |
| description | text | Описание шаблона. |
| wizard_ready | text | Доступен ли шаблон для выбора в Мастере узлов сети.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1).<br>См. также: Объект Template (`wizard_ready`). |
| readme | text | Инструкции по настройке, относящиеся к шаблону, для отображения в Мастере узлов сети. Поддерживает форматирование Markdown. |
| vendor | | Корневой элемент для поставщика шаблона (присутствует, если экспортированный шаблон содержит данные о поставщике). |
| name | string | (required) Имя поставщика шаблона. |
| version | string | (required) Версия шаблона.<br>Для готовых шаблонов версия отображается следующим образом: основная версия Zabbix, разделитель (”-”), номер ревизии (увеличивается с каждой новой версией шаблона и сбрасывается с каждой основной версией Zabbix). Например, 7.0-0, 7.0-3, 7.4-0, 7.4-3. |
| templates | | Корневой элемент для связанных шаблонов. |
| name | string | (required) Имя шаблона. |
| groups | | Корневой элемент для групп шаблонов. |
| name | string | (required) Имя группы шаблонов. |
| items | | Корневой элемент для элементов данных шаблона. |
| discovery_rules | | Корневой элемент для правил низкоуровневого обнаружения шаблона. |
| httptests | | Корневой элемент для веб-сценариев шаблона. |
| tags | | Корневой элемент для тегов шаблона. |
| tag | string | (required) Имя тега. |
| value | string | Значение тега. |
| macros | | Корневой элемент для пользовательских макросов шаблона. |
| macro | string | (required) Имя пользовательского макроса. |
| type | string | Тип пользовательского макроса.<br>Возможные значения:[1] TEXT (0, по умолчанию), SECRET_TEXT (1), VAULT (2).<br>См. также: Объект User macro (`type`). |
| value | string | Значение пользовательского макроса. |
| description | string | Описание пользовательского макроса. |
| config | | Корневой элемент для конфигурации макроса шаблона, отвечающей за то, как макрос отображается в Мастере узлов сети. |
| dashboards | | Корневой элемент для панелей шаблона. |
| valuemaps | | Корневой элемент для карт значений шаблона. |

Элементы шаблона

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        - uuid: f94f9f4699e94c369e6c98b2a2f485ce
          name: '       Zabbix'
          key: agent.ping
          description: '              "1"                .                        `nodata()`                .
          valuemap:
            name: '            Zabbix'
          tags:
            - tag: component
              value: system
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          name: '                   '
          key: system.sw.packages.get
```

```
      delay: 1h
      preprocessing: ( .        )
      tags:
        - tag: component
          value: os
      triggers: ( .        )
    - uuid: 403cebed115441369e94d35d070ca7b8
      name: '           '
      type: DEPENDENT
      key: vm.memory.utilization
      value_type: FLOAT
      units: '%'
      description: '                          `100-pavailable`.'
      preprocessing: ( .        )
      master_item:
        key: 'vm.memory.size[pavailable]'
      tags:
        - tag: component
          value: memory
      triggers: ( .        )
    (...)
```

| Элемент | Тип | Описание |
|---|---|---|
| uuid | string | (обязательно) Уникальный идентификатор этого элемента данных. |
| name | string | (обязательно) Имя элемента данных. |
| type | string | Тип элемента данных.<br>Возможные значения:[1] ZABBIX_PASSIVE (0, по умолчанию), 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). |
| snmp_oid | string | (обязательно для элементов данных SNMP_AGENT) Идентификатор объекта SNMP. |
| key | string | (обязательно) Ключ элемента данных. |
| delay | string | Интервал обновления элемента данных.<br>По умолчанию: 1m. Для элементов данных TRAP значение всегда будет 0. |
| history | string | Период времени (с использованием суффикса времени, пользовательского макроса или LLD-макроса), в течение которого должны храниться данные истории.<br>По умолчанию: 31d. |
| trends | string | Период времени (с использованием суффикса времени, пользовательского макроса или LLD-макроса), в течение которого должны храниться данные трендов.<br>По умолчанию: 365d. |
| status | string | Статус элемента данных.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1). |
| value_type | string | Тип получаемого значения.<br>Возможные значения:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, по умолчанию), TEXT (4), BINARY (5). |
| allowed_hosts | string | Список IP-адресов узлов сети, разделенных запятыми, которым разрешено отправлять данные для элемента данных.<br>Поддерживается для элементов данных TRAP и HTTP_AGENT. |
| units | string | Единицы получаемого значения (bps, B, и т. д.). |
| params | text | Дополнительные параметры в зависимости от типа элемента данных (выполняемый скрипт для элементов данных SSH и TELNET; SQL-запрос для элементов данных ODBC; формула для элементов данных CALCULATED; скрипт для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER). |
| ipmi_sensor | string | Датчик IPMI.<br>Поддерживается для элементов данных IPMI. |

| Элемент | Тип | Описание |
|---|---|---|
| authtype | string | Тип аутентификации.<br>Поддерживается для элементов данных SSH и HTTP_AGENT.<br>Возможные значения для элементов данных SSH:[1] PASSWORD (0, по умолчанию), PUBLIC_KEY (1).<br>Возможные значения для элементов данных HTTP_AGENT:[1] NONE (0, по умолчанию), BASIC (1), NTLM (2), Kerberos (3) или Digest (4). |
| username | string | (обязательно для элементов данных SSH и TELNET) Имя пользователя для аутентификации.<br>Поддерживается для элементов данных SIMPLE, ODBC, JMX и HTTP_AGENT.<br>При использовании для элементов данных JMX также следует указать `password` (см. ниже) либо оставить оба элемента пустыми. |
| password | string | (обязательно для элементов данных SSH и TELNET) Пароль для аутентификации.<br>Поддерживается для элементов данных SIMPLE, ODBC, JMX и HTTP_AGENT.<br>При использовании для элементов данных JMX также следует указать `username` (см. выше) либо оставить оба элемента пустыми. |
| publickey | string | (обязательно для элементов данных SSH) Имя файла открытого ключа. |
| privatekey | string | (обязательно для элементов данных SSH) Имя файла закрытого ключа. |
| description | text | Описание элемента данных. |
| inventory_link | string | Поле инвентаря узла сети, которое заполняется этим элементом данных.<br>Возможные значения:[1] NONE (0), ALIAS (4) и т. д. (поддерживаемые поля см. в разделе Инвентарь узла сети). |
| valuemap | | Корневой элемент для карт значений элемента данных. |
| name | string | (обязательно) Имя карты значений, используемой для элемента данных. |
| logtimefmt | string | Формат времени в записях журнала.<br>Поддерживается для элементов данных типа LOG. |
| preprocessing | | Корневой элемент для предварительной обработки значений элемента данных. |
| step | | Корневой элемент для шагов предварительной обработки значений элемента данных шаблона. |
| jmx_endpoint | string | Конечная точка JMX.<br>Поддерживается для элементов данных JMX. |
| master_item | | (обязательно для элементов данных DEPENDENT) Корневой элемент для master item зависимого элемента данных. |
| key | string | (обязательно) Ключ master item зависимого элемента данных. |
| timeout | string | Тайм-аут запроса опроса данных элемента данных.<br>Поддерживается для списка типов элементов данных Тайм-ауты. |
| url | string | (обязательно для элементов данных HTTP_AGENT) Строка URL. |
| query_fields | | Корневой элемент для параметров запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |
| name | string | (обязательно для элементов данных HTTP_AGENT) Имя параметра запроса. |
| value | string | Значение параметра запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |
| parameters | | Корневой элемент для пользовательских параметров.<br>Поддерживается для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER. |
| name | string | (обязательно для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER) Имя пользовательского параметра. |
| value | string | Значение пользовательского параметра.<br>Поддерживается для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER. |
| posts | string | Данные тела HTTP(S)-запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми.<br>Поддерживается для элементов данных HTTP_AGENT. |
| follow_redirects | string | Следовать перенаправлениям ответа при опросе данных.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| post_type | string | Тип тела данных POST.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] RAW (0, по умолчанию), JSON (2), XML (3). |
| http_proxy | string | Строка подключения к HTTP(S)-прокси.<br>Поддерживается для элементов данных HTTP_AGENT. |
| headers | | Корневой элемент для заголовков HTTP(S)-запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |

| Элемент | Тип | Описание |
|---|---|---|
| name | string | (обязательно для элементов данных HTTP_AGENT) Имя заголовка. |
| value | string | (обязательно для элементов данных HTTP_AGENT) Значение заголовка. |
| retrieve_mode | string | Какая часть ответа должна быть сохранена.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] BODY (0, по умолчанию), HEADERS (1), BOTH (2). |
| request_method | string | Тип метода запроса.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] GET (0, по умолчанию), POST (1), PUT (2), HEAD (3). |
| output_format | string | Как обрабатывать ответ.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] RAW (0, по умолчанию), JSON (1). |
| allow_traps | string | Разрешить заполнение значения аналогично элементу данных trapper.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br>Поддерживается для элементов данных HTTP_AGENT. |
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br>Поддерживается для элементов данных HTTP_AGENT. |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br>Поддерживается для элементов данных HTTP_AGENT. |
| verify_peer | string | Проверять ли подлинность сертификата узла сети.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| verify_host | string | Проверять ли, что имя узла для подключения совпадает с именем в сертификате узла сети.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| tags | | Корневой элемент для тегов элемента данных. |
| tag | string | (обязательно) Имя тега. |
| value | string | Значение тега. |
| triggers | | Корневой элемент для триггеров элемента данных шаблона. |

**Note:**
См. также: Объект элемента данных (см. соответствующее свойство с совпадающим именем).

Шаги предварительной обработки значения элемента данных шаблона

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          preprocessing:
            - type: JSONPATH
              parameters:
                - $.length()
            - type: DISCARD_UNCHANGED_HEARTBEAT
              parameters:
                - 12h
          (...)
```

| Element | Type | Description |
|---|---|---|
| type | string | (обязательно) Тип шага предварительной обработки значения элемента данных. Возможные значения:[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, вычисляется: полученное значение - предыдущее значение), CHANGE_PER_SECOND (10, вычисляется: (полученное значение - предыдущее значение)/(текущее время - время последней проверки)), 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 | | (обязательно) Корневой элемент для параметров шага предварительной обработки значения элемента данных. |
| parameter | string | Отдельный параметр шага предварительной обработки значения элемента данных. |
| error_handler | string | Тип действия, используемый в случае сбоя шага предварительной обработки. Возможные значения:[1] ORIGINAL_ERROR (0, по умолчанию), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3). |
| error_handler_params | string | Параметры обработчика ошибок. |

**Note:**
См. также: Объект предварительной обработки элемента данных (см. соответствующее свойство с совпадающим именем).

Триггеры шаблона

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          triggers:
            - uuid: b950c306394f4b3c902060a8273cbcde
              expression: 'change(/Linux by Zabbix agent/system.sw.packages.get)<>0'
              name: 'Linux:                          '
              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)>{$MEMORY.UTIL.MAX}'
              name: 'Linux:                  '
              event_name: 'Linux:                  (>{$MEMORY.UTIL.MAX}%   5m)'
              priority: AVERAGE
              description: '                       .'
              dependencies:
                - name: 'Linux:                '
                  expression: 'max(/Linux by Zabbix agent/vm.memory.size[available],5m)<{$MEMORY.AVAILABLE
              tags:
                - tag: scope
                  value: capacity
                - tag: scope
```

```
                value: performance
        (...)
```

| Element | Type | Description |
|---|---|---|
| uuid | string | (обязательно) Уникальный идентификатор этого триггера. |
| expression | string | (обязательно) Выражение триггера. |
| recovery_mode | string | Основа для генерации событий OK.<br>Возможные значения:[1] EXPRESSION (0, по умолчанию), RECOVERY_EXPRESSION (1), NONE (2). |
| recovery_expression | string | Выражение восстановления триггера. |
| correlation_mode | string | Режим корреляции (без корреляции событий или корреляция событий по тегу).<br>Возможные значения:[1] DISABLED (0, по умолчанию), TAG_VALUE (1). |
| correlation_tag | string | Имя тега, используемое для корреляции событий. |
| name | string | (обязательно) Имя триггера. |
| event_name | string | Имя события. |
| opdata | string | Оперативные данные. |
| url_name | string | Метка для URL, связанного с триггером. |
| url | string | URL, связанный с триггером. |
| status | string | Статус триггера.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1). |
| priority | string | Критичность триггера.<br>Возможные значения:[1] NOT_CLASSIFIED (0, по умолчанию), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5). |
| description | text | Описание триггера. |
| type | string | Тип генерации событий (одно проблемное событие или несколько проблемных событий).<br>Возможные значения:[1] SINGLE (0, по умолчанию), MULTIPLE (1). |
| manual_close | string | Ручное закрытие проблемных событий.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| dependencies | | Корневой элемент для зависимостей. |
| name | string | (обязательно) Имя зависимого триггера. |
| expression | string | (обязательно) Выражение зависимого триггера. |
| recovery_expression | string | Выражение восстановления зависимого триггера. |
| tags | | Корневой элемент для тегов триггера. |
| tag | string | (обязательно) Имя тега. |
| value | string | Значение тега. |

**Note:**
См. также: Объект Trigger (см. соответствующее свойство с совпадающим именем).

Правила низкоуровневого обнаружения шаблона

```
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:**
Большинство элементов правил низкоуровневого обнаружения шаблона совпадают с элементами элементов данных шаблона. В таблице ниже описаны только те элементы, которые отличаются от элементов данных шаблона.

| Element | Type | Description |
|---|---|---|
| type | string | Тип элемента данных.<br>Возможные значения:[1] ZABBIX_PASSIVE (0, по умолчанию), 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 | (обязательно) Ключ правила низкоуровневого обнаружения. |
| filter | | Корневой элемент для фильтров правил низкоуровневого обнаружения шаблона. |
| lifetime | string | Период времени (в секундах, суффикс времени или пользовательский макрос), по истечении которого обнаруженные ресурсы, которые больше не обнаруживаются, будут удалены.<br>По умолчанию: 7d. |
| lifetime_type | string | Сценарий удаления потерянных ресурсов LLD.<br>Возможные значения: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER. |
| enabled_lifetime | string | Период времени (в секундах, суффикс времени или пользовательский макрос), по истечении которого обнаруженные ресурсы, которые больше не обнаруживаются, будут отключены. |
| enabled_lifetime_type | string | Сценарий отключения потерянных ресурсов LLD.<br>Возможные значения: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER. |
| item_prototypes | | Корневой элемент для элементов-прототипов шаблона, которые совпадают с элементами данных шаблона. |
| trigger_prototypes | | Корневой элемент для триггеров-прототипов шаблона, которые совпадают с триггерами элементов данных шаблона. |
| graph_prototypes | | Корневой элемент для графиков-прототипов шаблона, которые совпадают с графиками узла сети. |
| host_prototypes | | Корневой элемент для прототипов узлов сети шаблона, которые совпадают с узлами сети. |
| parent_discovery_rule | | Корневой элемент для родительского правила низкоуровневого обнаружения (или прототипа правила) прототипа правила низкоуровневого обнаружения. Это свойство указывает, что это прототип LLD-правила, прямой дочерний элемент указанного правила/прототипа правила. |
| key | string | (обязательно) Ключ родительского правила низкоуровневого обнаружения (или прототипа правила). |
| master_item | string | (обязательно для правил DEPENDENT) Корневой элемент для master-элемента зависимого правила. |
| lld_macro_paths | | Корневой элемент для путей макросов правила низкоуровневого обнаружения. |
| lld_macro | string | (обязательно) Имя макроса правила низкоуровневого обнаружения. |
| path | string | (обязательно) Селектор значения, которое будет назначено соответствующему макросу. |
| preprocessing | | Корневой элемент для предварительной обработки значения правила низкоуровневого обнаружения. |
| step | | Корневой элемент для элементов шага предварительной обработки значения правила низкоуровневого обнаружения, которые совпадают с шагами предварительной обработки значения элемента данных шаблона, но с меньшим количеством возможных значений. См. также: объект предварительной обработки LLD-правила. |
| overrides | | Корневой элемент для правил переопределения правила низкоуровневого обнаружения. |
| name | string | (обязательно) Уникальное имя переопределения. |
| step | string | (обязательно) Уникальный порядковый номер переопределения. |
| stop | string | Остановить обработку следующих переопределений, если совпадает. |
| filter | | Корневой элемент для элементов фильтра правила переопределения правила низкоуровневого обнаружения шаблона, которые совпадают с фильтрами правил низкоуровневого обнаружения шаблона. |
| operations | | Корневой элемент для операций переопределения правила низкоуровневого обнаружения шаблона. |

**Note:**
См. также: объект LLD-правила (см. соответствующее свойство с совпадающим именем).

Фильтры правила низкоуровневого обнаружения шаблона

```
zabbix_export:
  (...)
  templates:
    (...)
      discovery_rules:
        - uuid: acfdea9c46ef48c68e6636f43b8f96a2
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#IFNAME}'
                value: '{$NET.IF.IFNAME.MATCHES}'
              - macro: '{#IFNAME}'
                value: '{$NET.IF.IFNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
          (...)
        - uuid: 2bbdc79f082d4c618e01bec625e9c90a
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#DEVNAME}'
                value: '{$VFS.DEV.DEVNAME.MATCHES}'
              - macro: '{#DEVNAME}'
                value: '{$VFS.DEV.DEVNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
              - macro: '{#DEVTYPE}'
                value: disk
          (...)
```

| Element | Type | Description |
|---|---|---|
| evaltype | string | Переопределяет метод оценки условий фильтра.<br>Возможные значения:[1] AND_OR (0, по умолчанию), AND (1), OR (2), FORMULA (3). |
| formula | string | Пользовательская формула вычисления для условий фильтра. |
| conditions | | Корневой элемент для условий фильтра. |
| macro | string | (обязательно) Имя макроса низкоуровневого обнаружения, для которого выполняется проверка. |
| value | string | Значение для сравнения. |
| operator | string | Оператор условия.<br>Возможные значения:[1] MATCHES_REGEX (8, по умолчанию), NOT_MATCHES_REGEX (9). |
| formulaid | string | (обязательно) Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но будет сгенерирован заново при последующем запросе. |

**Note:**
См. также: объект фильтра правила LLD (обратитесь к соответствующему свойству с совпадающим именем).

Операции замещения правил низкоуровневого обнаружения шаблона

| Элемент | Тип | Описание |
|---|---|---|
| operationobject | строка | Объект, к которому применяется операция.<br>Возможные значения:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3). |

| Элемент | Тип | Описание |
|---|---|---|
| operator | строка | Оператор условия замещения.<br>Возможные значения:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6). |
| value | строка | Регулярное выражение или строка для оператора условия замещения. |
| status | строка | Состояние объекта при операции замещения. |
| discover | строка | Добавляется ли объект в результате обнаружения. |
| delay | строка | Интервал обновления, установленный для прототипа элемента данных при операции замещения. |
| history | строка | Период хранения истории, установленный для прототипа элемента данных при операции замещения. |
| trends | строка | Период хранения динамики изменений, установленный для прототипа элемента данных при операции замещения. |
| severity | строка | Важность прототипа триггера, установленная при операции замещения. |
| tags | | Корневой элемент для тегов, установленных для объекта при операции замещения. |
| tag | строка | (обязательный) Имя тега. |
| value | строка | Значение тега. |
| templates | | Корневой элемент для шаблонов, присоединённых к прототипу узла сети при операции замещения. |
| name | строка | (обязательный) Имя шаблона. |
| inventory_mode | строка | Режим инвентаризации прототипа узла сети, установленный при операции замещения. |

**Note:**
Смотрите также: объект операции замещения правила LLD (смотрите соответствующее свойство с совпадающим именем).

Веб-сценарии шаблона

| Элемент | Тип | Описание |
|---|---|---|
| uuid | строка | (обязательный) Уникальный идентификатор конкретного веб-сценария. |
| name | строка | (обязательный) Имя веб-сценария. |
| delay | строка | Частота выполнения веб-сценария (используя секунды, единицы времени с суффиксом или пользовательский макрос).<br>По умолчанию: 1m. |
| attempts | целое число | Количество попыток выполнения шагов веб-сценария.<br>Возможные значения:[1] 1-10 (по умолчанию: 1). |
| agent | строка | Агент клиента. Zabbix будет притворяться выбранным браузером.<br>Эта опция полезна, когда веб-сайт возвращает разное содержимое для разных браузеров.<br>По умолчанию: Zabbix. |
| http_proxy | строка | Укажите используемый HTTP прокси, используя следующий формат: `http://[ [: ]@] .example.com[: ]` |
| variables | | Корневой элемент для переменных уровня сценария, которые могут использоваться в шагах сценария. |
| name | строка | (обязательный) Имя переменной. |
| value | текст | (обязательный) Значение переменной. |
| headers | | Корневой элемент для HTTP заголовков, которые будут отправлены при выполнении запроса.<br>Заголовки нужно перечислять с использованием такого же синтаксиса, что и в HTTP протоколе. |
| name | строка | (обязательный) Имя заголовка. |
| value | текст | (обязательный) Значение заголовка. |
| status | строка | Состояние веб-сценария.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1). |
| authentication | строка | Метод аутентификации.<br>Возможные значения:[1] NONE (0, по умолчанию), BASIC (1), NTLM (2), Kerberos (3) или Digest (4). |

| Элемент | Тип | Описание |
|---|---|---|
| http_user | строка | Имя пользователя для простой (BASIC HTTP), NTLM, Kerberos или Digest аутентификации. |
| http_password | строка | Пароль для для простой (BASIC HTTP), NTLM, Kerberos или Digest аутентификации. |
| verify_peer | строка | Проверка SSL сертификата веб-сервера.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| verify_host | строка | Проверка, что поле *Общее Имя (Common Name)* или поле *Дополнительное Имя Субъекта (Subject Alternate Name)* сертификата веб-сервера совпадают.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| ssl_cert_file | строка | Имя используемого файла SSL сертификата для аутентификации клиента (должен быть в формате PEM). |
| ssl_key_file | строка | Имя используемого файла закрытого ключа (private key) SSL для аутентификации клиента (должен быть в формате PEM). |
| ssl_key_password | строка | Пароль к файлу закрытого ключа (private key) SSL. |
| steps | | (обязательный) Корневой элемент для шагов веб-сценария шаблона. |
| tags | | Корневой элемент тегов веб-сценария. |
| tag | строка | (обязательный) Имя тега. |
| value | строка | Значение тега. |

**Note:**
Смотрите также: Объект веб-сценария (смотрите соответствующее свойство с совпадающим именем).

Шаги веб-сценария шаблона

| Элемент | Тип | Описание |
|---|---|---|
| name | строка | (обязательный) Имя шага веб-сценария. |
| url | строка | (обязательный) URL для мониторинга. |
| query_fields | | Корневой элемент для параметров запроса (массив полей HTTP, которые будут добавлены к URL при выполнении запроса). |
| name | строка | (обязательный) Имя параметра запроса. |
| value | строка | Значение параметра запроса. |
| posts | | Корневой элемент для переменных HTTP POST (строка (необработанные post данные) или массив полей HTTP (данные полей формы)). |
| name | строка | (обязательный) Имя поля post. |
| value | строка | (обязательный) Значение поля post. |
| variables | | Корневой элемент переменных (макросов) уровня шага, которые следует применить после этого шага.<br>Если значение переменной имеет префикс «regex:», то её значение извлекается из данных, возвращаемых этим шагом в соответствии с шаблоном регулярного выражения, следующим за префиксом «regex:». |
| name | строка | (обязательный) Имя переменной. |
| value | текст | (обязательный) Значение переменной. |
| headers | | Корневой элемент для заголовков HTTP, которые будут отправлены при выполнении запроса. |
| name | строка | (обязательный) Имя заголовка. |
| value | текст | (обязательный) Значение заголовка. |
| follow_redirects | строка | Следовать перенаправлениям HTTP.<br>Возможные значения:[1] NO (0), YES (1, по умолчанию). |
| retrieve_mode | строка | Режим получения ответа HTTP.<br>Возможные значения:[1] BODY (0, по умолчанию), HEADERS (1), BOTH (2). |
| timeout | строка | Таймаут (используя секунды, единицы времени с суффиксом или пользовательский макрос) шага выполнения.<br>По умолчанию: 15s. |
| required | строка | Текст, который должен присутствовать в ответе (игнорируется, если пусто). |
| status_codes | строка | Список (через запятую) принимаемых кодов состояния HTTP (например: 200-201,210-299; игнорируется, если пусто). |

**Note:**
Смотрите также: Объект шага веб-сценария (смотрите соответствующее свойство с совпадающим именем).

Конфигурация макросов шаблона

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      (...)
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          (...)
          config:
            type: TEXT
            priority: '1'
            label: '                              Zabbix'
            description: '   -  ,                            .'
        - macro: '{$CPU.UTIL.CRIT}'
          (...)
          config:
            type: TEXT
            priority: '2'
            section_name: Thresholds
            label: '                CPU'
            description: '                        CPU   %.         0    100         .'
            regex: '^-?([0-9]+|([0-9]+)\.([0-9]+)))$'
        (...)
        - macro: '{$IFCONTROL}'
          (...)
          config:
            type: CHECKBOX
            priority: '19'
            label: '              '
            description: '           ,                                "Link down".'
            options:
              - checked: '1'
                unchecked: '0'
        (...)
```

| Элемент | Тип | Описание |
|---|---|---|
| type | string | (обязательно) Тип поля ввода макроса.<br>Возможные значения:[1] NOCONF (0), TEXT (1), LIST (2), CHECKBOX (3). |
| priority | string | Позиция макроса в списке макросов. |
| section_name | string | Метка сворачиваемого раздела, в котором сгруппирован макрос. |
| label | string | (обязательно для макросов TEXT, LIST, CHECKBOX) Метка макроса. |
| description | text | Справочный текст макроса. Поддерживает форматирование Markdown. |
| required | string | Помечает макрос как обязательный.<br>Возможные значения:[1] NO (0), YES (1).<br>Поддерживается для макросов TEXT и LIST. |
| regex | string | Регулярное выражение для проверки ввода пользователя в поле текстового ввода.<br>Поддерживается для макросов TEXT. |
| options | | Корневой элемент для элементов LIST или значений CHECKBOX. |
| value | string | (обязательно для макросов LIST) Значение элемента LIST. |
| text | string | (обязательно для макросов LIST) Текст элемента LIST. |
| checked | string | (обязательно для макросов CHECKBOX) Значение, представляющее состояние ”включено”.<br>Возможные значения:[1] 0 (false), 1 (true).<br>Поддерживается для макросов CHECKBOX. |
| unchecked | string | (обязательно для макросов CHECKBOX) Значение, представляющее состояние ”выключено”.<br>Возможные значения:[1] 0 (false), 1 (true).<br>Поддерживается для макросов CHECKBOX. |

**Note:**
См. также: Объект конфигурации макроса (см. соответствующее свойство с совпадающим именем).

Панели шаблона

```
zabbix_export:
  (...)
  templates:
    (...)
      dashboards:
        - uuid: c689ad3115fd46a4b927d1f70ee2e5a4
          name: Filesystems
          pages:
            - name: Overview
              widgets: (see table below)
        (...)
```

| Element | Type | Description |
|---|---|---|
| uuid | string | (required) Уникальный идентификатор этой панели. |
| name | string | (required) Имя панели шаблона. |
| display pe- riod | integer | Период отображения страниц панели. |
| auto_start | string | Автоматический запуск слайд-шоу.<br>Возможные значения:[1] NO (0), YES (1, default). |
| pages | | Корневой элемент для страниц панели шаблона. |
| name | string | Имя страницы. |
| display period | integer | Период отображения страницы. |
| sortorder | integer | Порядок сортировки страницы. |
| widgets | | Корневой элемент для виджетов панели шаблона. |

**Note:**
См. также: Объект панели шаблона (см. соответствующее свойство с совпадающим именем).

Виджеты панели шаблона

```
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})]:                    (              )'
                    - 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})]:                    ,   %, (                    )
                    - type: STRING
                      name: reference
                      value: AAAEC
        (...)
```

| Element | Type | Description |
|---|---|---|
| type | string | (required) Тип виджета. |
| name | string | Имя виджета. |
| x | integer | Горизонтальная позиция от левой стороны панели шаблона.<br>Возможные значения:[1] 0-71. |
| y | integer | Вертикальная позиция от верхней стороны панели шаблона.<br>Возможные значения:[1] 0-63. |
| width | integer | Ширина виджета.<br>Возможные значения:[1] 1-72. |
| height | integer | Высота виджета.<br>Возможные значения:[1] 1-64. |
| hide_header | string | Скрыть заголовок виджета.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| fields | | Корневой элемент для полей виджета панели шаблона. |
| type | string | (required) Тип поля виджета.<br>Возможные значения:[1] INTEGER (0), STRING (1), ITEM (4), ITEM_PROTOTYPE (5), GRAPH (6), GRAPH_PROTOTYPE (7), MAP (8), SERVICE (9), SLA (10), USER (11), ACTION (12), MEDIA_TYPE (13). |
| name | string | (required) Имя поля виджета. |
| value | mixed | (required) Значение поля виджета в зависимости от типа поля. |

**Note:**
См. также: Объект виджета панели шаблона (см. соответствующее свойство с совпадающим именем).

Карта значений шаблона

```
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
        (...)
```

| Элемент | Тип | Описание |
|---|---|---|
| uuid | string | (обязательно) Уникальный идентификатор этой карты значений. |
| name | string | (обязательно) Имя карты значений. |
| mapping | | Корневой элемент для сопоставлений. |
| type | string | Тип сопоставления по совпадению.<br>Возможные значения:[1] EQUAL (0, по умолчанию), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6). |
| value | string | Исходное значение. |
| newvalue | string | (обязательно) Значение, в которое сопоставляется исходное значение. |

**Note:**
См. также: Объект карты значений (см. соответствующее свойство с совпадающим именем).

**Примечания** [1] Целочисленные значения API в скобках, например, ENABLED (0), упоминаются только для справки. Для получения дополнительной информации смотрите связанную страницу объекта API в записи таблицы или в конце каждого раздела.

#### 4. Узлы сети

Обзор

Узлы сети экспортируются со многими связанными объектами и отношениями между объектами.

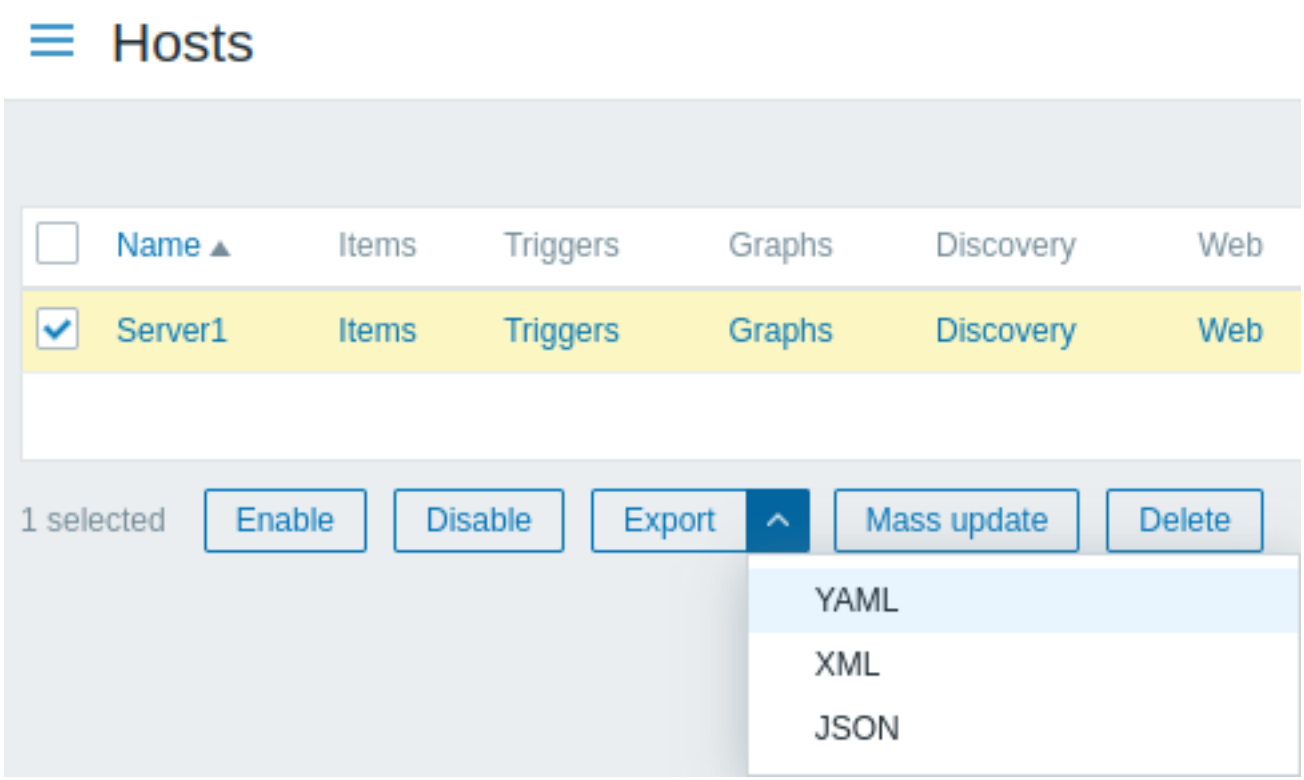
Экспорт узла сети содержит:

- Связанные группы узов сети
- Данные узла сети
- Связи с шаблонами
- Связи с группами узлов сети
- Интерфейсы узла сети
- Напрямую присоединённые элементы данных
- Напрямую присоединённые триггеры
- Напрямую присоединённые правила обнаружения со всеми прототипами
- Напрямую присоединённые веб-сценарии
- Макросы узла сети
- Данные инвентаризации узла сети
- Преобразования значений
- Присоединённые графики

Экспорт

Чтобы экспортировать узел сети, сделайте следующее:

1. Перейдите к: *Сбор данных → Узлы сети (Data collection → Hosts)*.
2. Отметьте флажками узлы сети для экспорта.
3. Нажмите *Экспорт (Export)* ниже списка.



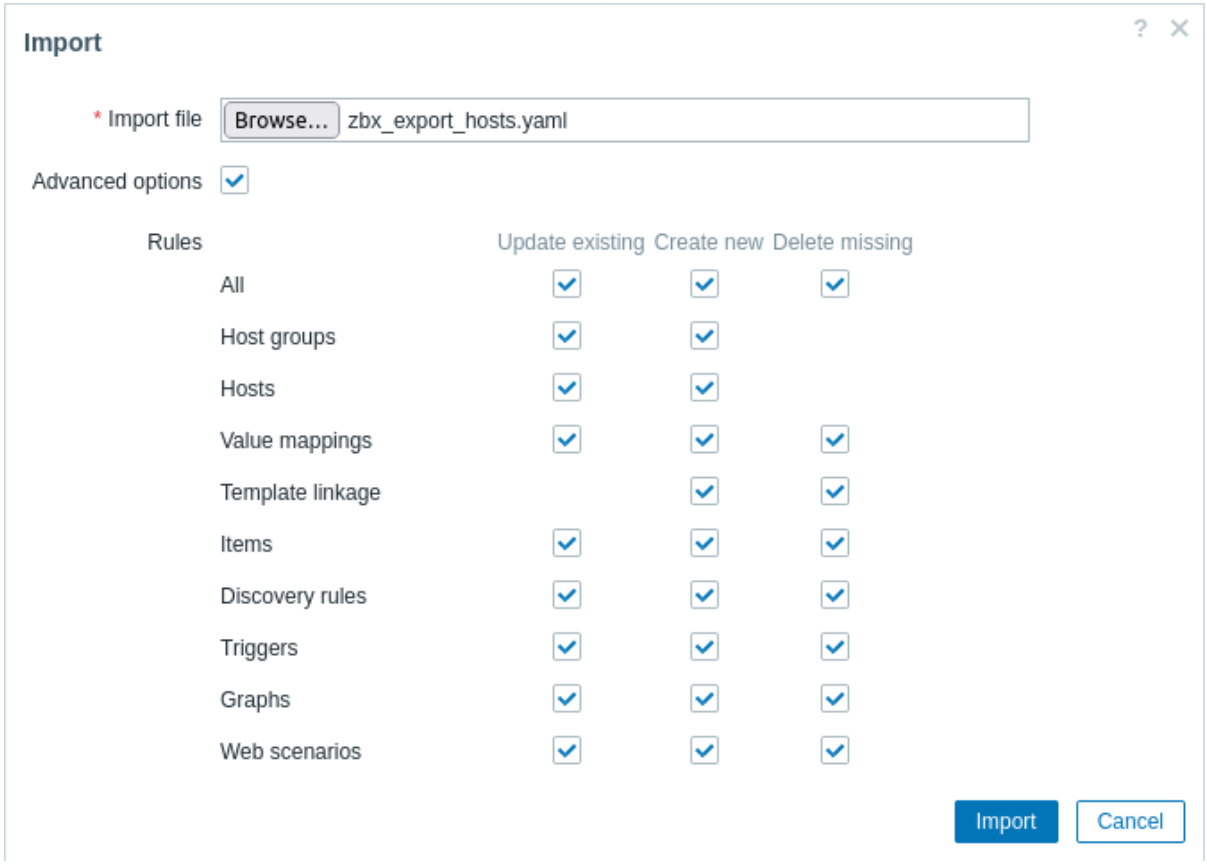

В зависимости от выбранного формата узлы сети экспортируются в локальный файл с именем по умолчанию:

- `zabbix_export_hosts.yaml` — при экспорте в YAML (опция по умолчанию для экспорта);
- `zabbix_export_hosts.xml` — при экспорте в XML;
- `zabbix_export_hosts.json` — при экспорте в JSON.

Импорт

Чтобы импортировать узлы сети, выполните следующие действия:

1. Перейдите в *Сбор данных* → *Узлы сети*.
2. Нажмите *Импорт* в правом верхнем углу.
3. Выберите файл импорта.
4. Нажмите *Импорт* в правом нижнем углу формы настройки.




Если установить флажок *Дополнительные параметры*, будет отображен подробный список всех элементов, доступных для импорта — отметьте или снимите отметку для каждого правила импорта по мере необходимости.

Если щелкнуть флажок в строке *Все*, будут отмечены/сняты отметки у всех элементов ниже.

Правила импорта:

| Правило | Описание |
|---|---|
| *Обновлять существующие* | Существующие элементы будут обновлены с использованием данных из файла импорта. В противном случае они не будут обновлены. |
| *Создавать новые* | Новые элементы будут созданы с использованием данных из файла импорта. В противном случае они не будут созданы. |
| *Удалять отсутствующие* | Существующие элементы, отсутствующие в файле импорта, будут удалены. В противном случае они не будут удалены.<br>Если для *Связи с шаблоном* отмечено *Удалять отсутствующие*, текущая связь с шаблоном, отсутствующая в файле импорта, будет разорвана. Наследуемые от отвязанных шаблонов сущности (элементы данных, триггеры, графики и т. д.) не будут удалены (если только для каждой сущности также не выбрана опция *Удалять отсутствующие*). |

В веб-интерфейсе будет отображено сообщение об успешном завершении импорта или об ошибке.

Формат экспорта

При экспорте узла сети Zabbix формирует структурированный формат YAML, JSON или XML. Экспорт включает элементы узла сети, такие как метаданные узла сети, элементы данных, макросы, триггеры и другие.

Каждый элемент выполняет определенную функцию и может содержать вложенные элементы.

В следующих разделах описан каждый элемент формата экспорта. В примерах используется узел сети с шаблоном MySQL by Zabbix agent 2. В целях демонстрации элементов шаблон отвязывается после создания узла сети. При отвязке все элементы остаются в конфигурации (в отличие от отвязки с очисткой, которая их удаляет).

Многоточие (`...`) обозначает элементы, опущенные для краткости. Примечание ( . ) используется, когда элемент более подробно объясняется в последующих разделах.

```
zabbix_export:
  version: '7.4'
  host_groups:
    - uuid: 748ad4d098d447d492bb935c907f652f
      name: Databases
  hosts: ( .        )
  graphs: ( .        )
```

**Метаданные узла сети**

| Element | Type | Description |
|---|---|---|
| version | string | (обязательно) Версия Zabbix. |
| host_groups | | (обязательно) Корневой элемент для групп узлов сети. |
| uuid | string | (обязательно) Уникальный идентификатор этой группы узлов сети. |
| name | string | (обязательно) Имя группы узлов сети. |
| hosts | | Корневой элемент для узлов сети. |
| graphs | | Корневой элемент для графиков узла сети. |

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      name: 'MySQL server'
      groups:
        - name: Databases
      interfaces: (see table below)
      items: (see table below)
      discovery_rules: (see table below)
      tags:
        - tag: location
          value: Riga
      macros:
        (...)
        - macro: '{$MYSQL.DSN}'
          value: 192.0.2.1
        (...)
        - macro: '{$MYSQL.PASSWORD}'
          type: SECRET_TEXT
          description: '            MySQL.'
        (...)
      valuemaps: (see table below)
  (...)
```

**Узлы сети**

| Element | Type | Description |
|---|---|---|
| host | string | (обязательно) Уникальное имя узла сети. |
| name | string | Отображаемое имя узла сети. |
| description | text | Описание узла сети. |
| monitored_by | string | Как осуществляется мониторинг узла сети.<br>Возможные значения:[1] SERVER (0, по умолчанию), PROXY (1) или PROXY_GROUP (2). |
| proxy | | Корневой элемент для прокси. |
| name | string | (обязательно) Имя прокси (если есть), который мониторит узел сети. |
| proxy_group | | Корневой элемент для группы прокси. |
| name | string | (обязательно) Имя группы прокси (если есть), которая используется для мониторинга узла сети. |
| status | string | Статус узла сети.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1). |

| Element | Type | Description |
|---|---|---|
| ipmi_authtype | string | Тип аутентификации сеанса IPMI.<br>Возможные значения:[1] DEFAULT (-1, по умолчанию), NONE (0), MD2 (1), MD5 (2), STRAIGHT (4), OEM (5), RMCP_PLUS (6). |
| ipmi_privilege | string | Уровень привилегий сеанса IPMI.<br>Возможные значения:[1] CALLBACK (1), USER (2, по умолчанию), OPERATOR (3), ADMIN (4), OEM (5). |
| ipmi_username | string | Имя пользователя для проверок IPMI. |
| ipmi_password | string | Пароль для проверок IPMI. |
| templates | | Корневой элемент для связанных шаблонов. |
| name | string | (обязательно) Имя шаблона. |
| groups | | Корневой элемент для групп узлов сети, к которым принадлежит узел сети. |
| name | string | (обязательно) Имя группы узлов сети. |
| interfaces | | Корневой элемент для интерфейсов узла сети. |
| items | | Корневой элемент для элементов данных узла сети. |
| discovery_rules | | Корневой элемент для правил низкоуровневого обнаружения узла сети. |
| http_tests | | Корневой элемент для веб-сценариев узла сети. |
| tags | | Корневой элемент для тегов узла сети. |
| tag | string | (обязательно) Имя тега. |
| value | string | Значение тега. |
| macros | | Корневой элемент для макросов узла сети. |
| macro | string | (обязательно) Имя пользовательского макроса. |
| type | string | Тип пользовательского макроса.<br>Возможные значения:[1] TEXT (0, по умолчанию), SECRET_TEXT (1), VAULT (2). |
| value | string | Значение пользовательского макроса. |
| description | string | Описание пользовательского макроса. |
| inventory | | Корневой элемент для инвентарных данных узла сети. |
| <inventory_property> | string | Свойство инвентарных данных. У всех свойств есть соответствующий элемент (`type`, `name`, `os` и т. д.; например, см. Формат экспорта). |
| inventory_mode | string | Режим инвентаризации.<br>Возможные значения:[1] DISABLED (-1), MANUAL (0, по умолчанию), AUTOMATIC (1). |
| valuemaps | | Корневой элемент для карт значений узла сети. |

**Note:**
См. также: Объект Host (обратитесь к соответствующему свойству с совпадающим именем).

Интерфейсы узла сети

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      interfaces:
        - ip: 192.0.2.1
          interface_ref: if1
      (...)
```

| Element | Type | Description |
|---|---|---|
| default | string | Является ли это основным интерфейсом узла сети. Обратите внимание, что на узле сети может быть только один основной интерфейс одного типа.<br>Возможные значения:[1] NO (0), YES (1, default). |
| type | string | Тип интерфейса.<br>Возможные значения:[1] ZABBIX (1, default), SNMP (2), IPMI (3), JMX (4). |
| useip | string | Использовать ли IP в качестве интерфейса для подключения к узлу сети (в противном случае будет использоваться DNS).<br>Возможные значения:[1] NO (0), YES (1, default). |
| ip | string | (required for IP connections) IP address (IPv4 or IPv6). |
| dns | string | (required for DNS connections) DNS name. |
| port | string | Номер порта. |
| details | | Корневой элемент для сведений об интерфейсе. |

| Element | Type | Description |
|---|---|---|
| version | string | Использовать эту версию SNMP.<br>Возможные значения:[1] SNMPV1 (1), SNMP_V2C (2, default), SNMP_V3 (3). |
| community | string | (required for SNMPv1 and SNMPv2 items) SNMP community. |
| max_repetitions | string | Максимальное значение повторений для нативных массовых SNMP-запросов (GetBulkRequest-PDUs).<br>Поддерживается для элементов данных SNMPv2 и SNMPv3 (`discovery[]` и `walk[]`).<br>По умолчанию: 10. |
| contextname | string | Имя контекста SNMPv3.<br>Поддерживается для элементов данных SNMPv3. |
| securityname | string | Имя безопасности SNMPv3.<br>Поддерживается для элементов данных SNMPv3. |
| securitylevel | string | Уровень безопасности SNMPv3.<br>Поддерживается для элементов данных SNMPv3.<br>Возможные значения:[1] NOAUTHNOPRIV (0, default), AUTHNOPRIV (1), AUTHPRIV (2). |
| authprotocol | string | Протокол аутентификации SNMPv3.<br>Поддерживается для элементов данных SNMPv3.<br>Возможные значения:[1] MD5 (0, default), SHA1 (1), SHA224 (2), SHA256 (3), SHA384 (4), SHA512 (5). |
| authpassphrase | string | Фраза-пароль аутентификации SNMPv3.<br>Поддерживается для элементов данных SNMPv3. |
| privprotocol | string | Протокол конфиденциальности SNMPv3.<br>Поддерживается для элементов данных SNMPv3.<br>Возможные значения:[1] DES (0, default), AES128 (1), AES192 (2), AES256 (3), AES192C (4), AES256C (5). |
| privpassphrase | string | Фраза-пароль конфиденциальности SNMPv3.<br>Поддерживается для элементов данных SNMPv3. |
| bulk | string | Использовать массовые запросы для SNMP.<br>Возможные значения:[1] NO (0), YES (1, default). |
| interface_ref | string | Имя ссылки на интерфейс, которое будет использоваться в элементах данных (формат: `if<N>`). |

**Note:**
См. также: Объект интерфейса узла сети (ссылайтесь на соответствующее свойство с совпадающим именем).

Элементы данных узла сети

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: '                 binlog'
          type: DEPENDENT
          key: mysql.binlog_cache_disk_use
          value_type: FLOAT
          description: '            ,                          ,                                    ,
          preprocessing: ( .        )
          master_item:
            key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          tags:
            - tag: component
              value: cache
        (...)
        - name: '               '
          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: '                                          .'
        tags:
          - tag: component
            value: memory
        triggers: ( .        )
      (...)
      - name:
        type: DEPENDENT
        key: mysql.uptime
        units: uptime
        description: '          ,                    .'
        preprocessing: ( .         )
        master_item:
          key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
        tags:
          - tag: component
            value: application
        triggers: ( .        )
      (...)
```

| Element | Type | Description |
|---|---|---|
| name | string | (обязательно) Имя элемента данных. |
| type | string | Тип элемента данных.<br>Возможные значения:[1] ZABBIX_PASSIVE (0, по умолчанию), 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). |
| snmp_oid | string | (обязательно для элементов данных SNMP_AGENT) Идентификатор объекта SNMP. |
| key | string | (обязательно) Ключ элемента данных. |
| delay | string | Интервал обновления элемента данных.<br>По умолчанию: 1m. Значение всегда будет 0 для элементов данных TRAP. |
| history | string | Период времени (с использованием суффикса времени, пользовательского макроса или LLD-макроса), в течение которого должны храниться данные истории.<br>По умолчанию: 31d. |
| trends | string | Период времени (с использованием суффикса времени, пользовательского макроса или LLD-макроса), в течение которого должны храниться данные трендов.<br>По умолчанию: 365d. |
| status | string | Статус элемента данных.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1). |
| value_type | string | Тип получаемого значения.<br>Возможные значения:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, по умолчанию), TEXT (4), BINARY (5). |
| allowed_hosts | string | Список IP-адресов узлов сети, разделенных запятыми, которым разрешено отправлять данные для элемента данных.<br>Поддерживается для элементов данных TRAP и HTTP_AGENT. |
| units | string | Единицы получаемого значения (bps, B, и т. д.). |
| params | text | Дополнительные параметры в зависимости от типа элемента данных (выполняемый скрипт для элементов данных SSH и TELNET; SQL-запрос для элементов данных ODBC; формула для элементов данных CALCULATED; скрипт для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER). |
| ipmi_sensor | string | Датчик IPMI.<br>Поддерживается для элементов данных IPMI. |

| Element | Type | Description |
|---|---|---|
| authtype | string | Тип аутентификации.<br>Поддерживается для элементов данных SSH и HTTP_AGENT.<br>Возможные значения для элементов данных SSH:[1] PASSWORD (0, по умолчанию), PUBLIC_KEY (1).<br>Возможные значения для элементов данных HTTP_AGENT:[1] NONE (0, по умолчанию), BASIC (1), NTLM (2), Kerberos (3) или Digest (4). |
| username | string | (обязательно для элементов данных SSH и TELNET) Имя пользователя для аутентификации.<br>Поддерживается для элементов данных SIMPLE, ODBC, JMX и HTTP_AGENT.<br>При использовании для элементов данных JMX также следует указать `password` (см. ниже) либо оставить оба элемента пустыми. |
| password | string | (обязательно для элементов данных SSH и TELNET) Пароль для аутентификации.<br>Поддерживается для элементов данных SIMPLE, ODBC, JMX и HTTP_AGENT.<br>При использовании для элементов данных JMX также следует указать `username` (см. выше) либо оставить оба элемента пустыми. |
| publickey | string | (обязательно для элементов данных SSH) Имя файла открытого ключа. |
| privatekey | string | (обязательно для элементов данных SSH) Имя файла закрытого ключа. |
| description | text | Описание элемента данных. |
| inventory_link | string | Поле инвентаря узла сети, которое заполняется элементом данных.<br>Возможные значения:[1] NONE (0), ALIAS (4) и т. д. (см. Инвентарь узла сети для поддерживаемых полей). |
| valuemap | | Корневой элемент для карт значений элемента данных. |
| name | string | (обязательно) Имя карты значений, используемой для элемента данных. |
| logtimefmt | string | Формат времени в записях журнала.<br>Поддерживается для элементов данных с типом значения LOG. |
| preprocessing | | Корневой элемент для предварительной обработки значения элемента данных. |
| step | | Корневой элемент для шагов предварительной обработки значения элемента данных узла сети. |
| interface_ref | string | Ссылка на интерфейс узла сети (формат: `if<N>`). |
| jmx_endpoint | string | Конечная точка JMX.<br>Поддерживается для элементов данных JMX. |
| master_item | | (обязательно для элементов данных DEPENDENT) Корневой элемент для master item зависимого элемента данных. |
| key | string | (обязательно) Ключ master item зависимого элемента данных. |
| timeout | string | Тайм-аут запроса опроса данных элемента данных.<br>Поддерживается для списка типов элементов данных Timeouts. |
| url | string | (обязательно для элементов данных HTTP_AGENT) Строка URL. |
| query_fields | | Корневой элемент для параметров запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |
| name | string | (обязательно для элементов данных HTTP_AGENT) Имя параметра запроса. |
| value | string | Значение параметра запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |
| parameters | | Корневой элемент для пользовательских параметров.<br>Поддерживается для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER. |
| name | string | (обязательно для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER) Имя пользовательского параметра. |
| value | string | Значение пользовательского параметра.<br>Поддерживается для элементов данных ITEM_TYPE_SCRIPT и ITEM_TYPE_BROWSER. |
| posts | string | Данные тела HTTP(S)-запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми.<br>Поддерживается для элементов данных HTTP_AGENT. |
| follow_redirects | string | Следовать перенаправлениям ответа при опросе данных.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| post_type | string | Тип тела данных POST.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] RAW (0, по умолчанию), JSON (2), XML (3). |
| http_proxy | string | Строка подключения к HTTP(S)-прокси.<br>Поддерживается для элементов данных HTTP_AGENT. |
| headers | | Корневой элемент для заголовков HTTP(S)-запроса.<br>Поддерживается для элементов данных HTTP_AGENT. |

| Element | Type | Description |
|---|---|---|
| name | string | (обязательно для элементов данных HTTP_AGENT) Имя заголовка. |
| value | string | (обязательно для элементов данных HTTP_AGENT) Значение заголовка. |
| retrieve_mode | string | Какая часть ответа должна быть сохранена.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] BODY (0, по умолчанию), HEADERS (1), BOTH (2). |
| request_method | string | Тип метода запроса.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] GET (0, по умолчанию), POST (1), PUT (2), HEAD (3). |
| output_format | string | Способ обработки ответа.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] RAW (0, по умолчанию), JSON (1). |
| allow_traps | string | Разрешить заполнение значения аналогично элементу данных trapper.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br>Поддерживается для элементов данных HTTP_AGENT. |
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br>Поддерживается для элементов данных HTTP_AGENT. |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br>Поддерживается для элементов данных HTTP_AGENT. |
| verify_peer | string | Проверять ли подлинность сертификата узла сети.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| verify_host | string | Проверять ли, что имя узла сети для подключения совпадает с именем в сертификате узла сети.<br>Поддерживается для элементов данных HTTP_AGENT.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| tags | | Корневой элемент для тегов элемента данных. |
| tag | string | (обязательно) Имя тега. |
| value | string | Значение тега. |
| triggers | | Корневой элемент для триггеров элемента данных узла сети. |

**Note:**
См. также: Объект элемента данных (см. соответствующее свойство с совпадающим именем).

Шаги предварительной обработки значения элемента данных

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: 'Binlog cache disk use'
          (...)
          preprocessing:
            - type: JSONPATH
              parameters:
                - $.Binlog_cache_disk_use
            - type: DISCARD_UNCHANGED_HEARTBEAT
              parameters:
                - 6h
          (...)
```

| Element | Type | Description |
|---|---|---|
| type | string | (обязательно) Тип шага предварительной обработки значения элемента данных. Возможные значения:[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 | | (обязательно) Корневой элемент для параметров шага предварительной обработки значения элемента данных. |
| parameter | string | Отдельный параметр шага предварительной обработки значения элемента данных. |
| error_handler | string | Тип действия, используемый в случае сбоя шага предварительной обработки. Возможные значения:[1] ORIGINAL_ERROR (0, default), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3). |
| error_handler_params | string | Параметры обработчика ошибок. |

**Note:**
См. также: Объект предварительной обработки элемента данных (см. соответствующее свойство с совпадающим именем).

Триггеры элемента данных узла сети

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: Uptime
          (...)
          triggers:
            - expression: 'nodata(/MySQL server/mysql.uptime,30m)=1'
              name: 'MySQL:                          '
              event_name: 'MySQL:                          (            30m)'
              priority: INFO
              description: 'Zabbix                                              30    .'
              dependencies:
                - name: 'MySQL:           '
                  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:                '
              event_name: 'MySQL:                (uptime < 10m)'
              priority: INFO
              description: '        MySQL      10    .'
              tags:
                - tag: scope
                  value: notice
        (...)
```

| Element | Type | Description |
|---|---|---|
| uuid | string | (required) Уникальный идентификатор для этого триггера. |
| expression | string | (required) Выражение триггера. |

| Element | Type | Description |
|---|---|---|
| recovery_mode | string | Основа для генерации событий OK.<br>Возможные значения:[1] EXPRESSION (0, default), RECOVERY_EXPRESSION (1), NONE (2). |
| recovery_expression | string | Выражение восстановления триггера. |
| correlation_mode | string | Режим корреляции (без корреляции событий или корреляция событий по тегу).<br>Возможные значения:[1] DISABLED (0, default), TAG_VALUE (1). |
| correlation_tag | string | Имя тега, используемое для корреляции событий. |
| name | string | (required) Имя триггера. |
| event_name | string | Имя события. |
| opdata | string | Оперативные данные. |
| url_name | string | Метка для URL, связанного с триггером. |
| url | string | URL, связанный с триггером. |
| status | string | Статус триггера.<br>Возможные значения:[1] ENABLED (0, default), DISABLED (1). |
| priority | string | Серьезность триггера.<br>Возможные значения:[1] NOT_CLASSIFIED (0, default), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5). |
| description | text | Описание триггера. |
| type | string | Тип генерации событий (одно событие проблемы или несколько событий проблемы).<br>Возможные значения:[1] SINGLE (0, default), MULTIPLE (1). |
| manual_close | string | Ручное закрытие событий проблемы.<br>Возможные значения:[1] NO (0, default), YES (1). |
| dependencies | | Корневой элемент для зависимостей. |
| name | string | (required) Имя зависимого триггера. |
| expression | string | (required) Выражение зависимого триггера. |
| recovery_expression | string | Выражение восстановления зависимого триггера. |
| tags | | Корневой элемент для тегов триггера. |
| tag | string | (required) Имя тега. |
| value | string | Значение тега. |

**Note:**
См. также: Trigger object (обратитесь к соответствующему свойству с совпадающим именем).

Правила низкоуровневого обнаружения узла сети

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: '              '
          key: 'mysql.db.discovery["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          delay: 1h
          filter: ( .        )
          description: '                    .'
          interface_ref: if1
          item_prototypes:
            - name: '            {#DATABASE}'
              key: 'mysql.db.size["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}","{#DATABASE}"]'
              delay: 5m
              units: B
              description: '           .'
              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:**
Большинство элементов правил низкоуровневого обнаружения узла сети такие же, как и для элементов данных узла сети. В таблице ниже описаны только те элементы, которые отличаются от элементов данных узла сети.

| Element | Type | Description |
|---|---|---|
| type | string | Тип элемента данных.<br>Возможные значения:[1] ZABBIX_PASSIVE (0, по умолчанию), 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 | (обязательно) Ключ правила низкоуровневого обнаружения. |
| filter | | Корневой элемент для фильтров правил низкоуровневого обнаружения узла сети. |
| lifetime | string | Период времени (в секундах, с суффиксом времени или пользовательским макросом), по истечении которого больше не обнаруженные ресурсы будут удалены.<br>По умолчанию: 7d. |
| lifetime_type | string | Сценарий удаления потерянных ресурсов LLD.<br>Возможные значения: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER. |
| enabled_lifetime | string | Период времени (в секундах, с суффиксом времени или пользовательским макросом), по истечении которого больше не обнаруженные ресурсы будут отключены. |
| enabled_lifetime_type | string | Сценарий отключения потерянных ресурсов LLD.<br>Возможные значения: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER. |
| item_prototypes | | Корневой элемент для прототипов элементов данных узла сети, которые такие же, как и для элементов данных узла сети. |
| trigger_prototypes | | Корневой элемент для прототипов триггеров узла сети, которые такие же, как и для триггеров элементов данных узла сети. |
| graph_prototypes | | Корневой элемент для прототипов графиков узла сети, которые такие же, как и для графиков узла сети. |
| host_prototypes | | Корневой элемент для прототипов узлов сети, которые такие же, как и для узлов сети. |
| parent_discovery_rule | | Корневой элемент для родительского правила низкоуровневого обнаружения (или прототипа правила) прототипа правила низкоуровневого обнаружения. Это свойство указывает, что это прототип правила LLD, прямой дочерний элемент указанного правила/прототипа правила. |
| key | string | (обязательно) Ключ родительского правила низкоуровневого обнаружения (или прототипа правила). |
| master_item | string | (обязательно для правил DEPENDENT) Корневой элемент для master-элемента зависимого правила. |
| lld_macro_paths | | Корневой элемент для путей макросов правила низкоуровневого обнаружения. |
| lld_macro | string | (обязательно) Имя макроса правила низкоуровневого обнаружения. |
| path | string | (обязательно) Селектор значения, которое будет назначено соответствующему макросу. |
| preprocessing | | Корневой элемент для предварительной обработки значения правила низкоуровневого обнаружения. |

| Element | Type | Description |
|---|---|---|
| step | | Корневой элемент для элементов шага предварительной обработки значения правила низкоуровневого обнаружения, которые такие же, как и для шагов предварительной обработки значения элемента данных узла сети, за исключением меньшего числа возможных значений. См. также: объект предварительной обработки правила LLD. |
| overrides | | Корневой элемент для правил переопределения правила низкоуровневого обнаружения. |
| name | string | (обязательно) Уникальное имя переопределения. |
| step | string | (обязательно) Уникальный порядковый номер переопределения. |
| stop | string | Остановить обработку следующих переопределений, если совпадает. |
| filter | | Корневой элемент для элементов фильтра правила переопределения правила низкоуровневого обнаружения, которые такие же, как и для фильтров правил низкоуровневого обнаружения узла сети. |
| operations | | Корневой элемент для операций переопределения правил низкоуровневого обнаружения узла сети. |

**Note:**
См. также: объект правила LLD (ссылайтесь на соответствующее свойство с совпадающим именем).

Фильтры правила низкоуровневого обнаружения узла сети

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: 'Database discovery'
          (...)
          filter:
            evaltype: AND
            conditions:
              - macro: '{#DATABASE}'
                value: '{$MYSQL.DBNAME.MATCHES}'
              - macro: '{#DATABASE}'
                value: '{$MYSQL.DBNAME.NOT_MATCHES}'
                operator: NOT_MATCHES_REGEX
          (...)
```

| Element | Type | Description |
|---|---|---|
| evaltype | string | Переопределяет метод оценки условий фильтра.<br>Возможные значения:[1] AND_OR (0, по умолчанию), AND (1), OR (2), FORMULA (3). |
| formula | string | Пользовательская формула вычисления для условий фильтра. |
| conditions | | Корневой элемент для условий фильтра. |
| macro | string | (required) Имя макроса низкоуровневого обнаружения, для которого выполняется проверка. |
| value | string | Значение, с которым выполняется сравнение. |
| operator | string | Оператор условия.<br>Возможные значения:[1] MATCHES_REGEX (8, по умолчанию), NOT_MATCHES_REGEX (9). |
| formulaid | string | (required) Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но будет сгенерирован заново при последующем запросе. |

**Note:**
См. также: объект фильтра правила LLD (см. соответствующее свойство с совпадающим именем).

Операции переопределения правила низкоуровневого обнаружения узла сети

| Element | Type | Description |
|---|---|---|
| operationobject | string | Объект, к которому применяется операция.<br>Возможные значения:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3), DISCOVERY_PROTOTYPE (4). |
| operator | string | Оператор условия переопределения.<br>Возможные значения:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6). |
| value | string | Регулярное выражение или строка для оператора условия переопределения. |
| status | string | Состояние объекта после операции переопределения. |
| discover | string | Указывает, добавляется ли объект в результате обнаружения. |
| delay | string | Интервал обновления, заданный для прототипа элемента данных после операции переопределения. |
| history | string | Период хранения истории, заданный для прототипа элемента данных после операции переопределения. |
| trends | string | Период хранения трендов, заданный для прототипа элемента данных после операции переопределения. |
| severity | string | Уровень серьезности прототипа триггера, заданный после операции переопределения. |
| tags | | Корневой элемент для тегов, заданных для объекта после операции переопределения. |
| tag | string | (обязательно) Имя тега. |
| value | string | Значение тега. |
| templates | | Корневой элемент для шаблонов, связанных с прототипом узла сети после операции переопределения. |
| name | string | (обязательно) Имя шаблона. |
| inventory_mode | string | Режим инвентаризации прототипа узла сети, заданный после операции переопределения. |

**Note:**
См. также: Объект операции переопределения правила LLD (обратитесь к соответствующему свойству с совпадающим именем).

Веб-сценарии узла сети

| Элемент | Тип | Описание |
|---|---|---|
| uuid | строка | (обязательный) Уникальный идентификатор для этого веб-сценария. |
| name | строка | (обязательный) Имя веб-сценария. |
| delay | строка | Частота (используя секунды, суффиксы времени или пользовательские макросы) выполнения веб-сценария.<br>По умолчанию: 1m. |
| attempts | целое число | Количество попыток выполнения шагов веб-сценария.<br>Возможные значения: 1-10 (по умолчанию: 1). |
| agent | строка | Агент клиента. Zabbix будет притворяться выбранным браузером.<br>Эта опция полезна, когда веб-сайт возвращает разное содержимое для разных браузеров.<br>По умолчанию: Zabbix. |
| http_proxy | строка | Прокси, который будет использоваться этим веб-сценарием, в формате: `http://[ _ [: ]@] . .com[: ]` |
| variables | | Корневой элемент переменных на уровне сценария, которые могут быть использованы в шагах сценария. |
| name | строка | (обязательный) Имя переменной. |
| value | текст | (обязательный) Значение переменной. |
| headers | | Корневой элемент заголовков HTTP, которые будут отосланы при выполнении запроса.<br>Заголовки должны быть перечислены, используя такой же синтаксис, как если бы они появлялись в протоколе HTTP. |
| name | строка | (обязательный) Имя заголовка. |
| value | текст | (обязательный) Значение заголовка. |
| status | строка | Состояние веб-сценария.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1). |

| Элемент | Тип | Описание |
|---|---|---|
| authentication | строка | Метод аутентификации.<br>Возможные значения:[1] NONE (0, по умолчанию), BASIC (1), NTLM (2), Kerberos (3) или Digest (4). |
| http_user | строка | Имя пользователя для BASIC (HTTP), NTLM, Kerberos или Digest аутентификации. |
| http_password | строка | Пароль, используемый для BASIC (HTTP), NTLM, Kerberos или Digest аутентификации. |
| verify_peer | строка | Проверка SSL сертификата веб-сервера.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| verify_host | строка | Проверка того, что поле *Общее Имя (Common Name)* или поле *Дополнительное Имя Субъекта (Subject Alternate Name)* сертификата веб-сервера совпадают.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| ssl_cert_file | строка | Имя файла SSL сертификата, используемого для аутентификации клиента (должен быть в формате PEM). |
| ssl_key_file | строка | Имя файла закрытого ключа SSL, используемого для аутентификации клиента (должен быть в формате PEM). |
| ssl_key_password | строка | Пароль к файлу закрытого ключа SSL. |
| steps | | (обязательный) Корневой элемент для шагов веб-сценария. |
| tags | | Корневой элемент для тегов веб-сценария. |
| tag | строка | (обязательный) Имя тега. |
| value | строка | Значение тега. |

**Note:**
Смотрите также: объект веб-сценария (смотрите соответствующее свойство с совпадающим именем).

Шаги веб-сценария узла сети

| Элемент | Тип | Описание |
|---|---|---|
| name | строка | (обязательный) Имя шага веб-сценария. |
| url | строка | (обязательный) Наблюдаемый URL. |
| query_fields | | Корневой элемент для параметров запроса (массив полей HTTP, которые будут добавлены к URL при выполнении запроса). |
| name | строка | (обязательный) Имя параметра запроса. |
| value | строка | Значение параметра запроса. |
| posts | | Корневой элемент для переменных HTTP POST как строка (сырые post данные) или как массив полей HTTP (данные полей формы). |
| name | строка | (обязательный) Имя поля post. |
| value | строка | (обязательный) Значение поля post. |
| variables | | Корневой элемент для переменных (макросов) уровня шага, которые могут применяться после этого шага.<br>Если значение переменной имеет префикс «regex:», тогда её значение извлекается из данных, которые получены на этом шаге, в соответствии с шаблоном регулярного выражения, следующим за префиксом «regex:». |
| name | строка | (обязательный) Имя переменной. |
| value | текст | (обязательный) Значение переменной. |
| headers | | Корневой элемент HTTP заголовков, которые будут отправлены при выполнении запроса. |
| name | строка | (обязательный) Имя заголовка. |
| value | текст | (обязательный) Значение заголовка. |
| follow_redirects | строка | Следование HTTP перенаправлениям.<br>Возможные значения:[1] NO (0), YES (1, по умолчанию). |
| retrieve_mode | строка | Режим получения HTTP ответа.<br>Возможные значения:[1] BODY (0, по умолчанию), HEADERS (1), BOTH (2). |
| timeout | строка | Время ожидания (используя секунды, суффиксы времени или пользовательские макросы) выполнения шага.<br>По умолчанию: 15s. |
| required | строка | Текст, который должен присутствовать в ответе (игнорируется, если пусто). |
| status_codes | строка | Список допустимых кодов состояний, разделённых запятыми (например: 200-201,210-299; игнорируется, если пустой). |

**Note:**
Смотрите также: объект шага веб-сценария (смотрите соответствующее свойство с совпадающим именем).

Графики узла сети

```
zabbix_export:
  (...)
  graphs:
    - name: 'MySQL: Bandwidth'
      graph_items: (see table below)
    (...)
```

| Element | Type | Description |
|---|---|---|
| uuid | string | Уникальный идентификатор графика. |
| name | string | (required) Имя графика. |
| width | integer | Ширина графика в пикселях.<br>Используется для предварительного просмотра и для круговых/развернутых графиков.<br>Возможные значения:[1] 20-65535 (default: 900). |
| height | integer | Высота графика в пикселях.<br>Используется для предварительного просмотра и для круговых/развернутых графиков.<br>Возможные значения:[1] 20-65535 (default: 900). |
| yaxismin | double | Минимальное значение оси Y.<br>Поддерживается для FIXED минимального значения оси Y.<br>По умолчанию: 0. |
| yaxismax | double | Максимальное значение оси Y.<br>Поддерживается для FIXED минимального значения оси X.<br>По умолчанию: 0. |
| show_work_period | string | Выделять нерабочие часы.<br>Поддерживается для графиков NORMAL и STACKED.<br>Возможные значения:[1] NO (0), YES (1, default). |
| show_triggers | string | Отображать простые значения триггеров в виде линии.<br>Поддерживается для графиков NORMAL и STACKED.<br>Возможные значения:[1] NO (0), YES (1, default). |
| type | string | Тип графика.<br>Возможные значения:[1] NORMAL (0, default), STACKED (1), PIE (2), EXPLODED (3). |
| show_legend | string | Отображать легенду графика.<br>Возможные значения:[1] NO (0), YES (1, default). |
| show_3d | string | Включить 3D-стиль.<br>Поддерживается для графиков NORMAL и STACKED.<br>Возможные значения:[1] NO (0, default), YES (1). |
| percent_left | double | Показывать линию процентиля для левой оси.<br>Поддерживается для графиков NORMAL.<br>По умолчанию: 0. |
| percent_right | double | Показывать линию процентиля для правой оси.<br>Поддерживается для графиков NORMAL.<br>По умолчанию: 0. |
| ymin_type_1 | string | Минимальное значение оси Y.<br>Поддерживается для графиков NORMAL и STACKED.<br>Возможные значения:[1] CALCULATED (0, default), FIXED (1), ITEM (2). |
| ymin_item_1 | | (required if `ymin_type_1` is set to `ITEM`) Корневой элемент для сведений об отдельном элементе данных. |
| host | string | (required) Узел сети элемента данных. |
| key | string | (required) Ключ элемента данных. |
| ymax_type_1 | string | Максимальное значение оси Y.<br>Поддерживается для графиков NORMAL и STACKED.<br>Возможные значения:[1] CALCULATED (0, default), FIXED (1), ITEM (2). |
| ymax_item_1 | | (required if `ymax_type_1` is set to `ITEM`) Корневой элемент для сведений об отдельном элементе данных. |
| host | string | (required) Узел сети элемента данных. |
| key | string | (required) Ключ элемента данных. |

| Element | Type | Description |
|---|---|---|
| graph_items | | (required) Корневой элемент для элементов графика узла сети. |

**Note:**
См. также: Объект Graph (обратитесь к соответствующему свойству с совпадающим именем).

Элементы графика узла сети

```
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
    (...)
```

| Элемент | Тип | Описание |
|---|---|---|
| sortorder | integer | Порядок отображения. Меньшее значение отображается первым. Может использоваться для отображения линий или областей позади (или перед) другими. |
| drawtype | string | Стиль отображения элемента графика.<br>Поддерживается для графиков NORMAL.<br>Возможные значения:[1] SINGLE_LINE (0, по умолчанию), FILLED_REGION (1), BOLD_LINE (2), DOTTED_LINE (3), DASHED_LINE (4), GRADIENT_LINE (5). |
| color | string | Цвет элемента (6 символов, hex). |
| yaxisside | string | Сторона графика, на которой будет отображаться шкала Y элемента графика.<br>Поддерживается для графиков NORMAL и STACKED. |
| calc_fnc | string | Данные для отображения, если для элемента существует более одного значения.<br>Возможные значения:[1] MIN (1), AVG (2, по умолчанию), MAX (4), ALL (7; минимум, среднее и максимум; поддерживается для простых графиков), LAST (9, поддерживается для круговых/развернутых графиков). |
| type | string | Тип элемента графика.<br>Возможные значения:[1] SIMPLE (0, по умолчанию), GRAPH_SUM (2; значение элемента представляет весь круговой график; поддерживается для круговых/развернутых графиков). |
| item | | (обязательно) Отдельный элемент. |
| host | string | (обязательно) Узел сети элемента. |
| key | string | (обязательно) Ключ элемента. |

**Note:**
См. также: Объект элемента графика (см. соответствующее свойство с совпадающим именем).

Карты значений узла сети

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      valuemaps:
```

```
      - name: Example value map
        mappings:
          - value: '1'
            newvalue: Example value
(...)
```

| Element | Type | Description |
|---|---|---|
| uuid | string | (required) Уникальный идентификатор этой карты значений. |
| name | string | (required) Имя карты значений. |
| mapping | | Корневой элемент для сопоставлений. |
| type | string | Тип сопоставления по совпадению.<br>Возможные значения:[1] EQUAL (0, по умолчанию), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6). |
| value | string | Исходное значение. |
| newvalue | string | (required) Значение, на которое сопоставляется исходное значение. |

**Note:**
См. также: Объект карты значений (см. соответствующее свойство с совпадающим именем).

**Примечания** [1] Целочисленные значения API в скобках, например, `ENABLED (0)`, приведены только для справки. Подробнее смотрите на странице объекта API, указанной в таблице или в конце каждого раздела.

### 5. Карты сети

Обзор

Экспорт карты сети содержит:

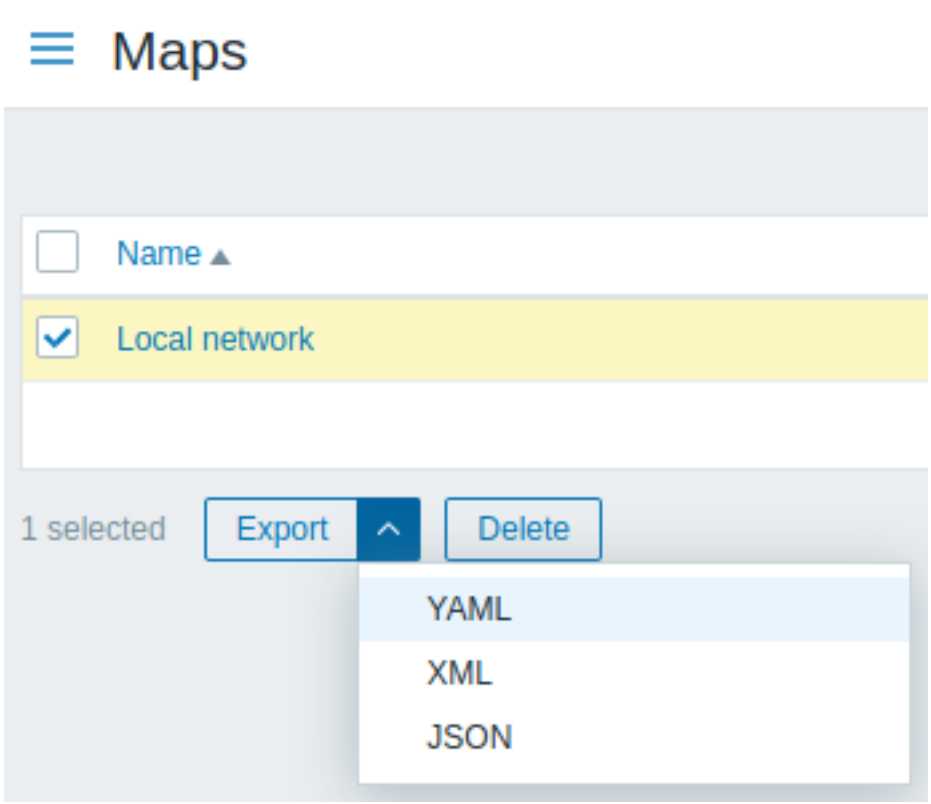
- Все связанные изображения
- Структуру карты (все настройки карты, все находящиеся на карте элементы с их настройками, связи карты и индикаторы состояний связей карты)

**Warning:**
Не экспортируются группы узлов сети, узлы сети, триггеры, другие карты и любые другие элементы, которые могут быть связаны с экспортируемой картой. Поэтому если хотя бы один из элементов, на который ссылается карта, отсутствует, импорт завершится с ошибкой.

Экспорт

Для экспорта карт сети сделайте следующее:

1. Перейдите к: *Мониторинг → Карты сети (Monitoring → Maps)*
2. Отметьте карты сети, которые нужно экспортировать
3. Нажмите на *Экспорт (Export)* ниже списка



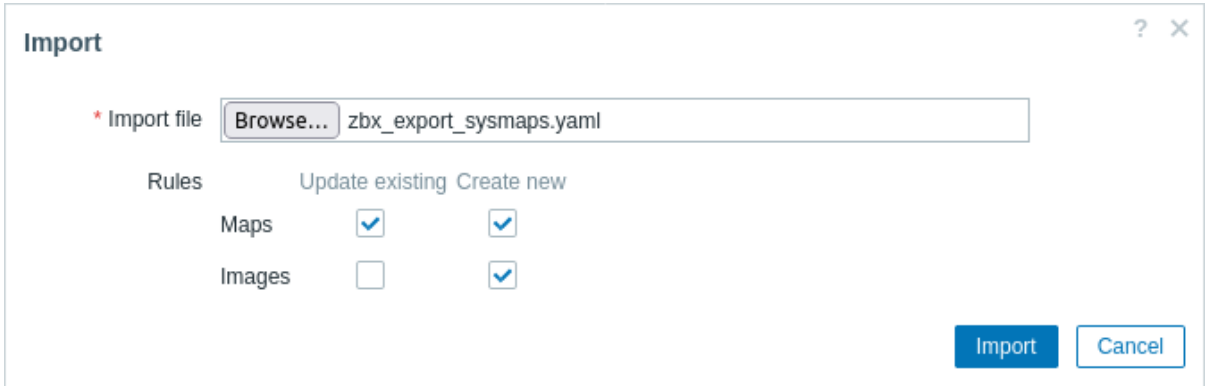

В зависимости от выбранного формата, карты сети экспортируются в локальный файл с именем по умолчанию:

- `zabbix_export_maps.yaml` — при экспорте в YAML (опция по умолчанию для экспорта)
- `zabbix_export_maps.xml` — при экспорте в XML
- `zabbix_export_maps.json` — при экспорте в JSON

Импорт

Чтобы импортировать карты сети, выполните следующие действия:

1. Перейдите в *Monitoring* → *Maps*.
2. Нажмите *Import* в правом верхнем углу.
3. Выберите файл импорта.
4. Отметьте нужные параметры в правилах импорта.
5. Нажмите *Import* в правом нижнем углу формы настройки.




Правила импорта:

| Rule | Description |
|---|---|
| *Update existing* | Существующие карты будут обновлены с использованием данных из файла импорта. В противном случае они не будут обновлены. |
| *Create new* | Новые карты будут созданы с использованием данных из файла импорта. В противном случае они не будут созданы. |

Если снять оба параметра для карт и установить соответствующие параметры для изображений, будут импортированы только изображения. Импорт изображений доступен только пользователям *Super admin*.

В веб-интерфейсе будет отображено сообщение об успешном или неудачном импорте.

**Warning:**
Если заменить существующее изображение, это повлияет на все карты, которые используют это изображение.

Формат экспорта

Экспорт в YAML:

```
zabbix_export:
  version: '7.4'
  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: {  }
```

**Экспортированные элементы** Экспортированные элементы описаны в таблице ниже.

| Элемент | Тип | Описание |
|---|---|---|
| images | | Корневой элемент для изображений. |
| name | строка | Уникальное имя изображения. |
| imagetype | целое число | Тип изображения.<br>Возможные значения:<br>1 — Изображение;<br>2 — Фон. |
| encodedImage | строка | Изображение в кодировке Base64. |
| maps | | Корневой элемент для карт сети. |

Карты сети

| Элемент | Тип | Описание |
|---|---|---|
| name | строка | Уникальное имя карты. |
| width | целое число | Ширина карты, в пикселях. |
| height | целое число | Высота карты, в пикселях. |
| label_type | целое число | Тип подписи к элементам карты.<br>Возможные значения:<br>0 — Подпись;<br>1 — IP адрес узла сети;<br>2 — Имя элемента;<br>3 — Только состояние;<br>4 — Ничего. |
| label_location | целое число | Расположение подписи к элементам карты по умолчанию.<br>Возможные значения:<br>0 — Снизу;<br>1 — Слева;<br>2 — Справа;<br>3 — Сверху. |

| Элемент | Тип | Описание |
|---|---|---|
| highlight | целое число | Включение подсветки иконок для активных триггеров и состояний узлов сети.<br>Возможные значения:<br>0 — No;<br>1 — Yes. |
| expandproblem | целое число | Отображение триггера с проблемой у элементов с одной проблемой.<br>Возможные значения:<br>0 — No;<br>1 — Yes. |
| markelements | целое число | Подсветка элементов карты, которые недавно изменили своё состояние.<br>Возможные значения:<br>0 — No;<br>1 — Yes. |
| show_unack | целое число | Отображение проблем.<br>Возможные значения:<br>0 — Количество всех проблем;<br>1 — Количество неподтвержденных проблем;<br>2 — Количество подтвержденных и неподтвержденных проблем раздельно. |
| severity_min | целое число | Минимальная важность триггеров, которая отображается на карте, по умолчанию.<br>Возможные значения:<br>0 — Не классифицировано;<br>1 — Информация;<br>2 — Предупреждение;<br>3 — Средняя;<br>4 — Высокая;<br>5 — Чрезвычайная. |
| show_suppressed | целое число | Отображение проблем, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.<br>Возможные значения:<br>0 — No;<br>1 — Yes. |
| grid_size | целое число | Размер ячейки сетки карты в пикселях.<br>Поддерживается, если `grid_show` равно 0.<br>Возможные значения: 20, 40, 50, 75 или 100. |
| grid_show | целое число | Отображение сетки в настройке карты.<br>Возможные значения:<br>0 — Yes;<br>1 — No. |
| grid_align | целое число | Автоматическое выравнивание иконок в настройке карты.<br>Возможные значения:<br>0 — Yes;<br>1 — No. |
| label_format | целое число | Использование расширенной конфигурации подписей.<br>Возможные значения:<br>0 — No;<br>1 — Yes. |
| label_type_host | целое число | Метод отображения подписи к узлам сети.<br>Поддерживается, если `label_format` равно 1.<br>Возможные значения:<br>0 — Подпись;<br>1 — IP адрес узла сети;<br>2 — Имя элемента;<br>3 — Только состояние;<br>4 — Ничего;<br>5 — пользовательская подпись. |

| Элемент | Тип | Описание |
|---|---|---|
| label_type_hostgroup | целое<br>число | Метод отображения подписи к группам узлов сети.<br>Поддерживается, если `label_format` равно 1.<br>Возможные значения:<br>0 — Подпись;<br>2 — Имя элемента;<br>3 — Только состояние;<br>4 — Ничего;<br>5 — Пользовательская подпись. |
| label_type_trigger | целое<br>число | Метод отображения подписи к триггеров.<br>Поддерживается, если `label_format` равно 1.<br>Возможные значения:<br>0 — Подпись;<br>2 — Имя элемента;<br>3 — Только состояние;<br>4 — Ничего;<br>5 — Пользовательская подпись. |
| label_type_map | целое<br>число | Метод отображения подписи к картам сети.<br>Поддерживается, если `label_format` равно 1.<br>Возможные значения:<br>0 — Подпись;<br>2 — Имя элемента;<br>3 — Только состояние;<br>4 — Ничего;<br>5 — Пользовательская подпись. |
| label_type_image | целое<br>число | Метод отображения подписи к изображениям.<br>Поддерживается, если `label_format` равно 1.<br>Возможные значения:<br>0 — Label;<br>2 — Element name;<br>4 — Nothing;<br>5 — Custom label. |
| label_string_host | строка | Пользовательская подпись к элементам узлов сети.<br>Поддерживается, если `label_type_host` равно 5. |
| label_string_hostgroup | строка | Пользовательская подпись к элементам групп узлов сети.<br>Поддерживается, если `label_type_hostgroup` равно 5. |
| label_string_trigger | строка | Пользовательская подпись к элементам триггеров.<br>Поддерживается, если `label_type_trigger` равно 5. |
| label_string_map | строка | Пользовательская подпись к элементам карт сети.<br>Поддерживается, если `label_type_map` равно 5. |
| label_string_image | строка | Пользовательская подпись к элементам изображений.<br>Поддерживается, если `label_type_image` равно 5. |
| expand_macros | целое<br>число | Раскрытие макросов в подписях при настройке карты.<br>Возможные значения:<br>0 — No;<br>1 — Yes. |
| background | | Корневой элемент для фонового изображения (если имеется).<br>Поддерживается, если `imagetype` равно 2. |
| name | строка | Имя фонового изображения. |
| iconmap | | Корневой элемент для соответствия иконок (если имеется). |
| name | строка | Имя соответствия иконок. |
| urls | | Корневой элемент для URL, используемых картами сети либо каждым элементом карты. |
| name | строка | Имя ссылки. |
| url | строка | Сама ссылка URL. |
| elementtype | целое<br>число | Тип элемента карты, к которому относится ссылка.<br>Возможные значения:<br>0 — Узел сети;<br>1 — Карта;<br>2 — Триггер;<br>3 — Группа узлов сети;<br>4 — Изображение. |

| Элемент | Тип | Описание |
|---|---|---|
| selements | | Корневой элемент для элементов карты. |
| shapes | | Корневой элемент для фигур карты. |
| type | целое число | Тип фигуры.<br>Возможные значения:<br>0 — Прямоугольник;<br>1 — Эллипс. |
| x | целое число | Координата X фигуры (в пикселях). |
| y | целое число | Координата Y фигуры (в пикселях). |
| width | целое число | Ширина фигуры. |
| height | целое число | Высота фигуры. |
| text | строка | Текст внутри фигуры. |
| font | целое число | Стиль шрифта для текста.<br>Возможные значения:<br>0 — Georgia, serif;<br>1 — ”Palatino Linotype”, ”Book Antiqua”, Palatino, serif;<br>2 — ”Times New Roman”, Times, serif;<br>3 — Arial, Helvetica, sans-serif;<br>4 — ”Arial Black”, Gadget, sans-serif;<br>5 — ”Comic Sans MS”, cursive, sans-serif;<br>6 — Impact, Charcoal, sans-serif;<br>7 — ”Lucida Sans Unicode”, ”Lucida Grande”, sans-serif;<br>8 — Tahoma, Geneva, sans-serif;<br>9 — ”Trebuchet MS”, Helvetica, sans-serif;<br>10 — Verdana, Geneva, sans-serif;<br>11 — ”Courier New”, Courier, monospace;<br>12 — ”Lucida Console”, Monaco, monospace. |
| font_size | целое число | Размер шрифта в пикселях. |
| font_color | строка | Цвет шрифта, представленный в виде 16-ричного кода. |
| text_halign | целое число | Выравнивание текста по горизонтали.<br>Возможные значения:<br>0 — По центру;<br>1 — По левой стороне;<br>2 — По правой стороне. |
| text_valign | целое число | Выравнивание текста по вертикали.<br>Возможные значения:<br>0 — По середине;<br>1 — По верхнему краю;<br>2 — По нижнему краю. |
| border_type | целое число | Тип рамки фигуры.<br>Возможные значения:<br>0 — Отсутствует;<br>1 — Жирная линия;<br>2 — Точечная линия;<br>3 — Пунктирная линия. |
| border_width | целое число | Ширина рамки в пикселях. |
| border_color | строка | Цвет рамки, представленный в виде 16-ричного кода. |
| background_color | строка | Цвет фона (заливки), представленный в виде 16-ричного кода. |
| zindex | целое число | Значение, используемое для указания порядка всех фигур и линий (z-index). |
| lines | | Корневой элемент для линий карты. |
| x1 | целое число | Координата X точки 1 линии в пикселях. |
| y1 | целое число | Координата Y точки 1 линии в пикселях. |
| x2 | целое число | Координата X точки 2 линии в пикселях. |

| Элемент | Тип | Описание |
|---|---|---|
| y2 | целое число | Координата Y точки 2 линии в пикселях. |
| line_type | целое число | Тип линии.<br>Возможные значения:<br>0 — Отсутствует;<br>1 — Жирная линия;<br>2 — Точечная линия;<br>3 — Пунктирная линия. |
| line_width | целое число | Ширина линии в пикселях. |
| line_color | строка | Цвет линии, представленный в виде 16-ричного кода. |
| zindex | целое число | Значение, используемое для указания порядка всех фигур и линий (z-index). |
| links | | Корневой элемент для связей между элементами карты. |
| drawtype | целое число | Стиль связи.<br>Возможные значения:<br>0 — Отсутствует;<br>1 — Жирная линия;<br>2 — Точечная линия;<br>3 — Пунктирная линия. |
| color | строка | Цвет связи (6 символов, hex). |
| label | строка | Подпись к связи. |
| selementid1 | идентификатор | Идентификатор первого элемента, с которым соединена связь. |
| selementid2 | идентификатор | Идентификатор второго элемента, с которым соединена связь. |
| linktriggers | | Корневой элемент для индикаторов состояния связей. |

**Note:**
Смотрите также: Объект карты сети (смотрите соответствующее свойство с совпадающим именем).

Элементы карты

| Элемент | Тип | Описание |
|---|---|---|
| elementtype | целое число | Тип элемента карты.<br>Возможные значения:<br>0 — Узел сети;<br>1 — Карта;<br>2 — Триггер;<br>3 — Группа узлов сети;<br>4 — Изображение. |
| elements | | Корневой элемент для объектов Zabbix (узел сети, группа узлов сети, карта и т.д.), которые представлены на карте.<br>Все объекты имеют свой соответствующий элемент (`host`, и т.д.; смотрите, например, Формат экспорта). |
| label | строка | Подпись к иконке. |
| label_location | целое число | Расположение подписи.<br>Возможные значения:<br>-1 — Использование умолчаний карты;<br>0 — Снизу;<br>1 — Слева;<br>2 — Справа;<br>3 — Сверху. |
| x | целое число | Расположение на оси X. |
| y | целое число | Расположение на оси Y. |

| Элемент | Тип | Описание |
|---|---|---|
| elementsubtype | целое<br>число | Подтип элемента.<br>Поддерживается, если `elementtype` равно 3.<br>Возможные значения:<br>0 — Одна группа узлов сети;<br>1 — Все узлы сети группы. |
| areatype | целое<br>число | Размер области.<br>Поддерживается, если `elementtype` равно 1.<br>Возможные значения:<br>0 — Вписать во всю карту;<br>1 — Пользовательский размер. |
| width | целое<br>число | Ширина области.<br>Поддерживается, если `areatype` равно 1. |
| height | целое<br>число | Высота области.<br>Поддерживается, если `areatype` равно 1. |
| viewtype | целое<br>число | Алгоритм расположения в области.<br>Поддерживается, если `elementsubtype` равно 1.<br>Возможные значения:<br>0 — Располагать равномерно в области. |
| use_iconmap | целое<br>число | Использование соответствия иконок для этого элемента. Уместно, только если соответствие иконок активировано на уровне карты.<br>Возможные значения:<br>0 — Нет;<br>1 — Да. |
| selementid | идентификатор | Уникальный идентификатор записи элемента. |
| icon_off | | Корневой элемент для изображения, используемого, когда элемент в состоянии «OK». |
| name | строка | Уникальное имя элемента. |
| icon_on | | Корневой элемент для изображения, используемого, когда элемент в состоянии «Проблема». |
| name | строка | Уникальное имя элемента. |
| icon_disabled | | Корневой элемент для изображения, используемого, когда элемент деактивирован. |
| name | строка | Уникальное имя элемента. |
| icon_maintenance | | Корневой элемент для изображения, используемого, когда элемент в обслуживании. |
| name | строка | Уникальное имя элемента. |
| urls | | Корневой элемент для URL, используемых картами сети либо каждым элементом карты. |
| name | строка | Имя ссылки. |
| url | строка | Сама ссылка URL. |
| evaltype | целое<br>число | Тип обработки тегов. |
| tags | | Корневой элемент для тегов проблем (для элементов узел сети и группа узлов сети).<br>Если теги указаны, на карте будут отображаться только проблемы с этими тегами. |
| tag | строка | Имя тега. |
| value | строка | Значение тега. |
| operator | целое<br>число | Оператор. |

**Note:**
Смотрите также: Объект карты сети (смотрите соответствующее свойство с совпадающим именем).

Индикаторы состояния связей карты

| Элемент | Тип | Описание |
|---|---|---|
| drawtype | целое число | Стиль связи, когда триггер в состоянии «Проблема». Возможные значения:<br>0 — Линия;<br>2 — Жирная линия;<br>3 — Точечная линия;<br>4 — Пунктирная линия. |
| color | строка | Цвет связи (6 символов, hex), когда триггер в состоянии «Проблема». |
| trigger | | Корневой элемент для триггера, используемого для индикации состояния связи. |
| description | строка | Имя триггера. |
| expression | строка | Выражение триггера. |
| recovery_expression | строка | Выражение восстановления триггера. |

**Note:**
Смотрите также: Объект карты сети (смотрите соответствующее свойство с совпадающим именем).

**6. Способы оповещений**

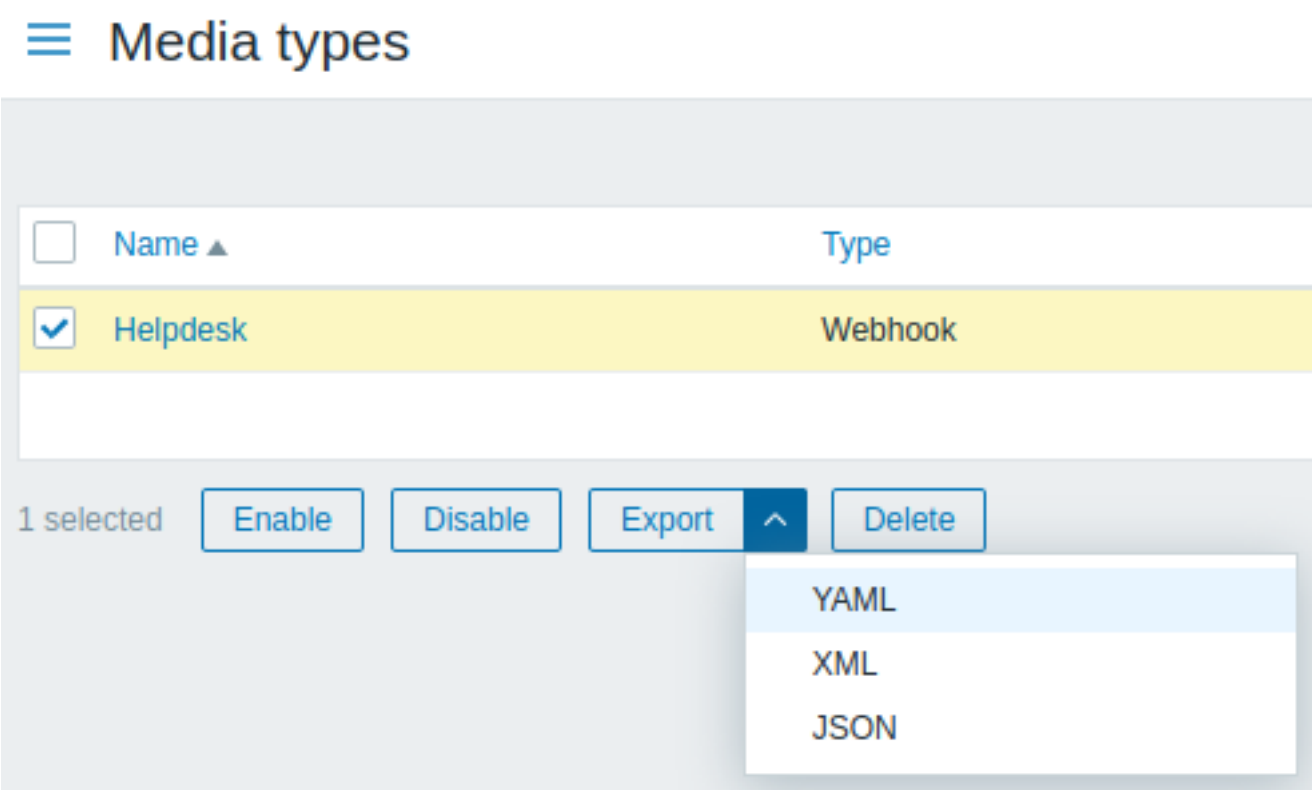
Обзор

Способы оповещений экспортируются вместе со всеми связанными объектами и со связями между объектами.

Экспорт

Для экспорта способов оповещений сделайте следующее:

1. Перейдите к: *Оповещения → Способы оповещений (Alerts → Media types)*
2. Отметьте способы оповещений, которые нужно экспортировать
3. Нажмите на *Экспорт (Export)* ниже списка



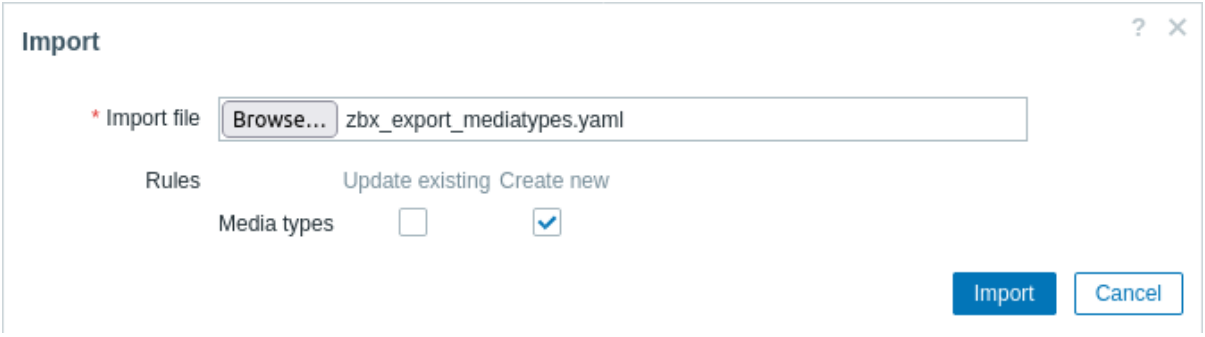

В зависимости от выбранного формата, способы оповещений экспортируются в локальный файл с именем по умолчанию:

- `zabbix_export_mediatypes.yaml` — при экспорте в YAML (опция по умолчанию для экспорта);
- `zabbix_export_mediatypes.xml` — при экспорте в XML;
- `zabbix_export_mediatypes.json` — при экспорте в JSON.

Импорт

Чтобы импортировать типы медиа, выполните следующие действия:

1. Перейдите в *Alerts* → *Media types*.
2. Нажмите *Import* в правом верхнем углу.
3. Выберите файл импорта.
4. Отметьте необходимые параметры в правилах импорта.
5. Нажмите *Import* в правом нижнем углу формы настройки.




Правила импорта:

| Rule | Description |
|---|---|
| *Update existing* | Существующие элементы будут обновлены с использованием данных из файла импорта. В противном случае они не будут обновлены. |
| *Create new* | Новые элементы будут созданы с использованием данных из файла импорта. В противном случае они не будут созданы. |

Сообщение об успешном или неудачном импорте будет отображено в веб-интерфейсе.

Формат экспорта

Экспорт в YAML:

```yaml
zabbix_export:
  version: '7.4'
  media_types:
    - name: Pushover
      type: WEBHOOK
      parameters:
        - name: endpoint
          value: 'https://api.pushover.net/1/messages.json'
        - name: eventid
          value: '{EVENT.ID}'
        - name: event_nseverity
          value: '{EVENT.NSEVERITY}'
        - name: event_source
          value: '{EVENT.SOURCE}'
        - name: event_value
          value: '{EVENT.VALUE}'
        - name: expire
          value: '1200'
        - name: message
          value: '{ALERT.MESSAGE}'
        - name: priority_average
          value: '0'
        - name: priority_default
          value: '0'
        - name: priority_disaster
          value: '0'
        - name: priority_high
          value: '0'
        - name: priority_information
          value: '0'
```

```
    - name: priority_not_classified
      value: '0'
    - name: priority_warning
      value: '0'
    - name: retry
      value: '60'
    - name: title
      value: '{ALERT.SUBJECT}'
    - name: token
      value: '<PUSHOVER TOKEN HERE>'
    - name: triggerid
      value: '{TRIGGER.ID}'
    - name: url
      value: '{$ZABBIX.URL}'
    - name: url_title
      value: Zabbix
    - name: user
      value: '{ALERT.SENDTO}'
  status: DISABLED
  max_sessions: '0'
  script: |
    try {
        var params = JSON.parse(value),
            request = new HttpRequest(),
            data,
            response,
            severities = [
                {name: 'not_classified', color: '#97AAB3'},
                {name: 'information', color: '#7499FF'},
                {name: 'warning', color: '#FFC859'},
                {name: 'average', color: '#FFA059'},
                {name: 'high', color: '#E97659'},
                {name: 'disaster', color: '#E45959'},
                {name: 'resolved', color: '#009900'},
                {name: 'default', color: '#000000'}
            ],
            priority;

        if (typeof params.HTTPProxy === 'string' && params.HTTPProxy.trim() !== '') {
            request.setProxy(params.HTTPProxy);
        }

        if ([0, 1, 2, 3].indexOf(parseInt(params.event_source)) === -1) {
            throw 'Incorrect "event_source" parameter given: "' + params.event_source + '".\nMust be 0
        }

        if (params.event_value !== '0' && params.event_value !== '1'
            && (params.event_source === '0' || params.event_source === '3')) {
            throw 'Incorrect "event_value" parameter given: ' + params.event_value + '\nMust be 0 or 1
        }

        if ([0, 1, 2, 3, 4, 5].indexOf(parseInt(params.event_nseverity)) === -1) {
            params.event_nseverity = '7';
        }

        if (params.event_value === '0') {
            params.event_nseverity = '6';
        }

        priority = params['priority_' + severities[params.event_nseverity].name] || params.priority_de

        if (isNaN(priority) || priority < -2 || priority > 2) {
```

```
        throw '"priority" should be -2..2';
    }

    if (params.event_source === '0' && isNaN(params.triggerid)) {
        throw 'field "triggerid" is not a number';
    }

    if (isNaN(params.eventid)) {
        throw 'field "eventid" is not a number';
    }

    if (typeof params.message !== 'string' || params.message.trim() === '') {
        throw 'field "message" cannot be empty';
    }

    data = {
        token: params.token,
        user: params.user,
        title: params.title,
        message: params.message,
        url: (params.event_source === '0')
            ? params.url + '/tr_events.php?triggerid=' + params.triggerid + '&eventid=' + params.e
            : params.url,
        url_title: params.url_title,
        priority: priority
    };

    if (priority == 2) {
        if (isNaN(params.retry) || params.retry < 30) {
            throw 'field "retry" should be a number with value of at least 30 if "priority" is set
        }

        if (isNaN(params.expire) || params.expire > 10800) {
            throw 'field "expire" should be a number with value of at most 10800 if "priority" is
        }

        data.retry = params.retry;
        data.expire = params.expire;
    }

    data = JSON.stringify(data);
    Zabbix.log(4, '[ Pushover Webhook ] Sending request: ' + params.endpoint + '\n' + data);

    request.addHeader('Content-Type: application/json');
    response = request.post(params.endpoint, data);

    Zabbix.log(4, '[ Pushover Webhook ] Received response with status code ' + request.getStatus()

    if (response !== null) {
        try {
            response = JSON.parse(response);
        }
        catch (error) {
            Zabbix.log(4, '[ Pushover Webhook ] Failed to parse response received from Pushover');
            response = null;
        }
    }

    if (request.getStatus() != 200 || response === null || typeof response !== 'object' || respons
        if (response !== null && typeof response === 'object' && typeof response.errors === 'objec
                && typeof response.errors[0] === 'string') {
            throw response.errors[0];
```

```
            }
            else {
                throw 'Unknown error. Check debug log for more information.';
            }
        }

        return 'OK';
    }
    catch (error) {
        Zabbix.log(4, '[ Pushover Webhook ] Pushover notification failed: ' + error);
        throw 'Pushover notification failed: ' + error;
    }
description: |
    .                  : https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/media/pus

             token,                Pushover.
               Pushover        Zabbix -                      "send to".
message_templates:
  - event_source: TRIGGERS
    operation_mode: PROBLEM
    subject: '     : {EVENT.NAME}'
    message: |
                  {EVENT.TIME} {EVENT.DATE}
            : {EVENT.NAME}
           : {HOST.NAME}
            : {EVENT.SEVERITY}
                : {EVENT.OPDATA}
          ID      : {EVENT.ID}
      {TRIGGER.URL}
  - event_source: TRIGGERS
    operation_mode: RECOVERY
    subject: '       {EVENT.DURATION}: {EVENT.NAME}'
    message: |
                    {EVENT.RECOVERY.TIME} {EVENT.RECOVERY.DATE}
            : {EVENT.NAME}
                : {EVENT.DURATION}
           : {HOST.NAME}
            : {EVENT.SEVERITY}
          ID      : {EVENT.ID}
      {TRIGGER.URL}
  - event_source: TRIGGERS
    operation_mode: UPDATE
    subject: '              {EVENT.AGE}: {EVENT.NAME}'
    message: |
      {USER.FULLNAME} {EVENT.UPDATE.ACTION}        {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}.
      {EVENT.UPDATE.MESSAGE}

                    : {EVENT.STATUS},     : {EVENT.AGE},        : {EVENT.ACK.STATUS}.
  - event_source: DISCOVERY
    operation_mode: PROBLEM
    subject: '      : {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}'
    message: |
                : {DISCOVERY.RULE.NAME}

      IP       : {DISCOVERY.DEVICE.IPADDRESS}
      DNS       : {DISCOVERY.DEVICE.DNS}
                : {DISCOVERY.DEVICE.STATUS}
                    : {DISCOVERY.DEVICE.UPTIME}

                  : {DISCOVERY.SERVICE.NAME}
                   : {DISCOVERY.SERVICE.PORT}
                    : {DISCOVERY.SERVICE.STATUS}
```

```
                              : {DISCOVERY.SERVICE.UPTIME}
      - event_source: AUTOREGISTRATION
        operation_mode: PROBLEM
        subject: '                : {HOST.HOST}'
        message: |
                  : {HOST.HOST}
          IP       : {HOST.IP}
                 : {HOST.PORT}
```

Экспортируемые элементы

Экспортируемые элементы описаны в таблицах ниже.

| Элемент | Тип | Описание |
|---|---|---|
| name | строка | (обязательный) Имя способа оповещения. |
| type | строка | (обязательный) Транспорт, используемый способом оповещения.<br>Возможные значения:[1] EMAIL (0), SMS (1), SCRIPT (2), WEBHOOK (4). |
| status | строка | Активирован ли способ оповещения.<br>Возможные значения:[1] ENABLED (0, по умолчанию), DISABLED (1) |
| max_sessions | целое число | Максимальное количество уведомлений, которые могут обрабатываться параллельно.<br>Возможные значения для SMS:[1] 1 (по умолчанию).<br>Возможные значения для остальных способов оповещения:[1] 0-100 где 0 — без ограничений). |
| attempts | целое число | Максимальное количество попыток отослать уведомление.<br>Возможные значения:[1] 1-10 (по умолчанию: 3). |
| attempt_interval | строка | Интервал (используя секунды или единицы времени с суффиксом) между повторными попытками.<br>Возможные значения:[1] 0-60s (по умолчанию: 10s). |
| description | строка | Описание способа оповещения. |
| message_templates | | Корневой элемент для шаблонов сообщения способа оповещения. |
| event_source | строка | (обязательный) Источник события.<br>Возможные значения:[1] TRIGGERS (0), DISCOVERY (1), AUTOREGISTRATION (2), INTERNAL (3), SERVICE (4). |
| operation_mode | строка | Режим работы.<br>Возможные значения:[1] PROBLEM (0), RECOVERY (1), UPDATE (2). |
| subject | строка | Тема сообщения. |
| message | строка | Текст сообщения. |

**Note:**
Смотрите также: Объект способа оповещения (смотрите соответствующее свойство с совпадающим именем).

**Email**

Следующие дополнительные элементы экспортируются только для типа медиа *Email*.

| Элемент | Тип | Описание |
|---|---|---|
| provider | string | Поставщик электронной почты. |
| smtp_server | string | SMTP-сервер. |
| smtp_port | integer | Порт SMTP-сервера для подключения.<br>По умолчанию: 25. |
| smtp_helo | string | SMTP helo. |
| smtp_email | string | Адрес электронной почты, с которого будут отправляться уведомления. |
| smtp_security | string | Уровень безопасности SMTP-соединения для использования.<br>Возможные значения:[1] NONE (0, по умолчанию), STARTTLS (1), SSL_OR_TLS (2). |
| smtp_verify_host | string | Проверка SSL host для SMTP.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| smtp_verify_peer | string | Проверка SSL peer для SMTP.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| smtp_authentication | string | Метод SMTP-аутентификации для использования.<br>Возможные значения:[1] NONE (0, по умолчанию), PASSWORD (1), OAUTH (2). |

| Элемент | Тип | Описание |
|---|---|---|
| username | string | Имя пользователя. |
| password | string | Пароль аутентификации. |
| redirection_url | string | URL веб-интерфейса Zabbix для возврата после авторизации OAuth. |
| client_id | string | Идентификатор клиента, зарегистрированный на сервере авторизации OAuth. |
| authorization_url | string | URL OAuth с параметрами для получения токенов доступа и обновления. |
| token_url | string | URL OAuth для обмена токена авторизации на токены доступа и обновления. |
| message_format | string | Формат сообщения.<br>Возможные значения:[1] TEXT (0), HTML (1, по умолчанию). |

**Note:**
См. также: Объект Media type (обратитесь к соответствующему свойству с совпадающим именем).

**SMS**

Следующие дополнительные элементы экспортируются только для способа оповещения *SMS*.

| Элемент | Тип | Описание |
|---|---|---|
| gsm_modem | строка | (обязательный) Имя последовательного устройства модема GSM. |

**Note:**
Смотрите также: Объект способа оповещения (смотрите соответствующее свойство с совпадающим именем).

**Скрипт**

Следующие дополнительные элементы экспортируются только для способа оповещения *Скрипт (Script)*.

| Element | Type | Description |
|---|---|---|
| script name | строка | (обязательный) Имя скрипта. |
| parameters | | Корневой элемент для параметров скрипта. |
| sortorder | строка | (обязательный) Порядок параметров скрипта, передаваемых скрипту в качестве аргументов командной строки. |
| value | строка | Значение параметра скрипта. |

**Note:**
Смотрите также: Объект способа оповещения (смотрите соответствующее свойство с совпадающим именем).

**Вебхук**

Следующие дополнительные элементы экспортируются только для способа оповещения *Вебхук (Webhook)*.

| Элемент | Тип | Описание |
|---|---|---|
| script | строка | Скрипт. |
| timeout | строка | Время ожидания запроса HTTP при выполнении скрипта JavaScript.<br>Possible values:[1] 1-60s (по умолчанию: 30s). |
| process_tags | строка | Следует ли обрабатывать возвращаемые теги.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| show_event_menu | строка | Если {EVENT.TAGS.*} были успешно раскрыты в полях `event_menu_url` и `event_menu_name`, это поле означает присутствие пункта в меню события.<br>Возможные значения:[1] NO (0, по умолчанию), YES (1). |
| event_menu_url | строка | URL пункта меню события. Поддерживаются макросы {EVENT.TAGS.*}. |
| event_menu_name | строка | Имя пункта меню события. Поддерживаются макросы {EVENT.TAGS.*}. |
| parameters | | Корневой элемент для параметров способа оповещений Вебхук. |
| name | строка | (обязательный) Имя параметра вебхука. |
| value | строка | Значение параметра вебхука. |

**Note:**
Смотрите также: Объект способа оповещения (смотрите соответствующее свойство с совпадающим именем).

Примечания

[1] Целочисленные значения API в скобках, например, ENABLED (0), упоминаются только для справки. Для получения дополнительной информации смотрите связанную страницу объекта API в записи таблицы или в конце каждого раздела.

## 15. Обнаружение

**Обзор** Функция обнаружения предоставляет автоматические способы добавления узлов сети и настройки мониторинга без ручного конфигурирования. Zabbix поддерживает несколько методов обнаружения, что позволяет адаптироваться к различным сценариям.

**Методы обнаружения**

- Обнаружение сети - автоматически обнаруживать узлы сети и службы, доступные в вашей сети.
- Автоматическая регистрация активного агента - разрешить активным агентам автоматически регистрировать себя на сервере.
- Низкоуровневое обнаружение - автоматически создавать элементы данных, триггеры, графики и узлы сети для обнаруженных сущностей, таких как файловые системы или сетевые интерфейсы, с использованием прототипов и вложенных правил обнаружения.

**1. Сетевое обнаружение**

Обзор

Zabbix предлагает функционал эффективного и очень гибкого автоматического обнаружения сетевых устройств.

При наличии настроенного должным образом сетевого обнаружения вы сможете:

- ускорить развёртывание Zabbix
- упростить администрирование
- использовать Zabbix в быстро меняющихся средах без излишнего администрирования

Обнаружение сетевых устройств в Zabbix базируется на основе следующей информации:

- диапазоны IP
- доступность внешних сервисов (FTP, SSH, WEB, POP3, IMAP, TCP и т.д.)
- информация, полученная от Zabbix агента (поддерживается только режим без шифрования)
- информация, полученная от SNMP агента

Обнаружение НЕ предоставляет:

- обнаружение топологии сети

Сетевое обнаружение, по сути, состоит из двух этапов: обнаружение и действия.

Обнаружение

Zabbix периодически сканирует указанные в правилах сетевого обнаружения диапазоны IP. Частота проверок настраивается для каждого правила отдельно.

Каждое правило имеет заданный набор проверок сервисов, которые будут выполняться по диапазону IP.

Правила обнаружения обрабатываются менеджером обнаружения (процесс discovery manager). Менеджер обнаружения создаёт задание для каждого правила со списком задач (сетевых проверок). Сетевые проверки выполняются параллельно доступными рабочими процессами обнаружения (процессы discovery worker, их количество настраивается в веб-интерфейсе для каждого правила). Последовательно планируются только проверки с одинаковым IP-адресом и портом, поскольку некоторые устройства не принимают параллельные подключения к одному и тому же порту.

Размер очереди сетевых проверок ограничен 2000000 или примерно 4 ГБ памяти. Если очередь заполняется, правило обнаружения будет пропущено, а в журнал будет выведено предупреждение. Вы можете использовать внутренний элемент данных `zabbix[discovery_queue]` для отслеживания количества проверок обнаружения в очереди.

Проверки обнаружения обрабатываются независимо от остальных проверок. Если какие-либо проверки не находят сервис (или завершаются с ошибкой), то другие проверки всё равно выполняются.

**Note:**
Если правило обнаружения изменяется во время выполнения, то текущее выполнение обнаружения прерывается.

Каждая проверка сервиса и хоста (IP), выполняемая модулем сетевого обнаружения, генерирует событие обнаружения.

| Событие | Результат проверки сервиса |
|---|---|
| *Сервис обнаружен (Service Discovered)* | Сервис «доступен» после того, как был «недоступен» или когда обнаружен первый раз. |
| *Сервис доступен (Service Up)* | Сервис «доступен» после того, как уже был «доступен». |
| *Сервис потерян (Service Lost)* | Сервис «недоступен» после того, как был «доступен». |
| *Сервис недоступен (Service Down)* | Сервис «недоступен» после того, как уже был «недоступен». |
| *Узел сети обнаружен (Host Discovered)* | По крайней мере один сервис на узле сети «доступен» после того, как все сервисы на этом узле сети были «недоступны», либо обнаружен сервис, относящийся к ещё не зарегистрированному узлу сети. |
| *Узел сети доступен (Host Up)* | По крайней мере один сервис на узле сети «доступен» после того, как по крайней мере один сервис на узле сети уже был «доступен». |
| *Узел сети потерян (Host Lost)* | Все сервисы на узле сети «недоступны» после того, как по крайней мере один был «доступен». |
| *Узел сети недоступен (Host Down)* | Все сервисы на узле сети «недоступны» после того, как они уже были «недоступны». |

Действия

События обнаружения могут быть основой соответствующих действий, таких как:

- Отправка оповещений
- Добавление/удаление узлов сети
- Активация/деактивация узлов сети
- Добавление узлов сети в группу
- Удаление узлов сети из группы
- Добавление тегов к узлу сети
- Удаление тегов с узла сети
- Присоединение шаблона к узлам сети/отсоединение шаблона от узлов сети
- Выполнение удалённых скриптов

Эти действия можно задать с учётом типа устройства, IP, состояния, времени доступности/недоступности и т.п. Для получения более подробных сведений о настройке действий на события, основанных на сетевом обнаружении, смотрите страницы операций действий и условий действий.

Поскольку действия на сетевое обнаружение основаны на событиях, они будут срабатывать и когда обнаруженный узел сети активен (online), и когда неактивен (offline). Крайне рекомендуется добавить условие действия *Состояние обнаружения: доступен*, чтобы избежать ситуации, когда такие действия как *Добавить узел сети* срабатывали бы при событиях *Сервис потерян* / *Сервис недоступен*. В противном случае, если обнаруженный узел сети удаляется вручную, по-прежнему будут генерироваться события *Сервис потерян* / *Сервис недоступен*, и он будет пересоздан во время следующего цикла обнаружения.

**Note:**
Присоединение обнаруженного узла сети к шаблонам будет массово заканчиваться неудачей, если любой из присоединяемых шаблонов имеет уникальные объекты (например, ключ элемента данных), совпадающие с уникальными объектами (т.е. ключом элемента данных), уже присутствующими на этом узле сети или в другом присоединяемом шаблоне.

Создание узлов сети

Узел сети добавляется, если выбрана операция *Добавить узел сети*. Узел сети также добавляется, даже если операция *Добавить узел сети* пропущена, если вы выбрали операции, приводящие к действиям над узлом сети. Это такие операции, как:

- активировать узел сети
- деактивировать узел сети
- добавить узел сети в группу узлов сети
- присоединить шаблон к узлу сети

Созданные узлы сети добавляются в группу *Discovered hosts* (по умолчанию, настраивается в *Администрирование → Общие → Другое* (*Administration → General →Other*)). Если вы желаете, чтобы узлы сети добавлялись в другую группу, добавьте операцию *Удалить из группы узлов сети* (укажите «Discovered hosts») и также добавьте операцию *Добавить в группу узлов сети* (укажите другую группу узлов сети), так как узел сети должен принадлежать по крайней мере одной группе узлов сети.

IP-адрес обнаруженного устройства вместе с источником обнаружения (Zabbix сервер, Zabbix прокси либо группа прокси) и типом интерфейса используются как критерии при поиске узла сети в системе. Если узел сети с таким IP-адресом, типом интерфейса и источником обнаружения уже существует, то он и будет целью для выполняемых операций. Если источник обнаружения отличается, то обнаруженный объект рассматривается как другой узел сети и может быть создан новый узел сети.

Если IP-адрес обнаруженного узла сети изменяется или интерфейс удаляется, то при следующем обнаружении будет создан новый узел сети.

Именование узлов сети

При добавлении узлов сети имя узла сети является результатом обратного преобразования DNS или IP адресом, если обратное преобразование завершилось неудачей. Преобразование выполняется на стороне Zabbix сервера или Zabbix прокси, в зависимости от того, кто выполняет обнаружение. Если преобразование завершается неудачей на прокси, оно не повторяется на стороне сервера. Если узел сети с таким же именем уже существует, следующий узел сети получит добавление к имени **_2**, затем **_3** и так далее.

Также возможно переопределить преобразования DNS/IP и вместо этого для имени узла сети использовать значение элемента данных, например:

- Вы можете обнаружить несколько серверов с помощью Zabbix агента, используя элемент данных Zabbix агента для обнаружения, и автоматически назначить им правильное имя на основании строкового значения, возвращаемого этим элементом данных
- Вы можете обнаружить несколько сетевых устройств SNMP, используя для обнаружения элемент данных агента SNMP, и автоматически назначить им соответствующие имена на основании строкового значения, возвращаемого этим элементом данных

Если имя узла сети было назначено, используя значение элемента данных, оно не обновляется во время последующих проверок обнаружения. Если невозможно назначить имя узла сети, используя значение элемента данных, то используется значение по умолчанию (DNS имя).

Если узел сети с обнаруженным IP адресом уже существует и источник обнаружения (Zabbix сервер, прокси или группа прокси) не изменился, новый узел сети не создаётся. Если источник обнаружения отличается, обнаруженный объект считается другим и может быть создан новый узел сети. Однако, если действие на обнаружение содержит операции (присоединение шаблона, добавление в группу узлов сети и так далее), они будут выполнены над существующим узлом сети, который имеет совпадающие IP адрес, тип интерфейса и источник обнаружения.

Удаление узлов сети

Узлы сети, которые созданы правилом сетевого обнаружения, удаляются автоматически из *Мониторинг → Обнаружение* (*Monitoring → Discovery*), если обнаруженный объект более не входит в диапазон IP адресов этого правила. Узлы сети удаляются немедленно.

Создание интерфейсов при добавлении узлов сети

Когда узлы сети добавляются в результате сетевого обнаружения, они получают интерфейсы, созданные в соответствии со следующими правилами:

- Обнаруженные сервисы — например, если проверка SNMP прошла успешно, то будет создан SNMP интерфейс.
- Если узел сети отвечает на запросы как Zabbix агента, так и на запросы SNMP, будут созданы интерфейсы обоих типов.
- Если критерием уникальности выступают данные с Zabbix агента или с SNMP, первый найденный для узла сети интерфейс будет создан как интерфейс по умолчанию. Другие IP адреса добавляются как дополнительные интерфейсы. Условия действий (такие как IP адрес узла сети) не влияют на добавление интерфейсов. *Обратите внимание*, что это будет работать, если все интерфейсы обнаружены тем же правилом обнаружения. Если различные правила обнаружения находят разные интерфейсы для того же узла сети, добавляется новый узел сети.
- Если узел сети отвечает только на проверки агентов, для него будет создан только агентский интерфейс. Если в дальнейшем он начнёт отвечать на SNMP, дополнительно будет создан SNMP интерфейс.

- Если были созданы 3 различных узла сети, будучи обнаруженными по критерию уникальности «IP», а затем правило обнаружения изменяется таким образом, что узлы сети A, B и C имеют одинаковый результат критерия уникальности, то B и C будут созданы как дополнительные интерфейсы для A, первого узла сети. Отдельные узлы сети B и C останутся. В *Мониторинг* → *Обнаружение* (*Monitoring* →*Discovery*) добавленные интерфейсы будут отображаться в столбце «Обнаруженное устройство (Discovered device)», чёрным шрифтом и с отступом; однако столбец «Наблюдаемый узел сети (Monitored host)» будет отображать только A, первый созданный узел сети. «Доступен/Недоступен (Uptime/Downtime)» не измеряется для IP адресов, которые считаются дополнительными интерфейсами.

Изменение настроек прокси

Узлы сети, обнаруженные различными прокси, не всегда рассматриваются как разные. Обнаружение и проверки уникальности зависят от структуры групп прокси: когда прокси выполняет правило обнаружения и создаёт узел сети, этот узел сети добавляется в родительскую группу прокси этого прокси-сервера, а не назначается самому прокси-серверу. Когда Zabbix оценивает уникальность IP во время обнаружения, проверяются узлы сети, контролируемые родительской группой прокси. Узлы сети, контролируемые отдельными прокси-серверами внутри этой группы (в том числе прокси-сервером, выполняющим обнаружение), игнорируются при проверке уникальности, что может привести к дублированию узлов сети, если несколько прокси-серверов контролируют перекрывающиеся подсети.

В то время как такое поведение позволяет обнаружению выполняться на перекрывающихся диапазонах IP, используемых различными подсетями, замена прокси, назначенного уже контролируемой подсети, является более сложным процессом, поскольку во избежание дубликатов нужно согласованно поменять прокси на обнаруженных узлах сети, а также поменять членство в родительской группе прокси.

Например, шаги по замене прокси в правиле обнаружения:

1. деактивировать правило обнаружения
2. засинхронизировать конфигурацию прокси
3. заменить прокси в правиле обнаружения
4. заменить прокси для всех узлов сети, обнаруженных этим правилом (чтобы избежать дубликатов, убедитесь, что узлы сети в родительской группе прокси-серверов и любые узлы сети, отслеживаемые отдельными прокси-серверами в этой группе, обновлены)
5. активировать правило обнаружения

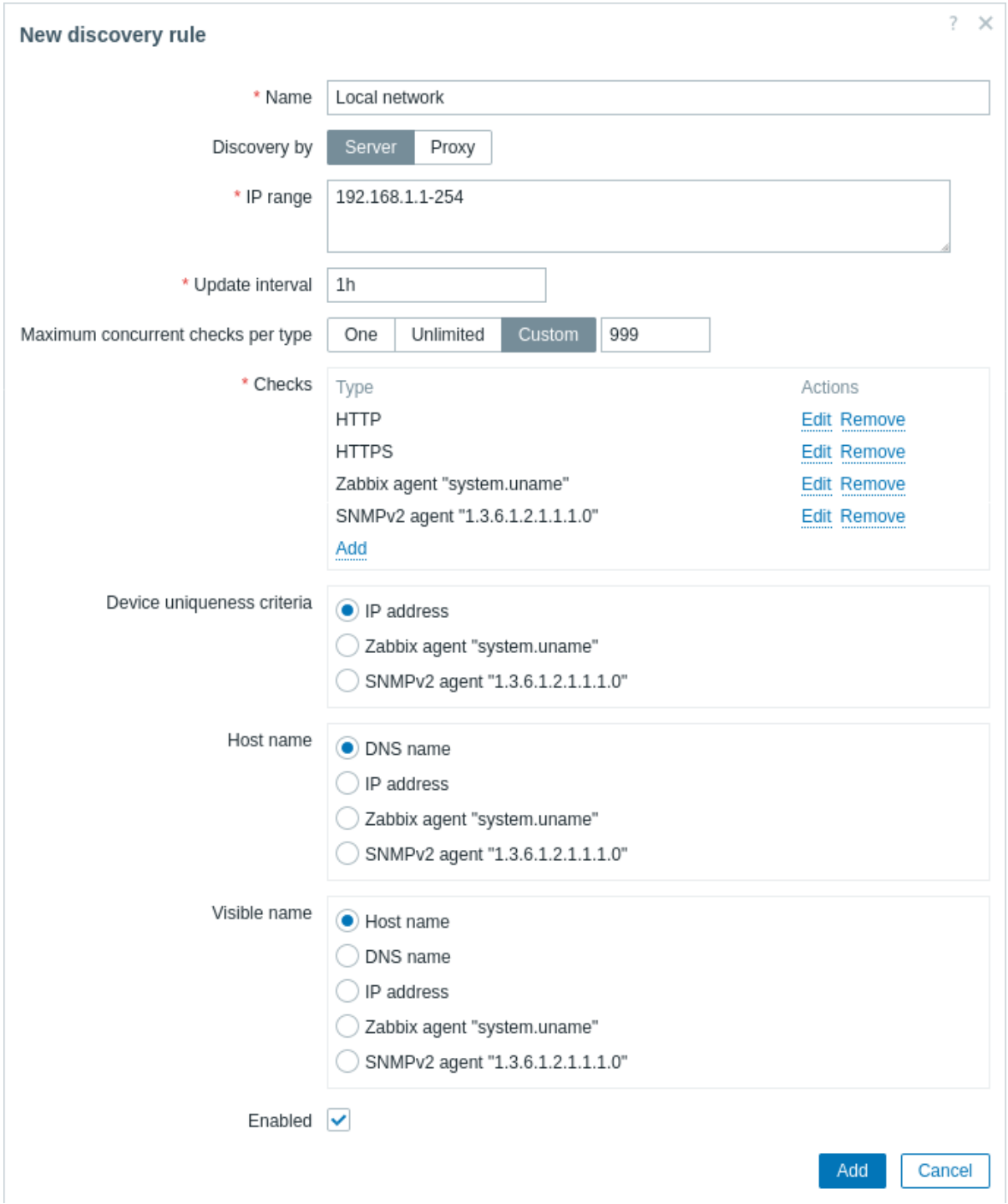
**1. Настройка правила сетевого обнаружения**

Обзор

Чтобы настроить правило обнаружения сети, используемое Zabbix для обнаружения узлов сети и сервисов:

- Перейдите в *Сбор данных* → *Обнаружение*
- Нажмите *Создать правило обнаружения* (или на имя правила, чтобы изменить существующее)
- Отредактируйте атрибуты правила обнаружения

Атрибуты правила




Все обязательные поля ввода отмечены красной звездочкой.

| Parameter | Description |
|---|---|
| *Name* | Уникальное имя правила. Например, ”Local network”. |
| *Discovery by* | Обнаружение выполняется:<br>**Server** - сервером Zabbix<br>**Proxy** - прокси Zabbix (выбирается в поле имени прокси) |
| *IP range* | Диапазон IP-адресов для обнаружения. Он может иметь следующие форматы:<br>Один IP-адрес: 192.168.1.33<br>Диапазон IP-адресов: 192.168.1-10.1-255. Диапазон ограничен общим количеством охватываемых адресов (менее 64K).<br>Маска IP: 192.168.4.0/24<br>поддерживаемые маски IP:<br>/16 - /30 для IPv4-адресов<br>/112 - /128 для IPv6-адресов<br>Список: 192.168.1.1-255, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24<br>Это поле поддерживает пробелы, табуляцию и несколько строк. |

| Parameter | Description |
|---|---|
| *Update interval* | Этот параметр определяет, как часто Zabbix будет выполнять правило.<br>Интервал отсчитывается после завершения выполнения предыдущего экземпляра обнаружения, поэтому наложения не происходит.<br>Поддерживаются суффиксы времени, например 30s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы.<br>*Примечание*: если используется пользовательский макрос и его значение изменяется (например, 1w → 1h), следующий проверочный запуск будет выполнен в соответствии с предыдущим значением (в примере - в более отдаленном будущем). |
| *Maximum concurrent checks per type* | Установите максимальное количество потоков обнаружения (workers) на одну проверку службы для параллельной обработки проверок обнаружения:<br>**One** - один поток<br>**Unlimited** - неограниченное количество потоков (но не больше, чем в параметре StartDiscoverers)<br>**Custom** - задайте пользовательское количество потоков (0-999)<br>Обратите внимание, что все правила обнаружения с асинхронными проверками службы SNMPv3 обрабатываются одним worker из-за особенностей реализации libsnmp, то есть увеличение числа workers не приведет к увеличению скорости обнаружения. |
| *Checks* | Zabbix будет использовать этот список проверок для обнаружения. Нажмите Add для настройки новой проверки во всплывающем окне.<br>Поддерживаемые проверки: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Telnet, Zabbix agent, SNMPv1 agent, SNMPv2 agent, SNMPv3 agent, ICMP ping.<br>Обнаружение на основе протоколов использует функциональность **net.tcp.service[]** для проверки каждого узла сети, за исключением SNMP, который выполняет запрос SNMP OID.<br>Агент Zabbix проверяется путем запроса элемента данных в незашифрованном режиме.<br>Подробнее см. элементы данных агента.<br>Параметр 'Ports' может иметь один из следующих видов:<br>Один порт: 22<br>Диапазон портов: 22-45<br>Список: 22-45,55,60-70<br>Начиная с Zabbix 7.0, все проверки служб выполняются асинхронно, кроме проверок LDAP.<br>Начиная с Zabbix 7.0, проверка HTTP/HTTPS выполняется через libcurl. Если Zabbix server/proxy собран без libcurl, то проверки HTTP будут работать как в предыдущих версиях (то есть как проверки TCP), но проверки HTTPS работать не будут. |
| *Device uniqueness criteria* | Критерии уникальности могут быть следующими:<br>**IP address** - не обрабатывать несколько устройств с одним IP-адресом. Если устройство с таким же IP уже существует, оно будет считаться уже обнаруженным, и новый узел сети не будет добавлен.<br>**<discovery check>** - проверка агента Zabbix или SNMP agent.<br>Обратите внимание, что критерии уникальности, используемые во время обнаружения, не совпадают с идентификацией узла сети в системе при выполнении действий. Критерии уникальности во время обнаружения определяют, являются ли два или более обнаруженных устройства одинаковыми (или разными), тогда как в Zabbix критерием идентификации узла сети является только IP-адрес (см. Создание узла сети). |
| *Host name* | Задайте техническое имя узла сети для создаваемого узла сети, используя:<br>**DNS name** - DNS-имя (по умолчанию)<br>**IP address** - IP-адрес<br>**<discovery check>** - полученное строковое значение проверки обнаружения (например, проверка агента Zabbix, SNMP agent)<br>См. также: Именование узла сети. |
| *Visible name* | Задайте видимое имя узла сети для создаваемого узла сети, используя:<br>**Host name** - техническое имя узла сети (по умолчанию)<br>**DNS name** - DNS-имя<br>**IP address** - IP-адрес<br>**<discovery check>** - полученное строковое значение проверки обнаружения (например, проверка агента Zabbix, SNMP agent)<br>См. также: Именование узла сети. |
| *Enabled* | Если флажок установлен, правило активно и будет выполняться сервером Zabbix.<br>Если флажок снят, правило не активно. Оно не будет выполняться. |

Превышение ограничения на количество дескрипторов файлов

При большом количестве одновременных проверок существует риск исчерпания лимита файловых дескрипторов для менеджера обнаружения (discovery manager).

Количество файловых дескрипторов, необходимых для обнаружения, равно количеству рабочих процессов обнаружения (discovery worker), умноженному на 1000. По умолчанию используется 5 рабочих процессов обнаружения, в то время как мягкий предел системы составляет приблизительно 1024.

При достижении этого лимита Zabbix уменьшит количество по умолчанию одновременных проверок каждого типа для каждого рабочих процессов обнаружения и запишет предупреждение в файл журнала. Однако, если пользователь установил значение *Максимум одновременных проверок по каждому типу (Maximum concurrent checks per type)* больше значения, рассчитанного Zabbix, Zabbix будет использовать заданное пользователем значение для одного рабочего процесса.

Сценарий из реальной жизни

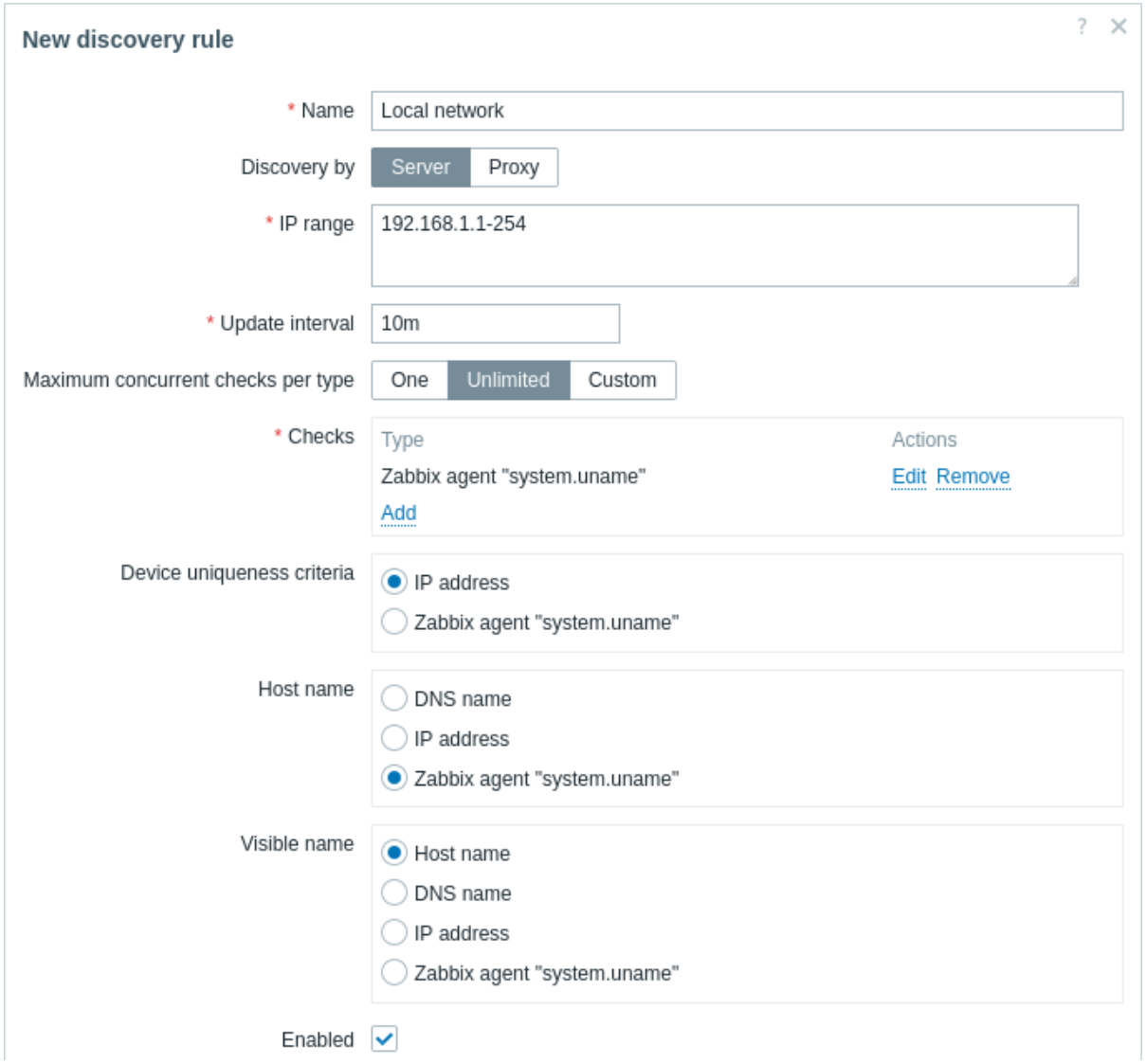
В этом примере мы бы хотели настроить сетевое обнаружение для локальной сети, имеющей диапазон IP адресов 192.168.1.1-192.168.1.254.

В нашем сценарии мы хотим:

- обнаруживать те узлы сети, на которых имеется Zabbix агент
- выполнять обнаружение каждые 10 минут
- добавлять узел сети для наблюдения, если время работы узла сети более 1 часа
- удалять узел сети, если он недоступен на протяжении более 24 часов
- добавлять узлы сети Linux в группу «Linux servers»
- добавлять узлы сети Windows в группу «Windows servers»
- использовать шаблон *Linux* для узлов сети Linux
- использовать шаблон *Windows* для узлов сети Windows

Шаг 1

Добавим правило сетевого обнаружения для нашего диапазона IP адресов.



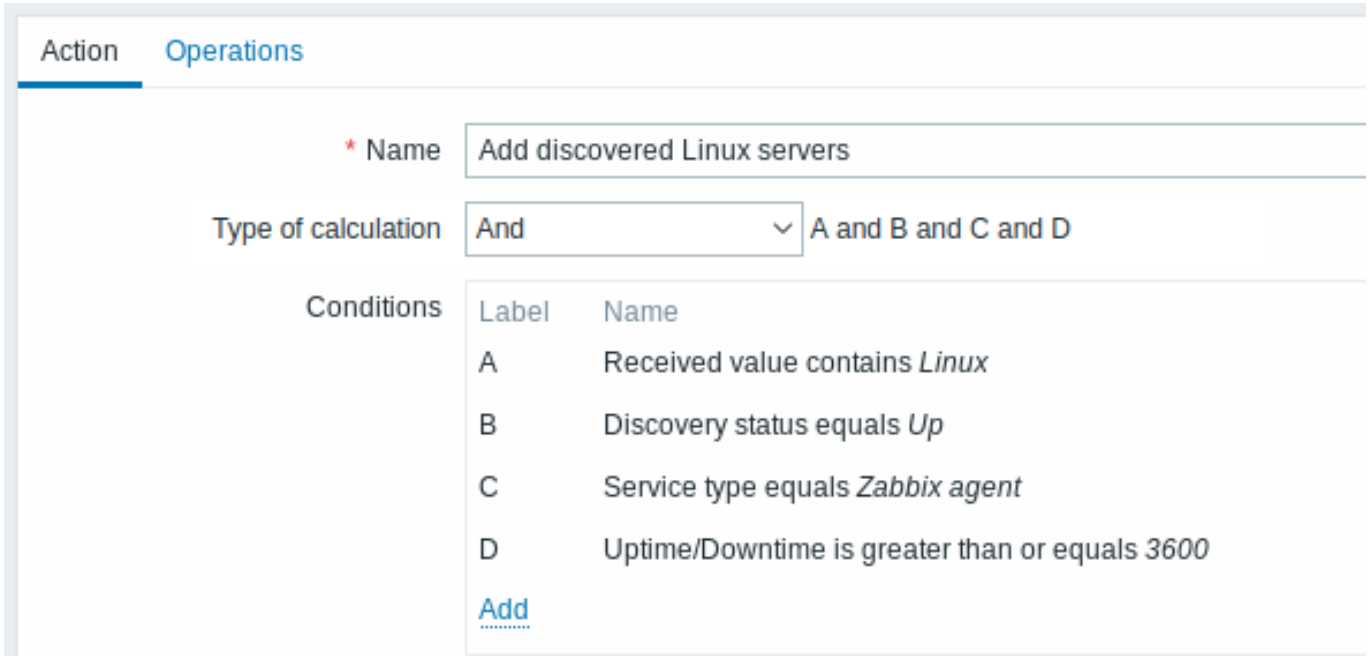

Zabbix будет пытаться обнаружить узлы сети в диапазоне IP адресов 192.168.1.1-192.168.1.254, подключаясь к Zabbix агенту и получая значение ключа **system.uname**. Полученное от агента значение может быть использовано для именования узла сети, а также для выполнения различных действий для разных операционных систем. Например, для присоединения шаблона *Windows* к серверам Windows, а шаблона *Linux* — к серверам Linux.

Правило будет выполняться каждые 10 минут.

Когда правило будет добавлено, Zabbix автоматически запустит обнаружение и порождение событий, основанных на обнаружении, для их дальнейшей обработки.

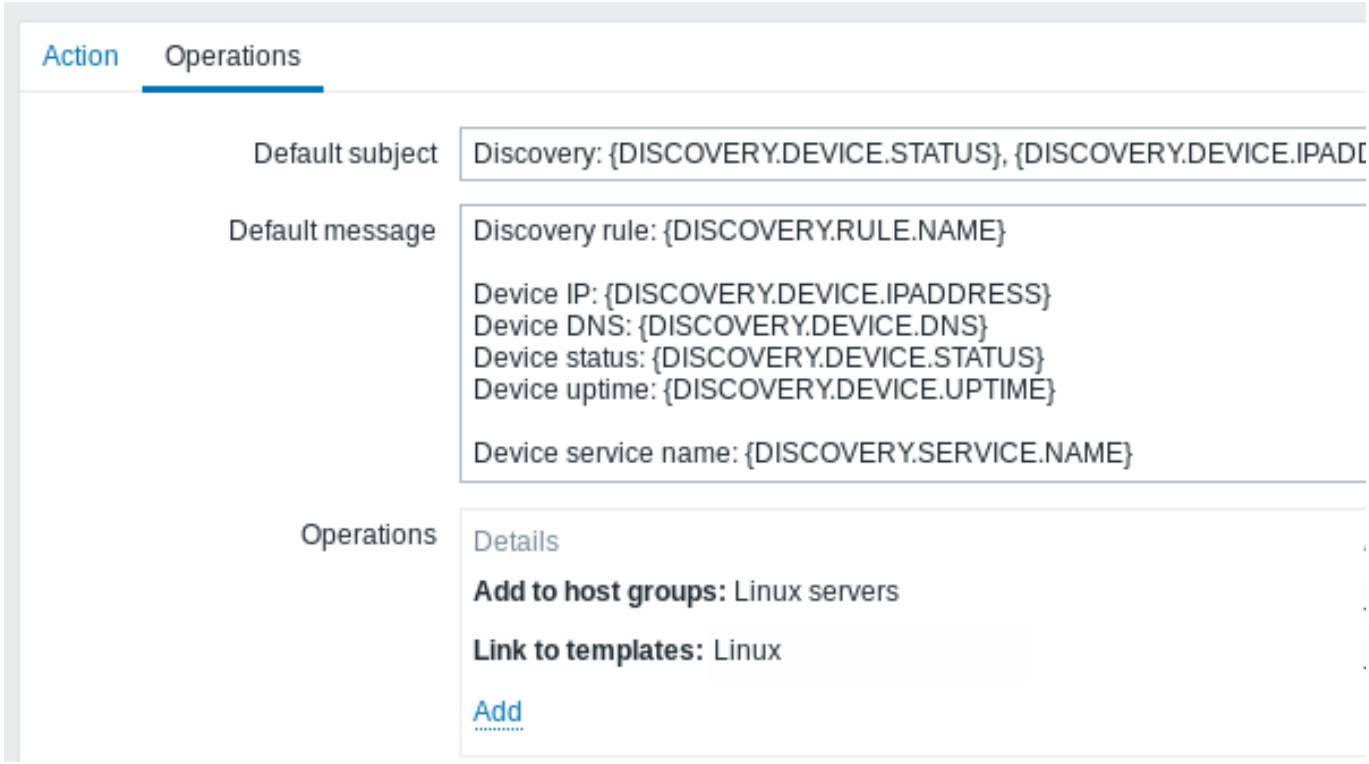
Шаг 2

Определим действие для добавления обнаруженных Linux серверов в соответвующие группы/шаблоны.




Это действие будет активировано, если:

- сервис «Zabbix agent» в состоянии «доступен»
- значение system.uname (ключ Zabbix агента, который мы использовали при создании правила) содержит «Linux»
- время работы — 1 час (3600 секунд) или больше

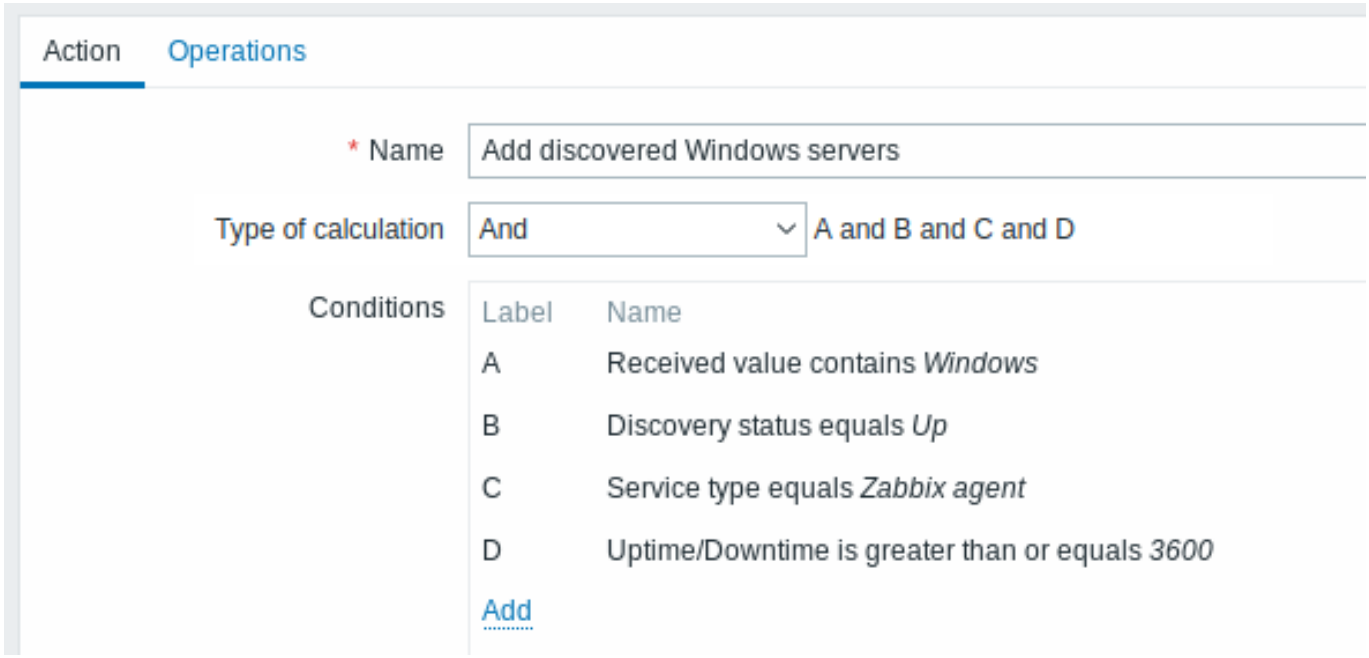


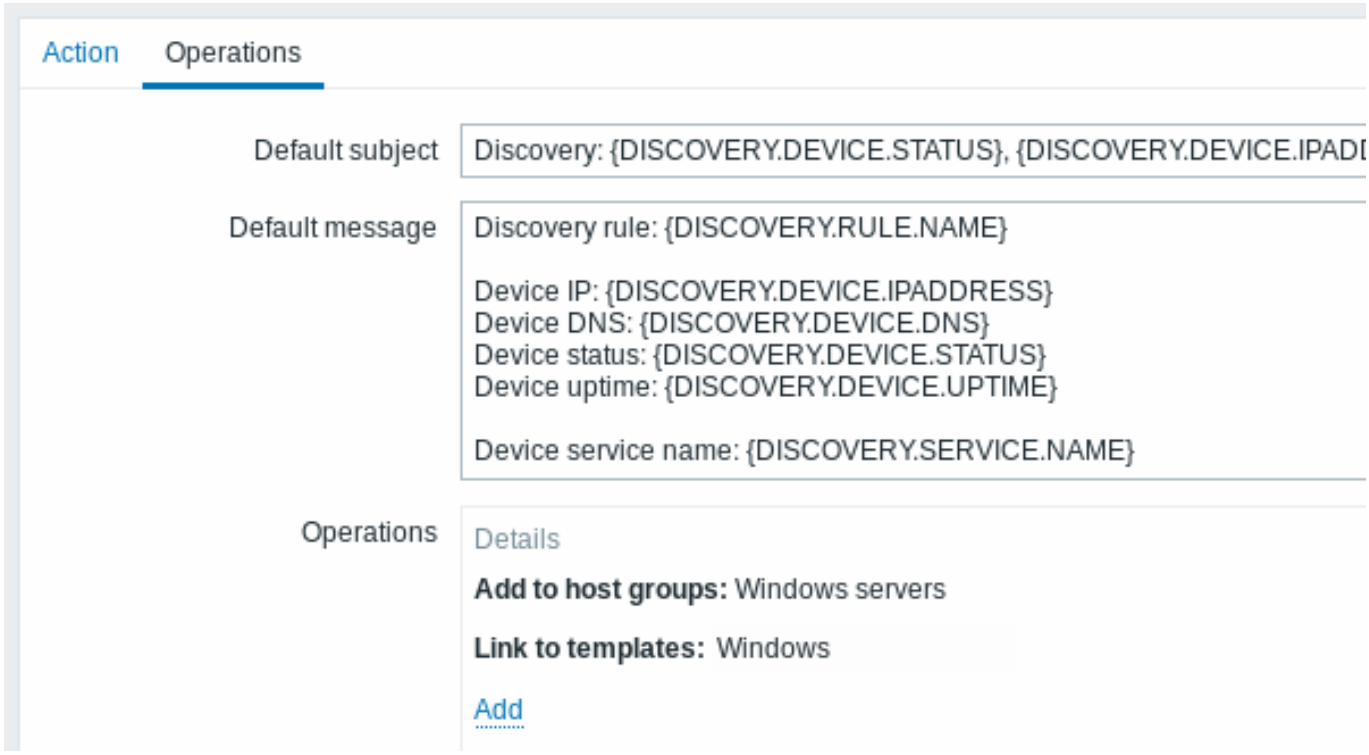

Это действие будет выполнять следующие операции:

- добавление обнаруженного узла сети в группу «Linux servers» (а также добавление узла сети, если он не был добавлен ранее)
- присоединение узла сети к шаблону *Linux*. Zabbix автоматически запустит наблюдение за узлом сети, используя элементы данных и триггеры из шаблона *Linux*.

Шаг 3

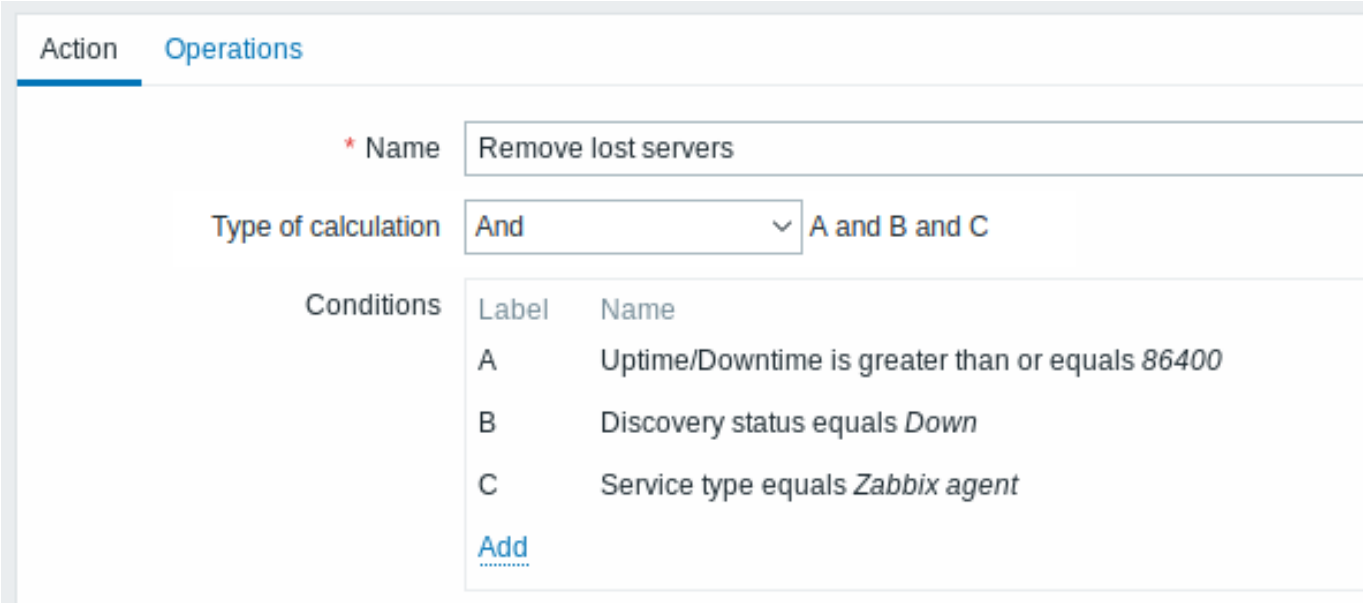
Зададим действие на обнаружение для добавления обнаруженных серверов Windows в соответствующие группы/шаблоны.

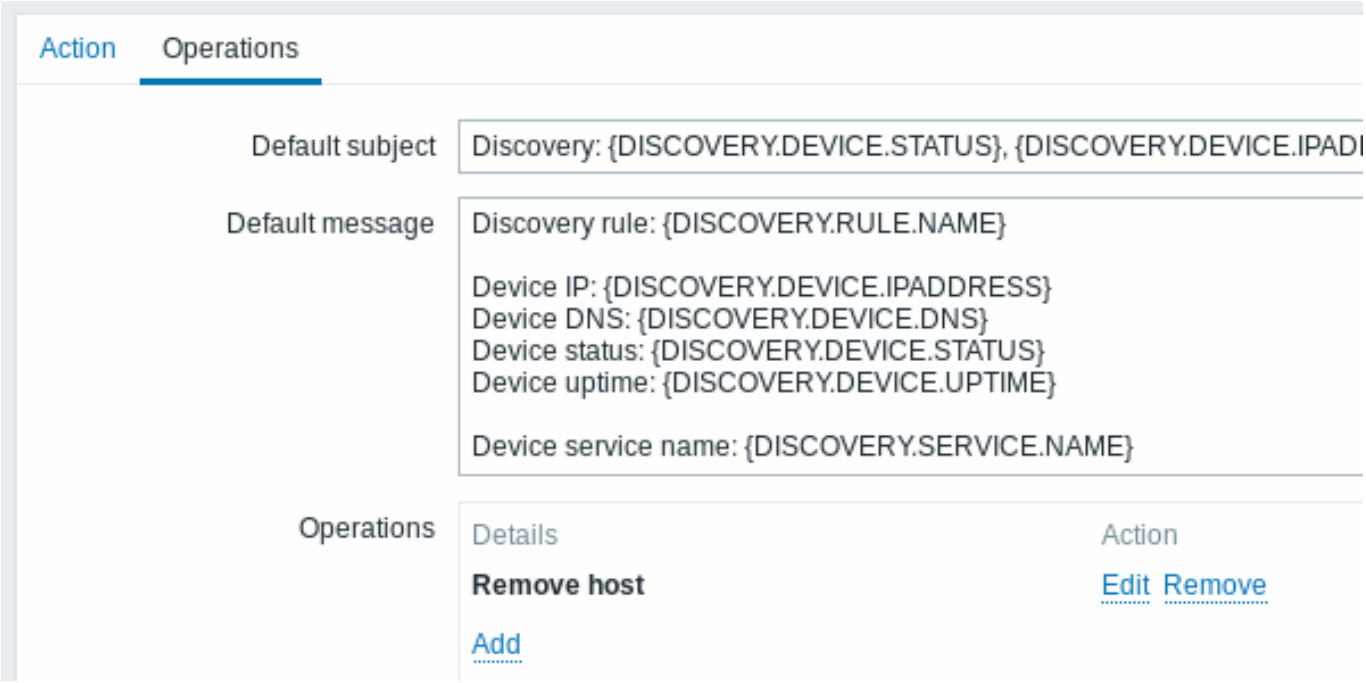



Шаг 4

Зададим действие на обнаружение для удаления потерянных серверов.




Сервер будет удалён, если сервис «Zabbix agent» в состоянии «Недоступен» на протяжении более чем 24 часов (86400 секунд).

**2. Авторегистрация активных агентов**

Обзор

Возможно разрешить авторегистрацию активных агентов Zabbix, после которой сервер может начать их отслеживать. Таким образом, новые узлы сети могут быть добавлены для мониторинга без их настройки на сервере вручную.

Авторегистрация может произойти, когда ранее неизвестный активный агент запрашивает проверки.

Эта функция может быть очень удобна для автоматического мониторинга новых облачных узлов. Как только у вас появится новый узел в облаке, Zabbix автоматически запустит сбор данных о производительности и доступности узла сети.

Авторегистрация активного агента также поддерживает мониторинг добавленных узлов сети с пассивными проверками. Когда активный агент запрашивает проверки, при условии, что параметры конфигурации `ListenIP` или `ListenPort` определенны в файле конфигурации, они отправляются на сервер. Если указано несколько IP-адресов, на сервер отправляется первый из них.

Сервер при добавлении нового автоматически зарегистрированного узла сети использует полученные IP адрес и порт

для настройки агента. Если значение IP-адреса не получено, используется тот, который использовался для входящего соединения. Если значение порта не получено, используется 10050.

Можно указать, что узел сети должен быть автоматически зарегистрирован с DNS-именем в качестве интерфейса агента по умолчанию.

Авторегистрация выполняется повторно:

- если изменяется информация метаданных узла сети:
  - из-за изменения HostMetadata и перезапуска агента
  - из-за изменения значения, возвращаемого HostMetadataItem
- для созданных вручную узлов сети с отсутствующими метаданными
- если узел сети изменён вручную для мониторинга другим Zabbix прокси
- если авторегистрация для того же узла сети приходит с нового Zabbix прокси

Период авторегистрации активного агента для Zabbix сервера и Zabbix прокси составляет 120 секунд. Таким образом, в случае удаления обнаруженного узла авторегистрация будет выполнена повторно через 120 секунд.

Настройка

Задайте сервер

Убедитесь, что у вас в файле конфигурации агента — `zabbix_agentd.conf` — указан Zabbix сервер:

```
ServerActive=10.0.0.1
```

Если вы не указали параметр *Hostname* в `zabbix_agentd.conf`, то в качестве имени узла сети будет использоваться системное имя узла сети, на котором находится агент. Например, системное имя узла сети в Linux можно получить, выполнив команду `hostname`.

Если параметр *Hostname* в файле конфигурации агента Zabbix задан как разделённый запятыми список, то узлы сети будут созданы для каждого из перечисленных имён.

Перезапустите агента после любых изменений, сделанных в файле конфигурации.

Действие для авторегистрации активного агента

Когда сервер получает запрос на авторегистрацию от агента, он вызывает действие. Для авторегистрации агента необходимо настроить действие с источником события "Autoregistration".

**Note:**
Для авторегистрации активных агентов настройка сетевого обнаружения не требуется.

В веб-интерфейсе Zabbix перейдите в *Alerts* → *Actions*, выберите *Autoregistration actions* и нажмите *Create action*:

- На вкладке Action задайте имя для вашего действия
- При необходимости укажите conditions. В условиях для имени узла сети/метаданных узла сети можно использовать поиск по подстроке или сопоставление с регулярным выражением. Если вы собираетесь использовать условие "Host metadata", см. следующий раздел.
- На вкладке Operations добавьте нужные операции, такие как - 'Add host', 'Add to host group' (например, *Discovered hosts*), 'Link templates' и т. д.

**Note:**
Если узлы сети, которые будут проходить авторегистрацию, вероятно, будут поддерживать только активный мониторинг (например, если они находятся за межсетевым экраном относительно вашего сервера Zabbix), возможно, вам стоит создать отдельный шаблон, например *Template_Linux-active*, для привязки.

Созданные узлы сети добавляются в группу *Discovered hosts* (по умолчанию, настраивается в *Administration* > *General* > *Other*). Если вы хотите, чтобы узлы сети добавлялись в другую группу, добавьте операцию *Remove from host group* (указав "Discovered hosts"), а также операцию *Add to host group* (указав другую группу узлов сети), поскольку узел сети должен принадлежать к группе узлов сети.

Безопасная авторегистрация

Безопасный вариант авторегистрации возможен путём настройки аутентификации на основе общего ключа (PSK-based) при использовании шифрованных соединений.

Уровень шифрования задаётся глобально в разделе *Администрирование* → *Общие* → *Авторегистрация* *(Administration →General → Autoregistration)*. Можно выбрать варианты без шифрования, шифрование TLS с использованием аутентификации PSK или оба варианта (так что какие-то узлы сети могут регистрироваться без шифрования, а какие-то — с шифрованием).

Аутентификация по общему ключу (PSK) проверяется сервером Zabbix до добавления узла сети. В случае успеха узел сети добавляется, а параметры *Подключения к узлу сети/Соединения с узла сети (Connections from/to host)* выставляются в только «PSK» с ключом и строкой идентификации PSK такими же, как в глобальных настройках авторегистрации.

**Attention:**
Чтобы обеспечить безопасность авторегистрации на инсталяциях, использующих прокси, должно быть включено шифрование между Zabbix сервером и прокси.

Использование DNS в качестве интерфейса по умолчанию

Параметры конфигурации HostInterface и HostInterfaceItem позволяют указать пользовательское значение для интерфейса узла сети во время авторегистрации.

Более конкретно, они полезны, если узел сети должен быть автоматически зарегистрирован с DNS-именем в качестве интерфейса агента по умолчанию, а не с его IP-адресом. В этом случае DNS-имя должно быть указано или возвращено как значение параметра HostInterface или HostInterfaceItem. При изменении значения одного из двух параметров — например, с IP адреса на DNS имя или наоборот — интерфейс по умолчанию автоматически зарегистрированного узла сети будет обновлён соответственно. Это обновление применяется к существующему узлу сети, без создания нового. Для того, чтобы отослать новое значение, агент необходимо перезапустить, чтобы он заново инициализировал процесс авторегистрации.

**Note:**
Если параметры HostInterface или HostInterfaceItem не настроены, вместо них будет использоваться параметр listen_dns. Это значение определяется путём выполнения обратного DNS-запроса IP адреса агента. Если обратный DNS не настроен правильно или возвращает недопустимое имя, это может привести к неправильной или неудачной авторегистрации из-за недопустимого значения интерфейса.

Использование метаданных узла сети

Когда агент отправляет запрос на авторегистрацию на сервер, он отправляет свое имя хоста (hostname). В некоторых случаях (например, облачные узлы Amazon) имени хоста недостаточно, чтобы сервер Zabbix различал обнаруженные узлы сети. Для отправки другой информации от агента на сервер могут быть дополнительно использованы метаданные узла сети.

Метаданные узла настраиваются в файле конфигурации агента — `zabbix_agentd.conf`. Существует 2 способа указать метаданные узла сети в файле конфигурации:

```
HostMetadata
HostMetadataItem
```

Смотрите описание опций по ссылке выше.

Параметр HostMetadataItem может возвращать до 65535 символов Юникода UTF-8. Более длинное значение будет усечено.

Обратите внимание, что в MySQL эффективная максимальная длина в символах будет меньше, если возвращаемое значение содержит многобайтовые символы. Например, значение, содержащее только 3-байтовые символы, будет ограничено 21844 символами, а значение, содержащее только 4-байтовые символы, — 16383 символами.

**Attention:**
Попытка авторегистрации происходит каждый раз, когда активный агент отправляет на сервер запрос на обновление активных проверок. Задержка между запросами указана в параметре агента RefreshActiveChecks. Первый запрос отправляется сразу после перезапуска агента.

Примеры

Авторегистрация по операционной системе с использованием HostMetadata

Допустим, вы хотите, чтобы узлы сети автоматически регистрировались сервером Zabbix. В вашей сети у вас есть активные агенты Zabbix (см. раздел «Настройка» выше). Также в вашей сети есть хосты Windows и хосты Linux, а в вашем веб-интерфейсе Zabbix вам доступны шаблоны «Linux by Zabbix agent» и «Windows by Zabbix agent». Поэтому вы хотите, чтобы при регистрации узла сети к регистрируемому узлу был применён соответствующий шаблон Linux/Windows. По умолчанию, на сервер при авторегистрации отправляется только имя узла сети, чего может быть недостаточно. Чтобы убедиться, что к узлу сети применяется правильный шаблон, вы должны использовать метаданные узла сети.

**Настройка в веб-интерфейсе**

Первое, что надо сделать, — это настроить веб-интерфейс. Создайте 2 действия. Первое действие:

- Имя: Авторегистрация узлов сети Linux
- Условия: Метаданные узлов сети содержит *Linux*
- Операции: Присоединить шаблоны: Linux by Zabbix agent

> **Note:**
> В данном случае можно опустить операцию «Добавить узел сети». Присоединение шаблона к узлу сети требует, чтобы узел сети уже был добавлен, поэтому сервер сделает это автоматически.

Второе действие:

- Имя: Авторегистрация узлов сети Windows
- Условия: Метаданные узлов сети содержит *Windows*
- Операции: Присоединить шаблоны: Windows by Zabbix agent

**Настройка агента**

Теперь вам нужно настроить агентов. В файлы конфигурации агентов добавьте следующую строку:

```
HostMetadataItem=system.uname
```

Таким способом вы обеспечите, что метаданные узлов сети будут содержать строку «Linux» или «Windows» в зависимости от узла сети, на котором работает агент. Пример метаданных узла сети в этом случае:

```
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
```

Не забудьте перезапустить агента после внесения любых изменений в его файл конфигурации.

Использование HostMetadata для управления авторегистрацией и шаблонами

***Шаг 1 — Защита регистрации с помощью HostMetadata***

Использование метаданных узла для обеспечения некоторой базовой защиты от регистрации нежелательных узлов сети.

**Настройка в веб-интерфейсе**

Создайте действие в веб-интерфейсе, используя некоторый трудно-угадываемый секретный код для запрета нежелательных узлов сети:

- Имя: Действие авторегистрации Linux
- Условия:
  - Тип вычисления: И
  - Условие (A): Метаданные узлов сети содержат //Linux//
  - Условие (B): Метаданные узлов сети содержат //21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae/
- Операции:
  - Отправить сообщение пользователям: Admin через все способы оповещения
  - Добавить в группы узлов сети: Linux сервера
  - Присоединить к шаблонам: Linux by Zabbix agent

Пожалуйста, обратите внимание, что использование этого метода в одиночку не предоставляет надёжную защиту, потому что данные передаются как простой текст. Для того, чтобы изменения вступили в силу немедленно, необходима перезагрузка кэша конфигурации.

**Настройка агента**

В файл конфигурации агента добавьте следующую строку:

```
HostMetadata=Linux    21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

где «Linux» — это платформа, а остаток строки — трудноугадываемый секретный текст.

Не забудьте перезапустить агента после внесения любых изменений в файл конфигурации.

***Step 2 — Добавление шаблона к зарегистрированному узлу сети***

К уже зарегистрированному узлу сети можно добавить дополнительные шаблоны. В данном случае шаблон MySQL by Zabbix agent будет присоединён только к узлам сети, у которых HostMetadata содержит токен `MySQL`.

**Настройка в веб-интерфейсе**

В веб-интерфейсе обновите действие:

- Имя: Действие авторегистрации Linux
- Условия:

  - Тип вычисления: И
  - Условие (A): Метаданные узлов сети содержат Linux
  - Условие (B): Метаданные узлов сети содержат 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
  - Условие (C): Метаданные узлов сети содержат MySQL
- Операции:
  - Отправить сообщение пользователям: Admin через все способы оповещения
  - Добавить в группы узлов сети: Linux сервера
  - Присоединить шаблоны: Linux by Zabbix agent
  - Присоединить шаблоны: MySQL by Zabbix Agent

**Настройка агента**

Обновите следующую строчку в конфигурационном файле агента:

```
HostMetadata=MySQL on Linux 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

Не забудьте перезапустить агента после внесения любых изменений в конфигурационный файл.

### 3. Низкоуровневое обнаружение (LLD)

**Обзор** Низкоуровневое обнаружение (LLD) предоставляет способ автоматически создавать элементы данных, триггеры и графики для различных сущностей на узле сети. Например, Zabbix может автоматически начать мониторинг файловых систем или сетевых интерфейсов на вашем компьютере без необходимости вручную создавать элементы данных для каждой файловой системы или сетевого интерфейса. LLD также может создавать узлы сети, например, для заполнения виртуальных машин, обнаруженных на гипервизоре, и вложенных правил обнаружения, что позволяет выполнять многоуровневое обнаружение. Кроме того, можно настроить Zabbix так, чтобы он автоматически удалял ненужные сущности на основе фактических результатов периодически выполняемого обнаружения.

Пользователь может определить собственные типы обнаружения при условии, что они следуют определенному JSON-протоколу.

Общая архитектура процесса обнаружения выглядит следующим образом.

Сначала пользователь создает правило обнаружения в *Сбор данных* > *Шаблоны*, в столбце *Обнаружение*. Правило обнаружения состоит из (1) элемента данных, который обнаруживает необходимые сущности (например, файловые системы или сетевые интерфейсы), и (2) прототипов элементов данных, триггеров и графиков, которые должны быть созданы на основе значения этого элемента данных.

Элемент данных, который обнаруживает необходимые сущности, похож на обычный элемент данных, встречающийся в других местах: сервер запрашивает у агента Zabbix (или у того типа, который задан для элемента данных) значение этого элемента данных, а агент отвечает текстовым значением. Разница в том, что значение, которое возвращает агент, должно содержать список обнаруженных сущностей в формате JSON. Хотя детали этого формата важны только для разработчиков пользовательских проверок обнаружения, необходимо знать, что возвращаемое значение содержит список пар макрос → значение. Например, элемент данных ”net.if.discovery” может вернуть две пары: ”{#IFNAME}” → ”lo” и ”{#IFNAME}” → ”eth0”.

Эти макросы используются в именах, ключах и других полях прототипов, где затем подставляются полученные значения для создания реальных элементов данных, триггеров, графиков или даже узлов сети для каждой обнаруженной сущности. См. полный список параметров для использования макросов LLD.

Когда сервер получает значение для элемента данных обнаружения, он анализирует пары макрос → значение и для каждой пары создает реальные элементы данных, триггеры и графики на основе их прототипов. В приведенном выше примере с ”net.if.discovery” сервер создал бы один набор элементов данных, триггеров и графиков для петлевого интерфейса ”lo” и еще один набор для интерфейса ”eth0”.

Обратите внимание, что начиная с **Zabbix 4.2** формат JSON, возвращаемого правилами низкоуровневого обнаружения, был изменен. Больше не ожидается, что JSON будет содержать объект ”data”. Теперь низкоуровневое обнаружение принимает обычный JSON, содержащий массив, чтобы поддерживать новые возможности, такие как предварительная обработка значения элемента данных и пользовательские пути к значениям макросов низкоуровневого обнаружения в JSON-документе.

Встроенные ключи обнаружения были обновлены так, чтобы возвращать массив строк LLD в корне JSON-документа. Zabbix автоматически извлечет макрос и значение, если поле массива использует синтаксис {#MACRO} в качестве ключа. Все новые встроенные проверки обнаружения будут использовать новый синтаксис без элементов ”data”. При обработке значения низкоуровневого обнаружения сначала определяется корень (массив в `$.` или `$.data`).

Хотя элемент ”data” был удален из всех встроенных элементов данных, связанных с обнаружением, для обратной совместимости Zabbix по-прежнему будет принимать JSON-нотацию с элементом ”data”, хотя ее использование не рекомендуется. Если JSON содержит объект только с одним элементом массива ”data”, то содержимое этого элемента будет автоматически извлечено с использованием JSONPath `$.data`. Низкоуровневое обнаружение теперь принимает необязательные пользовательские макросы LLD с пользовательским путем, указанным в синтаксисе JSONPath.

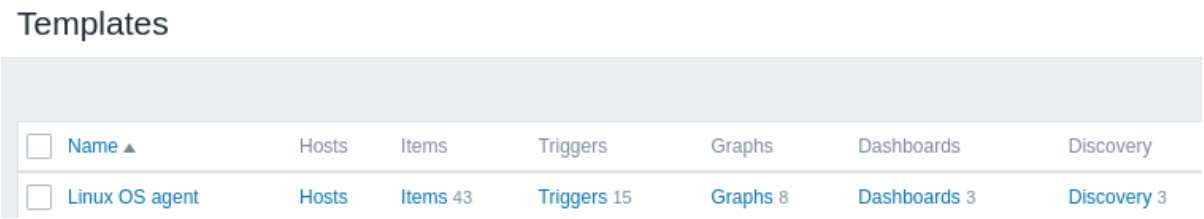
**Warning:**
В результате указанных выше изменений более новые агенты больше не смогут работать со старым сервером Zabbix.

См. также: Обнаруженные сущности

**Настройка низкоуровневого обнаружения** Мы продемонстрируем низкоуровневое обнаружение на примере обнаружения файловой системы.

Чтобы настроить обнаружение, выполните следующие действия:

- Перейдите в: *Сбор данных* > *Шаблоны* или *Узлы сети*.
- Нажмите *Обнаружение* в строке соответствующего шаблона/узла сети.



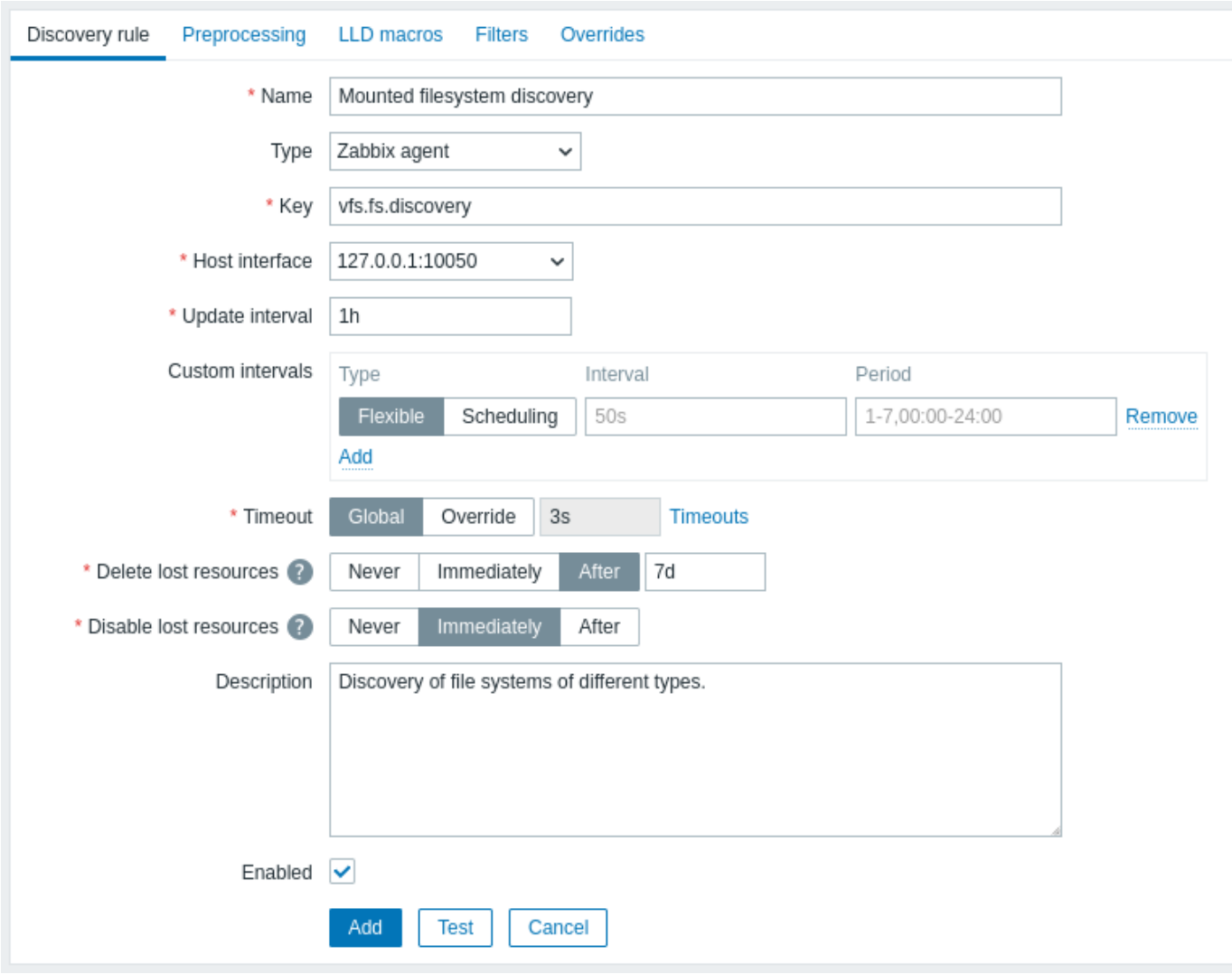

- Нажмите *Создать правило обнаружения* в правом верхнем углу экрана.
- Заполните форму правила обнаружения, указав необходимые сведения.

Правило обнаружения

Диалог правила обнаружения содержит пять вкладок, представляющих (слева направо) поток обработки данных во время обнаружения:

- *Правило обнаружения (Discovery rule)* — самое главное, определяет встроенный элемент данных или пользовательский сценарий для получения данных обнаружения.
- *Предобработка (Preprocessing)* — применяет какую-либо предобработку к данным обнаружения.
- *LLD макросы (LLD macros)* — позволяет извлечь значения некоторых макросов, чтобы использовать в обнаруженных элементах данных, триггерах и т.д.
- *Фильтры (Filters)* — позволяет отфильтровать обнаруженные значения.
- *Замещения (Overrides)* — позволяет изменить прототипы элементов данных, триггеров, графиков или узлов сети, когда они применяются к конкретным найденным объектам.

Вкладка **Правило обнаружения (Discovery rule)** содержит ключ элемента данных, используемого для обнаружения (а также некоторые общие атрибуты правила обнаружения):




Все обязательные поля ввода отмечены красной звёздочкой.

| Параметр | Описание |
|---|---|
| *Имя (Name)* | Имя правила обнаружения. |
| *Тип (Type)* | Тип проверки выполняемого обнаружения.<br>В данном примере мы используем тип *Zabbix агент*.<br>Правило обнаружения также может являться зависимым элементом данных, зависящим от обычного элемента данных. Оно не может зависеть от другого правила обнаружения. Для зависимых элементов данных выберите соответствующий тип (*Зависимый элемент данных (Dependent item)*) и укажите основной элемент данных в поле «Основной элемент данных (Master item)». Основной элемент данных должен существовать. |
| *Ключ (Key)* | Введите ключ элемента данных, используемого для обнаружения (до 2048 символов).<br>Например, вы можете использовать встроенный ключ элемента данных «vfs.fs.discovery», который возвращает строку JSON со списком файловых систем, присутствующих в компьютере, их типов и опций монтирования.<br>Обратите внимание, что другой вариант обраружения файловых систем — это использовать результаты, возвращаемые ключом агента «vfs.fs.get» (смотрите пример). |

| Параметр | Описание |
|---|---|
| *Интервал обновления (Update interval)* | Это поле задаёт, как часто Zabbix выполняет обнаружение. Вначале, когда вы только настраиваете обнаружение файловых систем, вы можете указать маленький интервал; но как только вы удостоверитесь что всё работает, вы можете установить его в 30 минут или более, потому что обычно файловые системы не меняются очень часто.<br>Поддерживаются суффиксы времени, например: 30s, 1m, 2h, 1d.<br>Поддерживаются пользовательские макросы.<br>*Обратите внимание*: интервал обновления может быть выставлен в «0» только если существует пользовательский интервал с ненулевым значением. Если укажете значение, равное «0», и пользовательский интервал (переменный или по расписанию) с ненулевым значением существует, элемент данных будет опрашиваться в течение действия переменного интервала.<br>Новые правила обнаружения будут проверяться в течение 60 секунд с момента их создания, если только для них не заданы интервал обновления по расписанию или переменный интервал обновления, а *Интервал обновления (Update interval)* не установлен в 0.<br>*Обратите внимание*, что уже созданное правило обнаружения можно выполнить незамедлительно нажатием кнопки *Выполнить сейчас*. |
| *Пользовательские интервалы (Custom intervals)* | Вы можете создавать пользовательские правила проверки элемента данных:<br>**Гибкий (Flexible)** — создание исключений из *Интервала обновления* (интервал с другой частотой обновления)<br>**По расписанию (Scheduling)** — создание пользовательского расписания проверки.<br>Для получения более подробной информации смотрите Пользовательские интервалы. |
| *Время ожидания (Timeout)* | Укажите тайм-аут проверки обнаружения. Выберите вариант тайм-аута:<br>**Глобальный (Global)** — используется глобальный тайм-аут или тайм-аут прокси (отображается в поле *Время ожидания* на сером фоне);<br>**Замещения (Override)** — применяется пользовательский тайм-аут (заданный в поле *Время ожидания*; допустимый диапазон: 1 - 600s). Поддерживаются суффиксы времени, например: 30s, 1m; также поддерживаются пользовательские макросы.<br>Щелчок по ссылке *Время ожидания (Timeouts)* позволяет настроить тайм-ауты прокси либо глобальные тайм-ауты (если прокси не используется). Обратите внимание, что ссылка *Время ожидания (Timeouts)* видна только пользователям с типом *Супер-администратор*, имеющим доступ к разделам веб-интерфейса *Администрирование (Administration)* → *Общие (General)* или *Администрирование (Administration)* → *Прокси (Proxies)*. |
| *Удаление потерянных ресурсов (Delete lost resources)* | Укажите, через какое время обнаруженный объект будет удалён, после того как его состояние обнаружения станет «более не обнаруживается (Not discovered anymore)»:<br>*Никогда (Never)* — не будет удалён;<br>*Незамедлительно (Immediately)* — будет удалён сразу же;<br>*После (After)* — будет удалён по истечении указанного периода времени. Значение должно быть больше, чем значение в поле *Деактивация потерянных ресурсов (Disable lost resources)*.<br>Поддерживаются суффиксы времени, например: 2h, 1d.<br>Поддерживаются пользовательские макросы.<br>*Обратите внимание*: Использование значения «Незамедлительно (Immediately)» не рекомендуется, так как простое ошибочное изменение фильтра может закончиться тем, что объект будет удалён вместе со всеми данными истории.<br>Обратите внимание, что ресурсы, деактивированные вручную, правилом низкоуровневого обнаружения удалены не будут. |
| *Деактивация потерянных ресурсов (Disable lost resources)* | Укажите, через какое время обнаруженный объект будет деактивирован, после того как его состояние обнаружения станет «более не обнаруживается (Not discovered anymore)»:<br>*Никогда (Never)* — не будет деактивирован;<br>*Незамедлительно (Immediately)* — будет деактивирован сразу же;<br>*После (After)* — будет деактивирован по истечении указанного периода времени. Значение должно быть больше, чем интервал обновления правила обнаружения.<br>Обратите внимание, что автоматически деактивированные ресурсы будут активированы снова, если будут заново обнаружены низкоуровневым обнаружением. Ресурсы, деактивированные вручную, не станут активированы снова при повторном обнаружении.<br>Это поле не отображается, если параметр *Удаление потерянных ресурсов (Delete lost resources)* задан в «Незамедлительно (Immediately)».<br>Поддерживаются суффиксы времени, например: 2h, 1d.<br>Поддерживаются пользовательские макросы. |
| *Описание (Description)* | Введите описание. |

| Параметр | Описание |
|---|---|
| *Активировано (Enabled)* | Если отмечено, правило будет обрабатываться. |

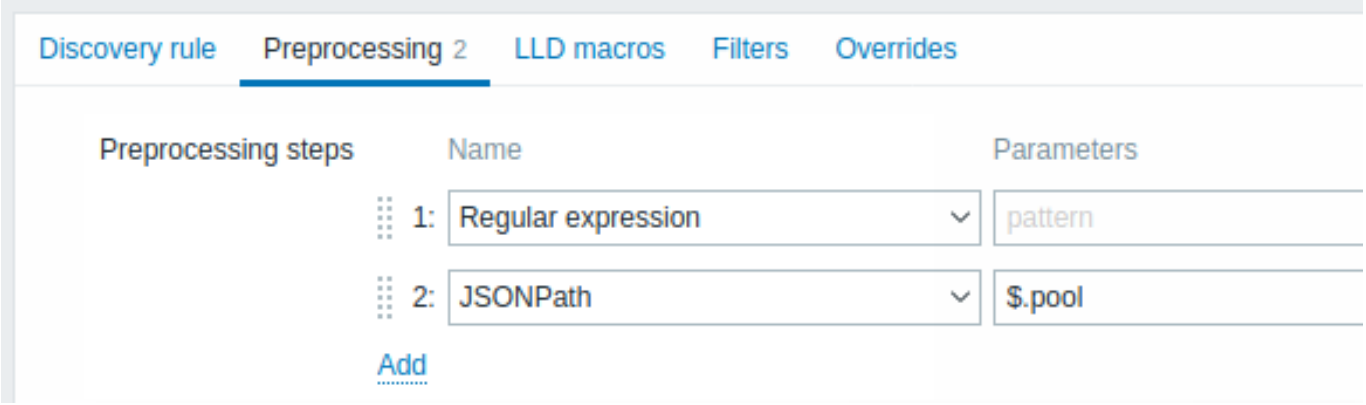
**Note:**
История правил обнаружения не сохраняется.

Предобработка

Вкладка **Предобработка** позволяет определить правила преобразования, применяемые к результату обнаружения. На этом этапе возможны одно или несколько преобразований. Преобразования выполняются в том порядке, в котором они определены. Вся предобработка выполняется сервером Zabbix.

См. также:

- Подробности предобработки
- Тестирование предобработки




| Type | | |
|---|---|---|
| | *Transformation* | Description |
| Text | | |
| | *Regular expression* | Сопоставьте полученное значение с регулярным выражением <pattern> и замените значение извлеченным <output>. Регулярное выражение поддерживает извлечение максимум 10 захваченных групп с помощью последовательности \N.<br>Параметры:<br>**pattern** - регулярное выражение<br>**output** - шаблон форматирования вывода. Экранирующая последовательность \N (где N=1…9) заменяется на N-ю совпавшую группу. Экранирующая последовательность \0 заменяется на совпавший текст.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| | *Replace* | Найдите строку поиска и замените ее другой строкой (или ничем). Все вхождения строки поиска будут заменены.<br>Параметры:<br>**search string** - строка для поиска и замены, с учетом регистра (обязательно)<br>**replacement** - строка, которой нужно заменить строку поиска. Строка замены также может быть пустой, что фактически позволяет удалить строку поиска при ее нахождении.<br>Можно использовать escape-последовательности для поиска или замены разрывов строк, возврата каретки, табуляции и пробелов ”\n \r \t \s”; обратную косую черту можно экранировать как ”\\” и escape-последовательности можно экранировать как ”\\n”. Экранирование разрывов строк, возврата каретки и табуляции автоматически выполняется во время обнаружения на низком уровне. |
| Structured data | | |

| Type | |
|---|---|
| *JSONPath* | Извлеките значение или фрагмент из данных JSON с помощью функциональности JSONPath.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *XML XPath* | Извлеките значение или фрагмент из данных XML с помощью функциональности XPath.<br>Чтобы эта опция работала, сервер Zabbix должен быть собран с поддержкой libxml.<br>Примеры:<br>`number(/document/item/value)` извлечет 10 из `<document><item><value>10</value></item></document>`<br>`number(/document/item/@attribute)` извлечет 10 из `<document><item attribute="10"></item></document>`<br>`/document/item` извлечет `<item><value>10</value></item>` из `<document><item><value>10</value></item></document>`<br>Обратите внимание, что пространства имен не поддерживаются.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *CSV to JSON* | Преобразуйте данные CSV-файла в формат JSON.<br>Дополнительные сведения см. в разделе: Предобработка CSV to JSON. |
| *XML to JSON* | Преобразуйте данные в формате XML в JSON.<br>Дополнительные сведения см. в разделе: Правила сериализации.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| SNMP | |
| *SNMP walk value* | Извлеките значение по указанному имени OID/MIB и примените параметры форматирования:<br>**Unchanged** - вернуть Hex-STRING как неэкранированную шестнадцатеричную строку (*note*: при этом по-прежнему применяются подсказки отображения);<br>**UTF-8 from Hex-STRING** - преобразовать Hex-STRING в строку UTF-8;<br>**MAC from Hex-STRING** - преобразовать Hex-STRING в строку MAC-адреса (при этом `' '` будет заменено на `':'`);<br>**Integer from BITS** - преобразовать первые 8 байт битовой строки, представленной последовательностью шестнадцатеричных символов (например, ”1A 2B 3C 4D”), в 64-битное беззнаковое целое. В битовых строках длиннее 8 байт последующие байты будут проигнорированы.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *SNMP walk to JSON* | Преобразуйте значения SNMP в JSON. Укажите имя поля в JSON и соответствующий путь SNMP OID. Значения полей будут заполнены значениями по указанному пути SNMP OID. Этот шаг предобработки можно использовать для обнаружения SNMP OID.<br>Доступны аналогичные параметры форматирования значений, как и в шаге *SNMP walk value*.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *SNMP get value* | Примените параметры форматирования к значению SNMP get:<br>**UTF-8 from Hex-STRING** - преобразовать Hex-STRING в строку UTF-8;<br>**MAC from Hex-STRING** - преобразовать Hex-STRING в строку MAC-адреса (при этом `' '` будет заменено на `':'`);<br>**Integer from BITS** - преобразовать первые 8 байт битовой строки, представленной последовательностью шестнадцатеричных символов (например, ”1A 2B 3C 4D”), в 64-битное беззнаковое целое. В битовых строках длиннее 8 байт последующие байты будут проигнорированы.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| Custom scripts | |

| | |
|---|---|
| Type | |
| *JavaScript* | Введите код JavaScript в модальном редакторе, который открывается при щелчке в поле параметра или по значку карандаша рядом с ним.<br>Обратите внимание, что доступная длина JavaScript зависит от используемой базы данных.<br>Дополнительные сведения см. в разделе: Предобработка Javascript |
| Validation | |
| *Does not match regular expression* | Укажите регулярное выражение, которому значение не должно соответствовать.<br>Например, `Error:(.*?)\.`<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *Check for error in JSON* | Проверьте наличие сообщения об ошибке уровня приложения, расположенного по JSONPath. Остановите обработку, если проверка выполнена успешно и сообщение не пустое; в противном случае продолжите обработку со значением, которое было до этого шага предобработки. Обратите внимание, что ошибки внешнего сервиса сообщаются пользователю как есть, без добавления информации о шаге предобработки.<br>Например, `$.errors`. Если получен JSON вида `{"errors":"e1"}`, следующий шаг предобработки выполняться не будет.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *Check for error in XML* | Проверьте наличие сообщения об ошибке уровня приложения, расположенного по Xpath. Остановите обработку, если проверка выполнена успешно и сообщение не пустое; в противном случае продолжите обработку со значением, которое было до этого шага предобработки. Обратите внимание, что ошибки внешнего сервиса сообщаются пользователю как есть, без добавления информации о шаге предобработки.<br>Ошибка не будет сообщена в случае неудачного разбора некорректного XML.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| *Matches regular expression* | Укажите регулярное выражение, которому значение должно соответствовать.<br>Если установить флажок *Custom on fail*, можно указать пользовательские параметры обработки ошибки: либо отбросить значение, либо задать указанное значение, либо задать указанное сообщение об ошибке. |
| Throttling | |
| *Discard unchanged with heartbeat* | Отбрасывайте значение, если оно не изменилось в течение заданного периода времени (в секундах).<br>Поддерживаются положительные целые значения секунд (минимум - 1 секунда). В этом поле можно использовать суффиксы времени (например, 30s, 1m, 2h, 1d). В этом поле можно использовать пользовательские макросы и макросы обнаружения на низком уровне.<br>Для элемента обнаружения можно указать только один параметр ограничения частоты.<br>Например, `1m`. Если одинаковый текст будет передан в это правило дважды в течение 60 секунд, он будет отброшен.<br>*Note*: Изменение прототипов элементов не сбрасывает ограничение частоты. Ограничение частоты сбрасывается только при изменении шагов предобработки. |
| Prometheus | |
| *Prometheus to JSON* | Преобразуйте требуемые метрики Prometheus в JSON.<br>Подробнее см. в разделе Проверки Prometheus. |

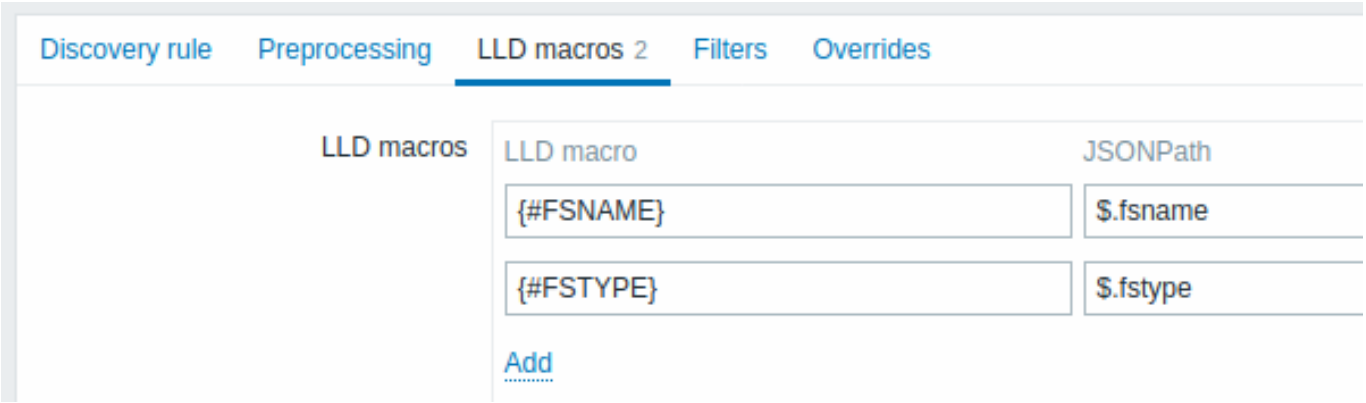
Обратите внимание: если правило обнаружения было применено к узлу сети через шаблон, содержимое этой вкладки доступно только для чтения.

Настраиваемые макросы

Вкладка **LLD макросы (LLD macros)** позволяет указать настраиваемые макросы низкоуровневого обнаружения.

Настраиваемые макросы полезны в случаях, когда возвращаемый JSON не имеет нужных макросов, уже определённых должным образом. Так, например:

- Встроенный ключ `vfs.fs.discovery` для обнаружения файловых систем возвращает JSON с некоторыми уже определёнными LLD макросами, такими как {#FSNAME} и {#FSTYPE}. Эти макросы могут быть непосредственно использованы в прототипах элементов данных и триггеров (смотрите последующие разделы на этой странице); определение настраиваемых макросов не требуется;
- Элемент данных агента `vfs.fs.get` также возвращает JSON с данными о файловой системе, однако, без готовых LLD макросов. В этом случае вы можете определить эти макросы самостоятельно, и выставить соответствие между ними и значениями в JSON-е, используя JSONPath:

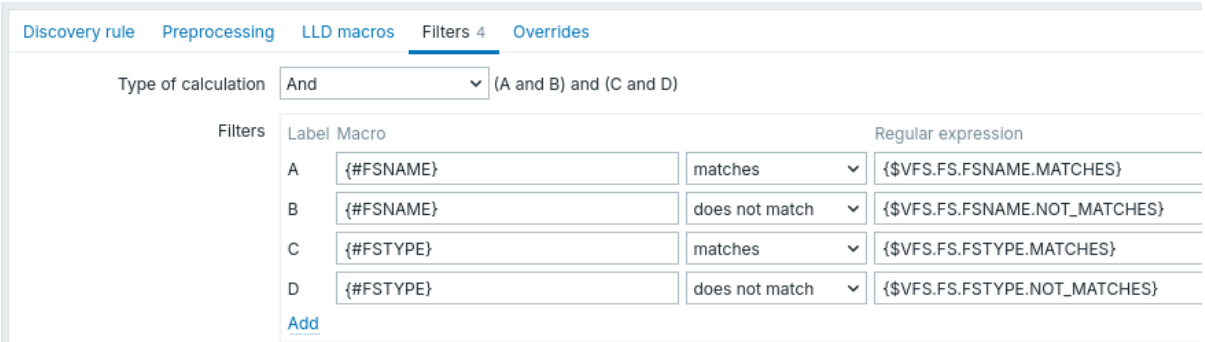



Извлечённые значения могут использоваться в обнаруженных элементах данных, триггерах и т.п. Обратите внимание, что значения будут извлекаться из результата обнаружения после выполнения шагов предобработки.

| Параметр | Описание |
|---|---|
| *LLD макрос (LLD macro)* | Имя макроса низкоуровневого обнаружения, с использованием следующего синтаксиса: {#MACRO}. |
| *JSONPath* | Путь, используемый для извлечения значения LLD макроса из строки LLD, используя синтаксис JSONPath.<br>Значения, извлекаемые из возвращаемого JSON-а, используются для замены LLD макросов в полях прототипов элемента данных, триггера и т.п.<br>JSONPath может быть указан, используя нотацию с точкой или с квадратными скобками. Нотация с квадратными скобками должна использоваться в случаях использования любых специальных символов и символов Unicode, наподобие следующего: `$['unicode + special chars #1']['unicode + special chars #2']`.<br><br>Например, `$.foo` извлечёт «bar» и «baz» из такого JSON-а: `[{"foo":"bar"}, {"foo":"baz"}]`<br>Обратите внимание, что `$.foo` также извлечёт «bar» и «baz» из такого JSON-а: `{"data":[{"foo":"bar"}, {"foo":"baz"}]}` из-за того, что одиночный объект "data" обрабатывается автоматически (для обратной совместимости с реализацией низкоуровневого обнаружения в версиях Zabbix до 4.2). |

Фильтр

Фильтры могут использоваться для того, чтобы генерировать реальные элементы данных, триггеры и графики только для объектов, соответствующих неким критериям. Вкладка **Фильтры (Filters)** содержит определения правил фильтрации обнаружения, позволяющих фильтровать обнаруженные значения:



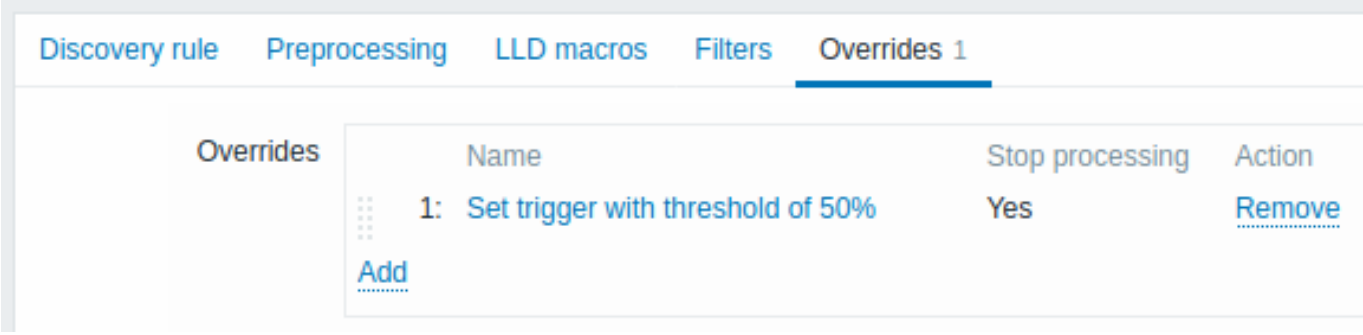
| Параметр | Описание |
|---|---|
| *Тип вычисления (Type of calculation)* | Доступны следующие опции расчёта фильтров:<br>**И (And)** — должны выполниться все фильтры;<br>**Или (Or)** — достаточно выполнения одного фильтра;<br>**И/Или (And/Or)** — используется *И* для разных имён макросов и *Или* для одинаковых имён макросов;<br>**Пользовательское выражение (Custom expression)** — появляется возможность указать пользовательское вычисление фильтров. Формула должна включать в себя все фильтры из списка. Ограничено 255 символами. |
| *Фильтры (Filters)* | Доступны следующие операторы условий фильтра: *совпадает*, *не соответствует*, *существует*, *не существует* (*matches*, *does not match*, *exists*, *does not exist*).<br>Операторы *совпадает* и *не соответствует* ожидают Perl Compatible Regular Expression (PCRE). Например, если вы заинтересованы только в файловых системах C:, D: и E:, вы можете поместить {#FSNAME} в текстовое поле «Макрос», а регулярное выражение «^C\|^D\|^E» — в поле «Регулярное выражение». Фильтрация также возможна по типам файловых систем, при использовании макроса {#FSTYPE} (например, «^ext\|^reiserfs») и по типу диска (поддерживается только Windows агентами), используя макрос {#FSDRIVETYPE} (например, «fixed»).<br>В поле «Регулярное выражение» вы можете ввести регулярное выражение или ссылку на глобальное регулярное выражение.<br>Для проверки регулярного выражения вы можете использовать «grep -E», например: `for f in ext2 nfs reiserfs smbfs; do echo $f \| grep -E '^ext\|^reiserfs' \|\| echo "SKIP: $f"; done`<br><br>Операторы *существует* и *не существует* позволяют фильтровать объекты на основе присутствия или отсутствия указанных LLD макросов в ответе.<br><br>Обратите внимание, что если какой-то макрос из фильтра пропущен в ответе, найденный объект будет игнорироваться, если только для этого макроса не задано условие «не существует».<br><br>Если отсутствие макроса влияет на результат выражения, то будет отображено предупреждение. Например, если {#B} отсутствует в:<br>{#A} совпадает 1 и {#B} совпадает 2 — выдаст предупреждение<br>{#A} совпадает 1 или {#B} совпадает 2 — предупреждения не будет. |

**Warning:**
Ошибка или опечатка в регулярном выражении, которое используется в LLD правиле (например, некорректное регулярное выражение «File systems for discovery»), может привести к удалению тысяч элементов конфигурации, данных истории и событий на большом количестве узлов сети.

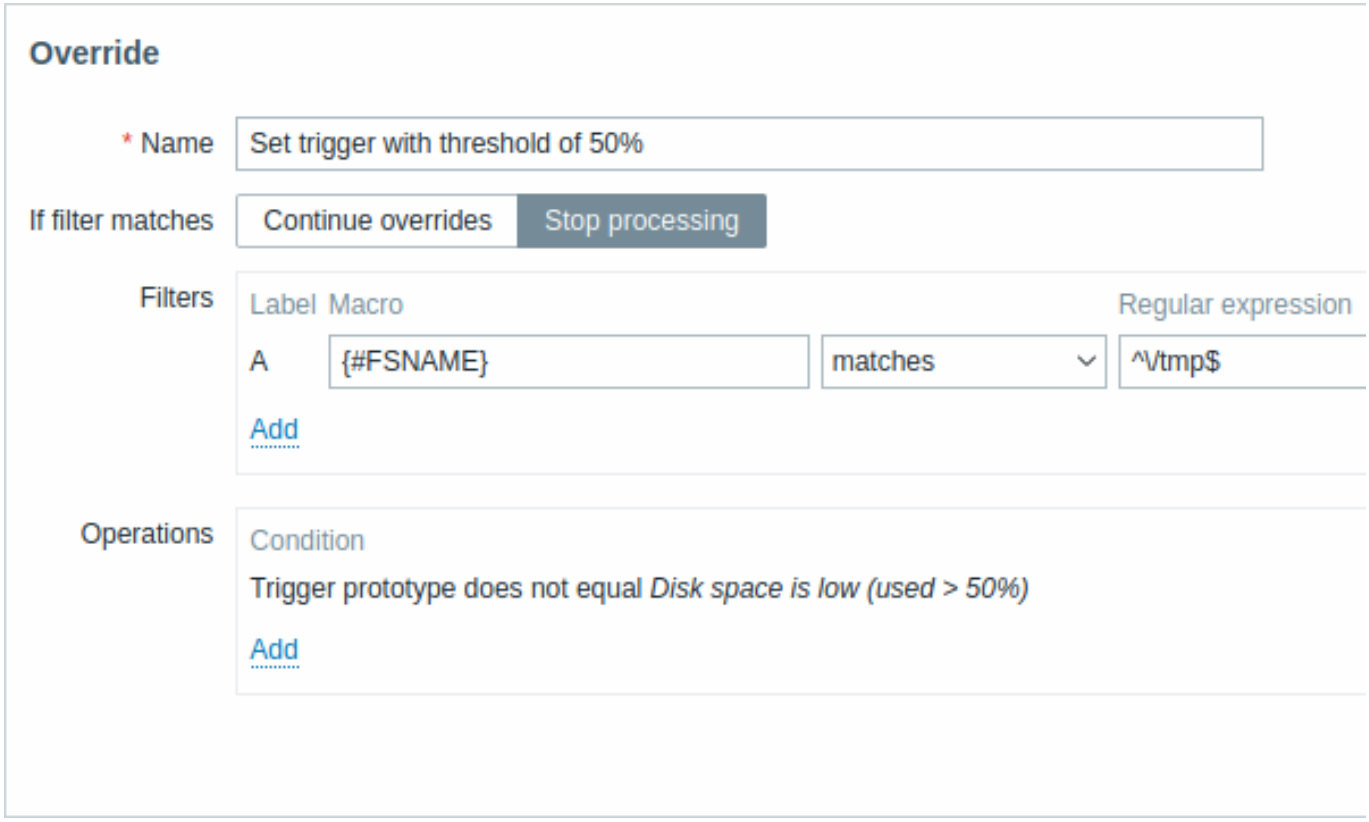
**Attention:**
Если имена файловых систем различаются только по регистру, то для корректной работы обнаружения необходимо, чтобы база данных Zabbix в MySQL была создана чувствительной к регистру.

Переопределение

Вкладка **Переопределения** позволяет задавать правила для изменения списка прототипов элементов данных, триггеров, графиков, узлов сети и обнаружения или их атрибутов для обнаруженных объектов, которые соответствуют заданным критериям.



Переопределения (если они есть) отображаются в списке с возможностью изменения порядка перетаскиванием и выполняются в порядке их определения. Чтобы настроить параметры нового переопределения, нажмите Add в блоке *Overrides*. Чтобы изменить существующее переопределение, нажмите на его имя. Откроется всплывающее окно, в котором можно изменить параметры правила переопределения.

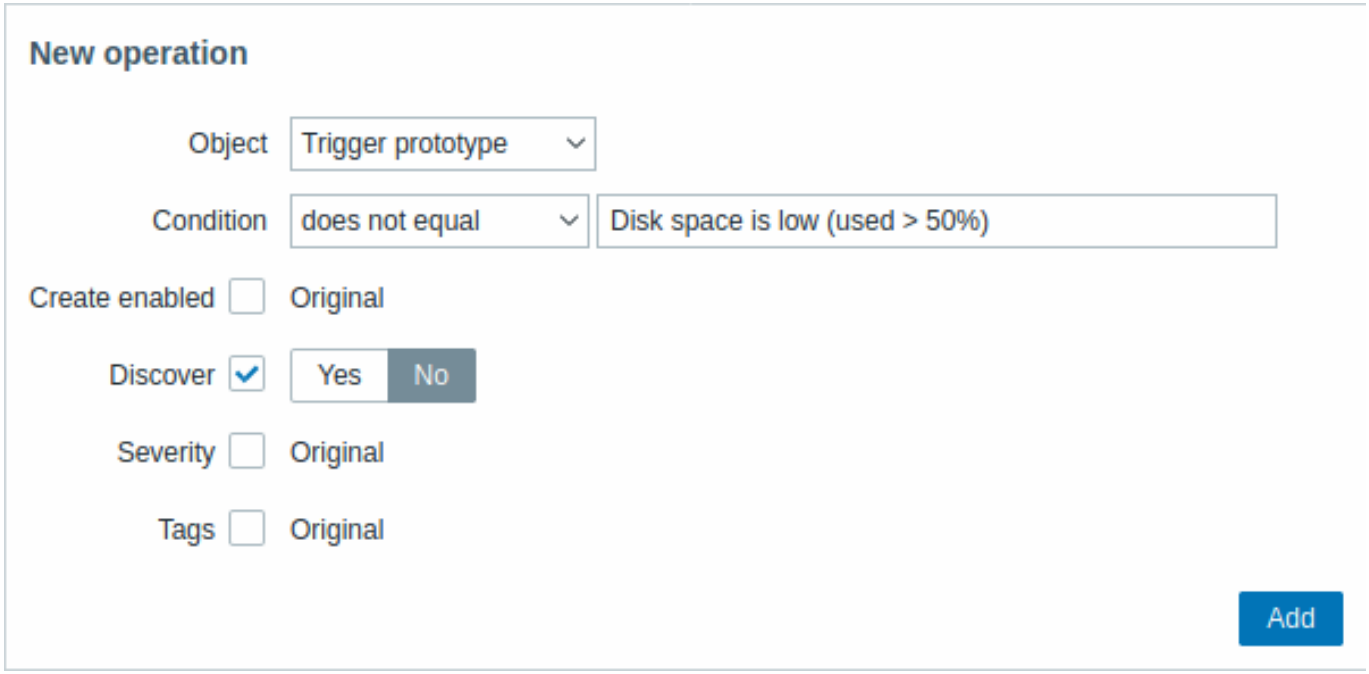



Все обязательные параметры отмечены красными звездочками.

| Parameter | Description |
|---|---|
| *Name* | Уникальное имя переопределения (в пределах правила LLD). |
| *If filter matches* | Определяет, будут ли обрабатываться следующие переопределения, если условия фильтра выполнены:<br>**Continue overrides** - последующие переопределения будут обработаны.<br>**Stop processing** - будут выполнены операции из предыдущих (если есть) и этого переопределения, последующие переопределения будут проигнорированы для совпавших строк LLD. |
| *Filters* | Определяет, к каким обнаруженным сущностям следует применять переопределение.<br>Фильтры переопределения обрабатываются после фильтров правила обнаружения и имеют ту же функциональность. |
| *Operations* | Операции переопределения отображаются со следующими сведениями:<br>**Condition** - тип объекта и условие, которому должно соответствовать имя объекта; например: прототип триггера не равен *Disk space is low (used > 50%)*.<br>**Actions** - отображаются ссылки для редактирования и удаления операции. |

#### Настройка операции

Чтобы настроить параметры новой операции, нажмите Add в блоке Operations. Чтобы изменить существующую операцию, нажмите Edit рядом с операцией. Откроется всплывающее окно, в котором можно изменить параметры операции.




| Parameter | Description |
|---|---|
| *Object* | Доступны пять типов объектов:<br>Прототип элемента данных<br>Прототип триггера<br>Прототип графика<br>Прототип узла сети<br>Прототип обнаружения |
| *Condition* | Позволяет фильтровать сущности, к которым следует применять операцию. |
| Operator | Поддерживаемые операторы:<br>**equals** - применить к этому прототипу<br>**does not equal** - применить ко всем прототипам, кроме этого<br>**contains** - применить, если имя прототипа содержит эту строку<br>**does not contain** - применить, если имя прототипа не содержит эту строку<br>**matches** - применить, если имя прототипа соответствует регулярному выражению<br>**does not match** - применить, если имя прототипа не соответствует регулярному выражению |
| Pattern | Регулярное выражение или строка для поиска. |
| Object:<br>*Item pro-to-type* | |
| *Create enabled* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа элемента данных:<br>*Yes* - элемент данных будет добавлен в включенном состоянии.<br>*No* - элемент данных будет добавлен к обнаруженной сущности, но в отключенном состоянии. |
| *Discover* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа элемента данных:<br>*Yes* - элемент данных будет добавлен.<br>*No* - элемент данных не будет добавлен. |

| Parameter | | Description |
|---|---|---|
| | *Update interval* | Если флажок установлен, появятся два параметра, позволяющие задать другой интервал для элемента данных:<br>*Delay* - интервал обновления элемента данных. Поддерживаются пользовательские макросы, LLD-макросы и суффиксы времени (например, 30s, 1m, 2h, 1d) (поддержка LLD-макросов восстановлена в Zabbix 7.4.11). Если используется *Custom interval*, следует установить значение 0.<br>*Custom interval* - нажмите Add, чтобы указать гибкие интервалы или интервалы по расписанию. Подробности см. в разделе Пользовательские интервалы. |
| | *History* | Если флажок установлен, появятся кнопки, позволяющие задать другой период хранения истории для элемента данных:<br>*Do not store* - если выбрано, история не будет храниться.<br>*Store up to* - если выбрано, справа появится поле для ввода периода хранения. Поддерживаются пользовательские макросы и LLD-макросы (поддержка LLD-макросов восстановлена в Zabbix 7.4.11). |
| | *Trends* | Если флажок установлен, появятся кнопки, позволяющие задать другой период хранения трендов для элемента данных:<br>*Do not store* - если выбрано, тренды не будут храниться.<br>*Store up to* - если выбрано, справа появится поле для ввода периода хранения. Поддерживаются пользовательские макросы и LLD-макросы (поддержка LLD-макросов восстановлена в Zabbix 7.4.11). |
| | *Tags* | Если флажок установлен, появится новый блок, позволяющий указать пары тег-значение. Поддерживаются пользовательские макросы и LLD-макросы.<br>Эти теги будут добавлены к тегам, указанным в прототипе элемента данных, даже если имена тегов совпадают. |
| Object: *Trigger prototype* | | |
| | *Create enabled* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа триггера:<br>*Yes* - триггер будет добавлен в включенном состоянии.<br>*No* - триггер будет добавлен к обнаруженной сущности, но в отключенном состоянии. |
| | *Discover* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа триггера:<br>*Yes* - триггер будет добавлен.<br>*No* - триггер не будет добавлен. |
| | *Severity* | Если флажок установлен, появятся кнопки уровня серьезности триггера, позволяющие изменить его уровень серьезности. |
| | *Tags* | Если флажок установлен, появится новый блок, позволяющий указать пары тег-значение. Поддерживаются пользовательские макросы и LLD-макросы.<br>Эти теги будут добавлены к тегам, указанным в прототипе триггера, даже если имена тегов совпадают. |
| Object: *Graph prototype* | | |
| | *Discover* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа графика:<br>*Yes* - график будет добавлен.<br>*No* - график не будет добавлен. |
| Object: *Host prototype* | | |

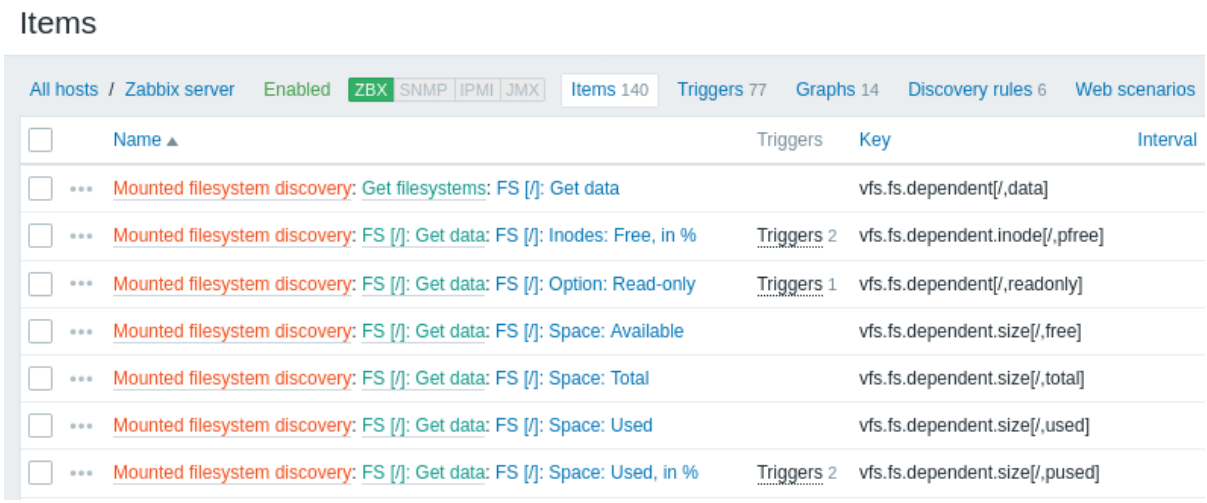
| Parameter | Description |
|---|---|
| *Create enabled* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа узла сети:<br>*Yes* - узел сети будет создан в включенном состоянии.<br>*No* - узел сети будет создан в отключенном состоянии. |
| *Discover* | Если флажок установлен, появятся кнопки, позволяющие переопределить исходные настройки прототипа узла сети:<br>*Yes* - узел сети будет обнаружен.<br>*No* - узел сети не будет обнаружен. |
| *Link templates* | Если флажок установлен, появится поле для ввода шаблонов. Начните вводить имя шаблона или нажмите *Select* рядом с полем и выберите шаблоны из списка во всплывающем окне.<br>Шаблоны из этого переопределения будут добавлены ко всем шаблонам, уже связанным с прототипом узла сети. |
| *Tags* | Если флажок установлен, появится новый блок, позволяющий указать пары тег-значение. Поддерживаются пользовательские макросы и LLD-макросы.<br>Эти теги будут добавлены к тегам, указанным в прототипе узла сети, даже если имена тегов совпадают. |
| *Host inventory* | Если флажок установлен, появятся кнопки, позволяющие выбрать другой режим инвентаризации для прототипа узла сети:<br>*Disabled* - не заполнять инвентарные данные узла сети<br>*Manual* - указывать сведения вручную<br>*Automated* - автоматически заполнять инвентарные данные узла сети на основе собранных метрик. |

Кнопки диалога

Кнопки в нижней части диалога позволяют выполнить несколько видов операций.

| | |
|---|---|
| Add (Добавить) | Добавление правила обнаружения. Эта кнопка доступна только для новых правил обнаружения. |
| Update (Обновить) | Обновление свойств правила обнаружения. Эта кнопка доступна только для уже существующих правил обнаружения. |
| Clone (Клонировать) | Создание другого правила обнаружения на основе свойств текущего правила обнанужения. |
| Execute now (Выполнить сейчас) | Выполнение немедленного обнаружения на основе правила обнаружения. Правило обнаружения должно существовать. Смотрите более подробную информацию.<br>*Обратите внимание*, что когда обнаружение выполняется немедленно, кэш конфигурации не обновляется, поэтому на результат не повлияют совсем недавние изменения настроек правила обнаружения. |
| Test (Тест) | Протестировать настройки правила обнаружения. Используйте эту кнопку для проверки настроек конфигурации (таких как подключение и корректность параметров) без постоянного применения каких-либо изменений. |
| Delete (Удалить) | Удаление правила обнаружения |
| Cancel (Отмена) | Отмена изменения свойств правила обнаружения. |

**Обнаруженные объекты** Представленные ниже снимки экрана иллюстрируют, как выглядят уже обнаруженные элементы данных, триггеры и графики в настройках узла сети. Обнаруженные объекты имеют префикс — ссылку золотистого цвета, которая ведёт к правилу обнаружения, создавшему эти объекты.




Обратите внимание, что обнаруженные объекты не будут созданы в случае, если объекты с такими же условиями уникальности уже существуют — например, элемент данных с таким же ключом или график с таким же именем. В таком случае в веб-интерфейсе отобразится сообщение об ошибке, что правило низкоуровневого обнаружения не смогло создать определённые объекты. Само правило обнаружения, тем не менее, не станет неподдерживаемым из-за того, что некоторые объекты не смогли быть созданы и были пропущены. Правило обнаружения перейдёт к созданию/обновлению других объектов.

Если обнаруженный объект (узел сети, файловая система, интерфейс и т.д.) более не обнаруживается (или более не попадает под фильтр), то объекты, созданные на его основе, могут быть автоматически деактивированы и впоследствии удалены.

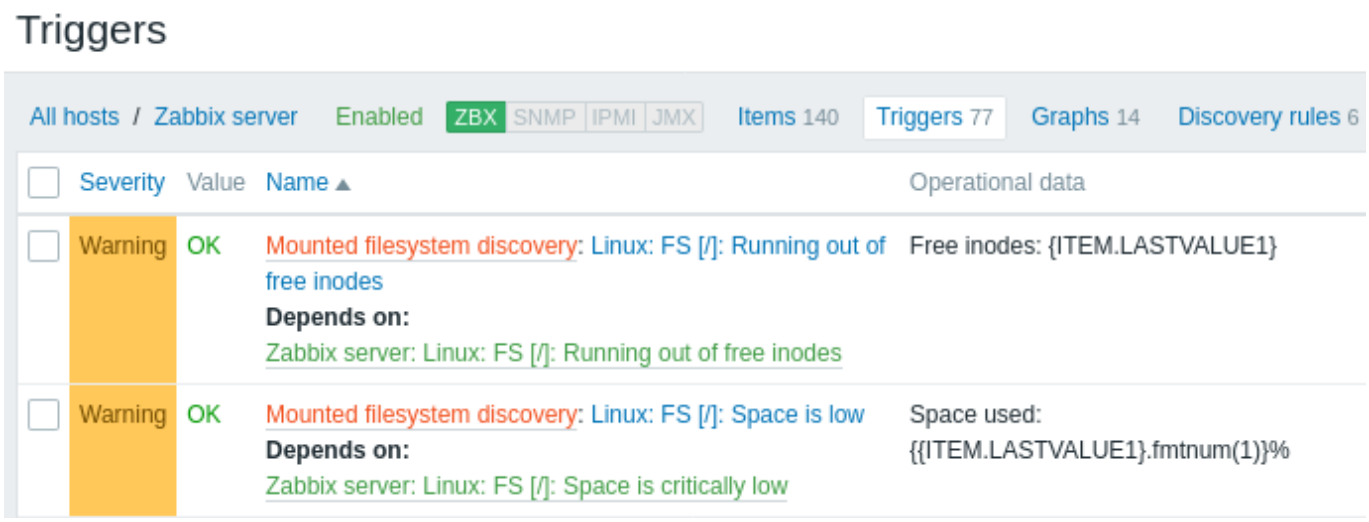
Потерянные ресурсы могут быть автоматически деактивированы на основе значения параметра *Удаление потерянных ресурсов (Delete lost resources)*. Это влияет на потерянные узлы сети, группы узлов сети, элементы данных, триггеры и графики.

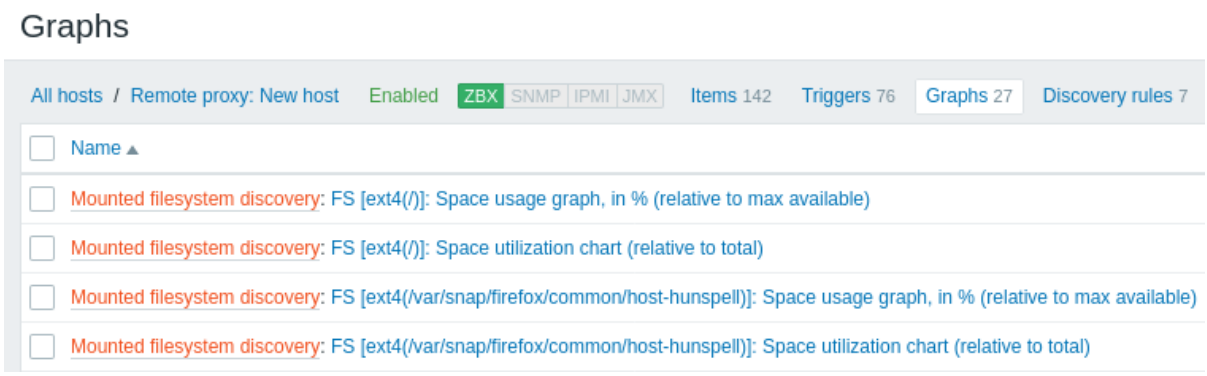
Потерянные ресурсы могут быть автоматически удалены на основе значения параметра *Деактивация потерянных ресурсов (Disable lost resources)*. Это влияет на потерянные узлы сети, элементы данных и триггеры.

Когда обнаруженный объект становится «Более не обнаруживается», в списке элементов данных будет отображаться оранжевый индикатор времени жизни. Переместите курсор мыши на этот индикатор, и вы увидите сообщение с количеством дней до момента удаления элемента данных.

Если объекты помечены на удаление, но не были удалены в назначенное время (деактивировано правило обнаружения или узел сети элемента данных), они удалятся при следующем выполнении правила обнаружения.

Объекты, которые содержат другие объекты, которые помечены на удаление, не будут обновлены, если будут изменены на уровне правила обнаружения. Например, триггеры на основе LLD не будут обновлены, если они содержат элементы данных, которые помечены на удаление.




**Другие типы обнаружения** Для получения более подробных сведений и инструкций по остальным типам доступных «из коробки» обнаружений смотрите следующие разделы:

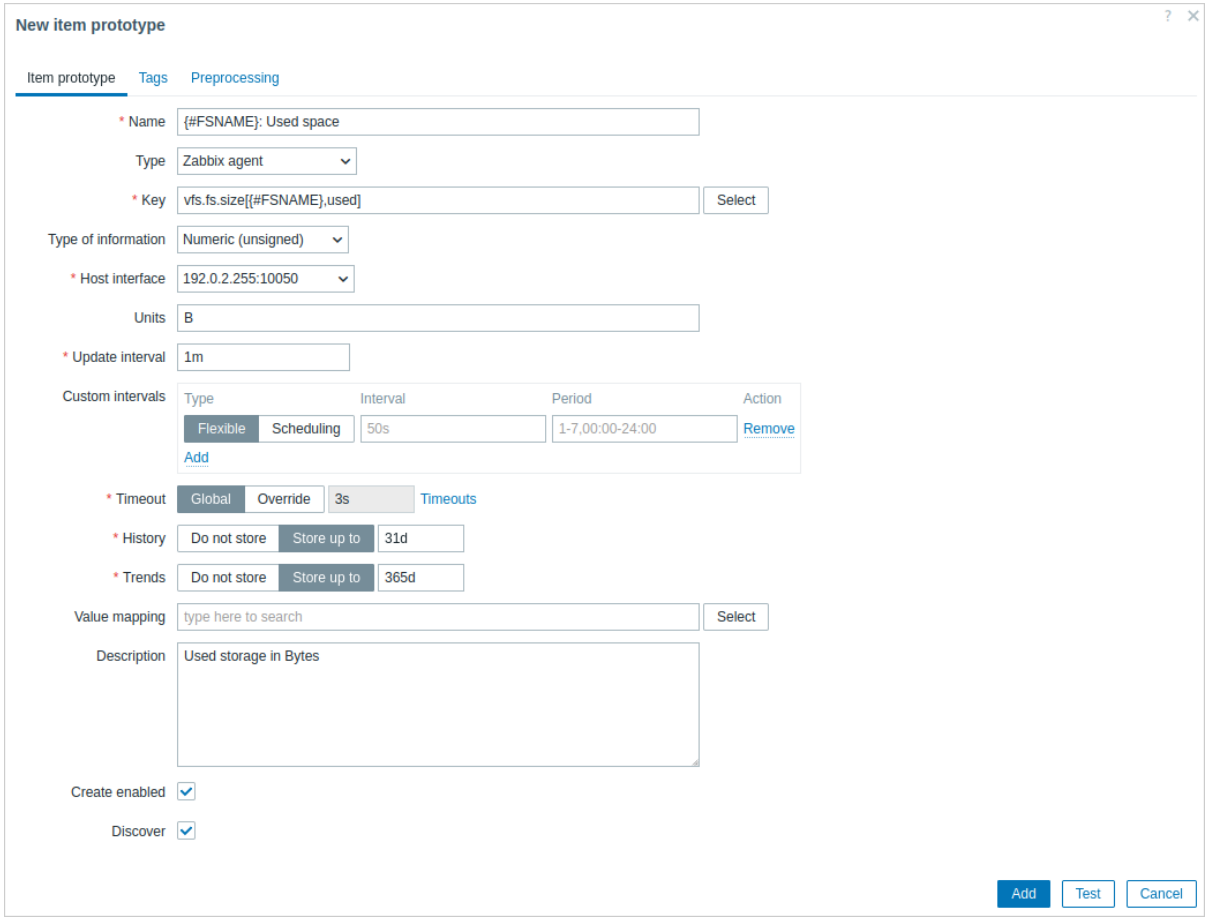
- обнаружение сетевых интерфейсов;
- обнаружение CPU и ядер CPU;
- обнаружение SNMP OID-ов;
- обнаружение объектов JMX;
- обнаружение с использованием запросов ODBC SQL;
- обнаружение служб Windows;
- обнаружение интерфейсов узлов сети в Zabbix.

Для получения более подробных сведений касательно JSON формата по обнаружению элементов данных и примера, каким образом реализовать своё собственное обнаружение файловых систем при помощи Perl скрипта, смотрите создание пользовательских правил LLD.

#### 1 Прототипы элементов данных

После того как правило создано, перейдите к элементам данных для этого правила и нажмите *Create item prototype*, чтобы создать прототип элемента данных.

Обратите внимание, как макрос {#FSNAME} используется там, где требуется имя файловой системы. Использование макроса низкоуровневого обнаружения обязательно в ключе элемента данных, чтобы обнаружение было обработано корректно. Когда правило обнаружения обрабатывается, этот макрос будет подставлен вместо обнаруженной файловой системы.




**Note:**
Когда параметры ключа элемента данных не используются, поместите LLD-макрос внутри квадратных скобок ключа элемента данных `[...]` (например `v_[{#MACRO}]`).

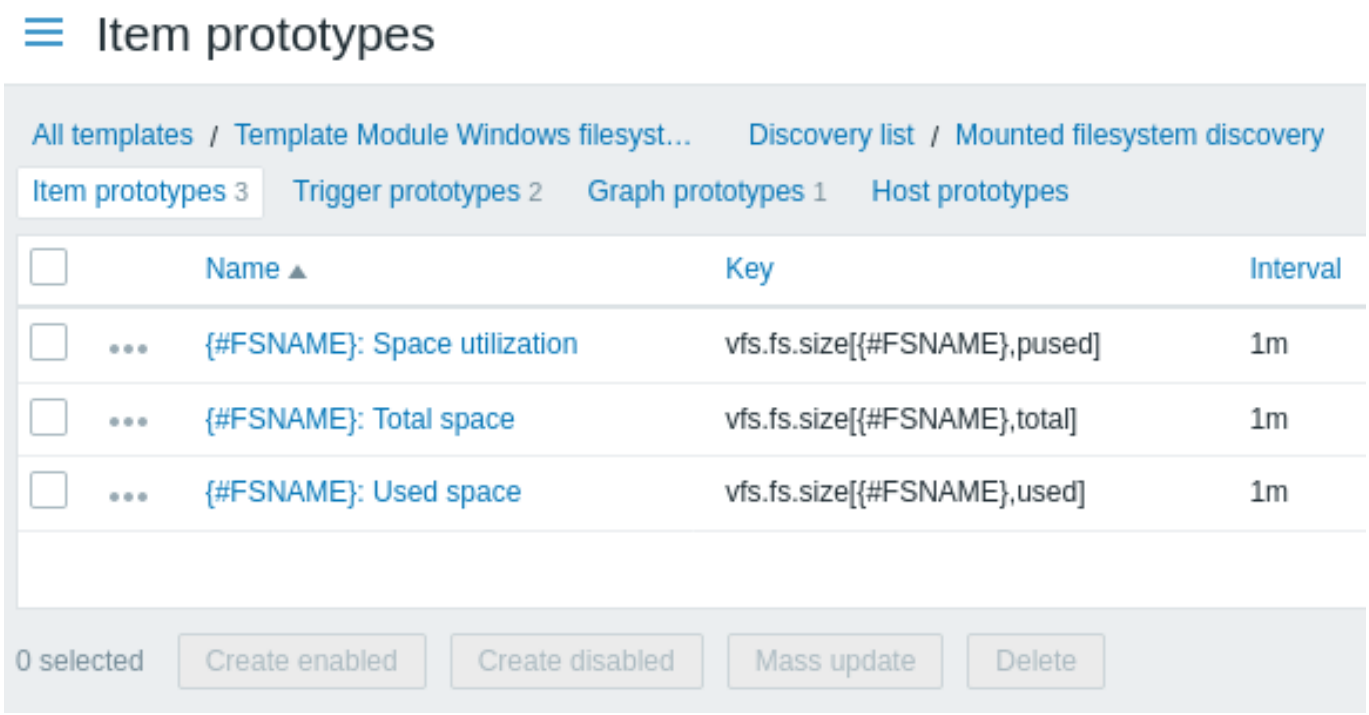
Низкоуровневое обнаружение макросы и пользовательские макросы поддерживаются в конфигурации прототипа элемента данных и параметрах предварительной обработки значения элемента данных parameters. Обратите внимание, что при использовании в интервалах обновления один макрос должен заполнять все поле. Несколько макросов в одном поле или макросы, смешанные с текстом, не поддерживаются.

**Note:**
Контекстно-зависимое экранирование макросов низкоуровневого обнаружения выполняется для безопасного использования в параметрах предварительной обработки регулярных выражений и XPath.

Атрибуты, специфичные для прототипов элементов данных:

| Parameter | Description |
|---|---|
| *Create enabled* | Если флажок установлен, элемент данных будет добавлен в включенном состоянии.<br>Если флажок снят, элемент данных будет добавлен в обнаруженную сущность, но в отключенном состоянии. |
| *Discover* | Если флажок установлен (по умолчанию), элемент данных будет добавлен в обнаруженную сущность.<br>Если флажок снят, элемент данных не будет добавлен в обнаруженную сущность, если только эта настройка не будет переопределена в правиле обнаружения. |

Для каждого интересующего нас показателя файловой системы можно создать несколько прототипов элементов данных:



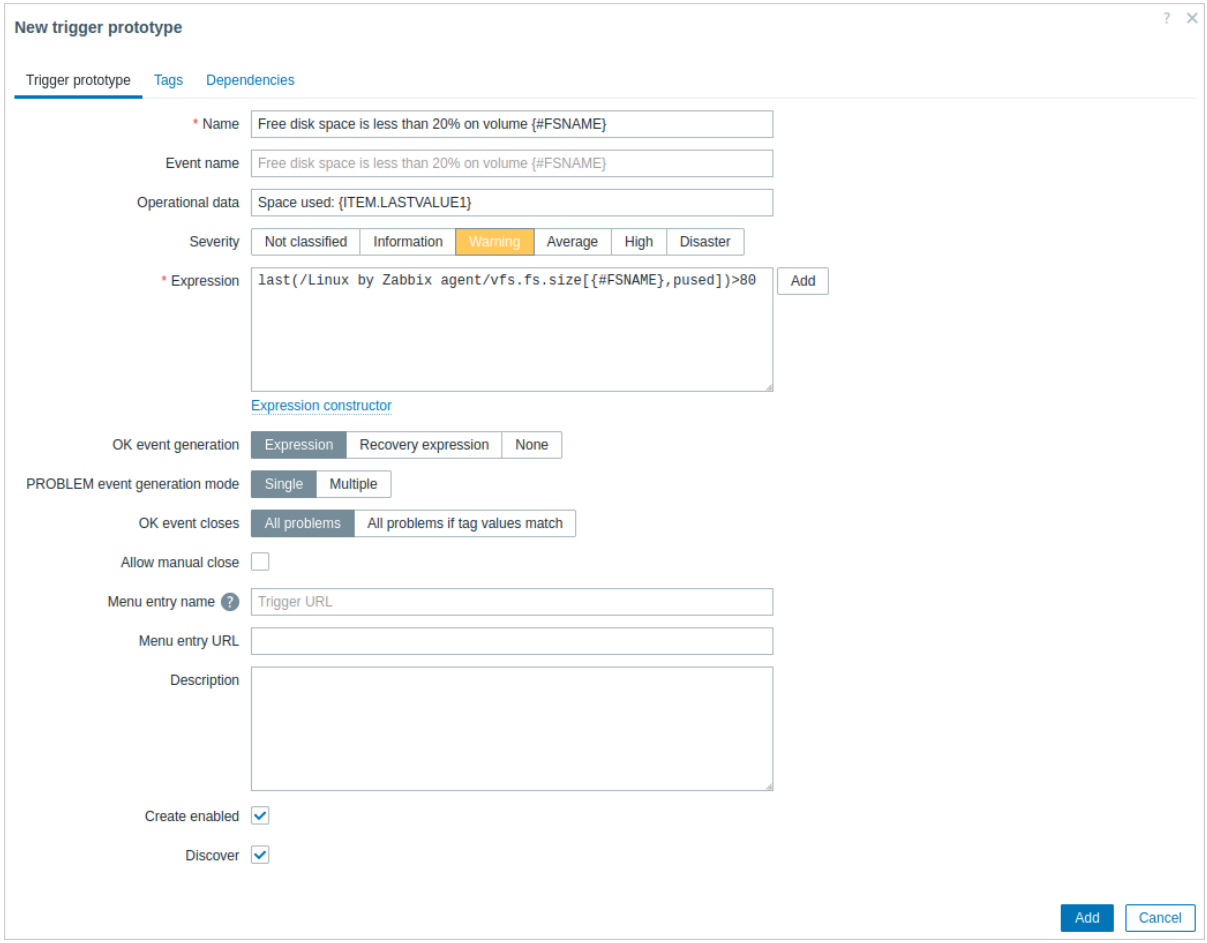

Нажмите значок с тремя точками, чтобы открыть меню для конкретного прототипа элемента данных с такими параметрами:

- *Create trigger prototype* - создать прототип триггера на основе этого прототипа элемента данных
- *Trigger prototypes* - нажмите, чтобы увидеть список со ссылками на уже настроенные прототипы триггеров этого прототипа элемента данных
- *Create dependent item* - создать зависимый элемент данных для этого прототипа элемента данных

Опция *Mass update* доступна, если вы хотите обновить свойства нескольких прототипов элементов данных одновременно.

**2. Прототипы триггеров**

Прототипы триггеров мы создаём аналогично тому, как создавали прототипы элементов данных:

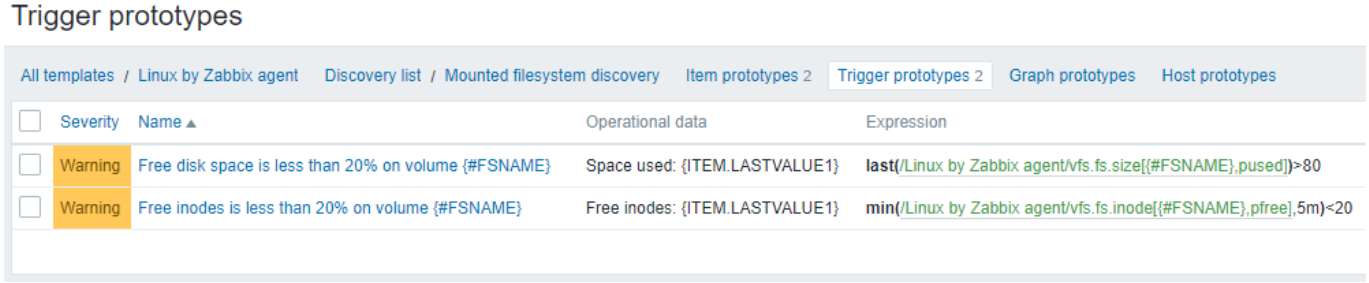



Атрибуты, специфичные для прототипов триггеров:

| Параметр | Описание |
|---|---|
| *Создать активированным (Create enabled)* | Если отмечено, триггер будет добавлен в активированном состоянии.<br>Если не отмечено, триггер будет добавлен для обнаруженного оъекта, но в деактивированном состоянии. |
| *Обнаружение (Discover)* | Если отмечено (по умолчанию), триггер будет добавлен для обнаруженного объекта.<br>Если не отмечено, триггер не будет добавлен для обнаруженного объекта, если только эта настройка не будет переопределена в правиле обнаружения. |

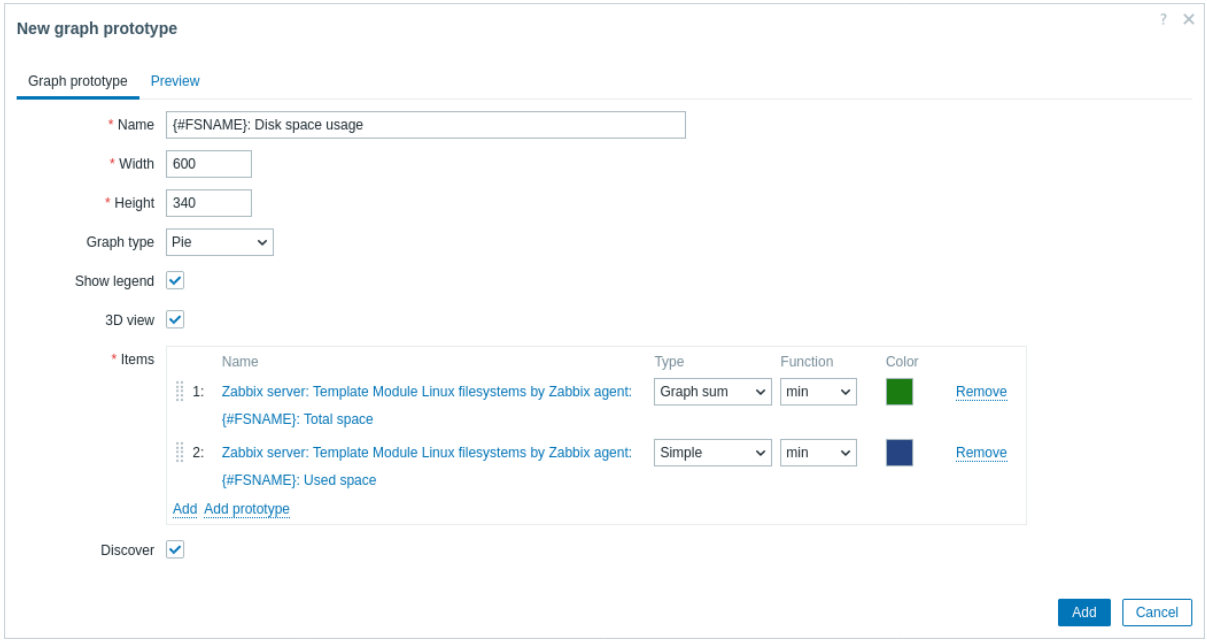
При создании из прототипа реального триггера может потребоваться бОльшая гибкость касательно констант, используемых в выражении для сравнения («20» в нашем примере). Смотрите, как пользовательские макросы с контекстом могут быть полезными для достижения такой гибкости.

Вы также можете определить зависимости между прототипами триггеров. Чтобы это сделать, перейдите на вкладку *Зависимости (Dependencies)*. Прототип триггера может зависеть от другого прототипа триггера из того же правила LLD или от обычного триггера. Прототип триггера не может зависеть от прототипа триггера из другого правила LLD или от триггера, созданного из прототипа триггера. Прототип триггера узла сети не может зависеть от триггера из шаблона.



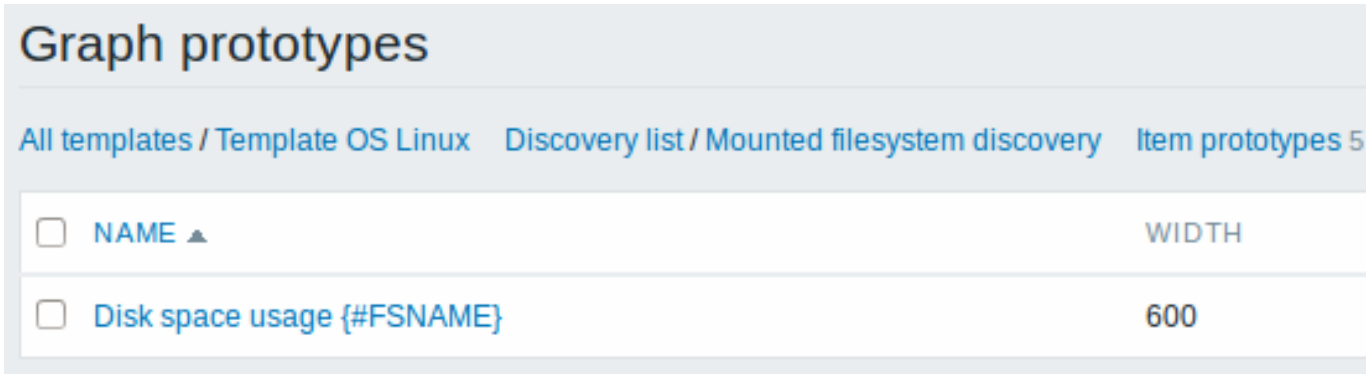

### 3 Прототипы графиков

Мы также можем создавать прототипы графиков:

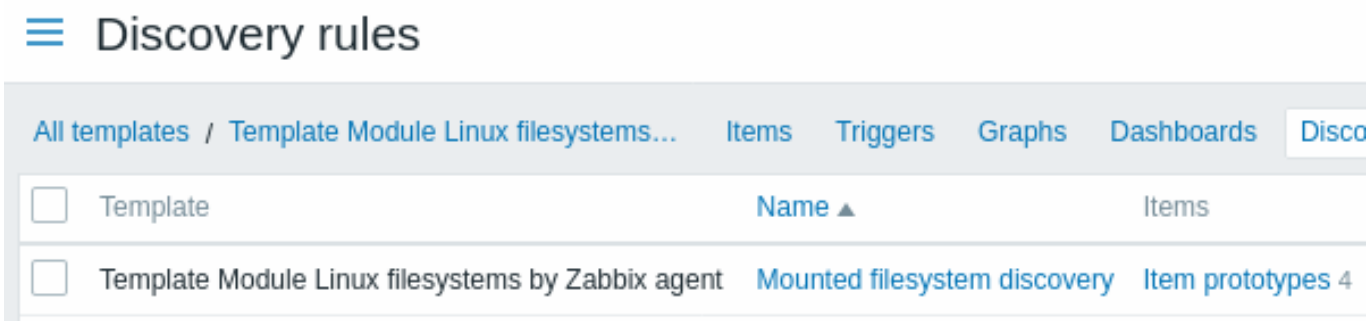



Атрибуты, специфичные для прототипов графиков:

| Parameter | Description |
|---|---|
| *Discover* | Если флажок установлен (по умолчанию), график будет добавлен к обнаруженному объекту.<br>Если флажок снят, график не будет добавлен к обнаруженному объекту, если только этот параметр не будет переопределен в правиле обнаружения. |




Наконец, мы создали правило обнаружения, которое выглядит так, как показано ниже. Оно содержит пять прототипов элементов данных, два прототипа триггеров и один прототип графика.




#### 4. Прототипы узлов сети

Прототипы узлов сети представляют собой заготовки для создания узлов сети с помощью правил низкоуровневого обнаружения. До того, как отработает обнаружение и будут созданы узлы сети, эти прототипы не могут иметь элементов данных и триггеров, за исключением тех, которые присоединены из шаблонов.

Конфигурация

Прототипы узлов сети настраиваются в правилах низкоуровневого обнаружения.

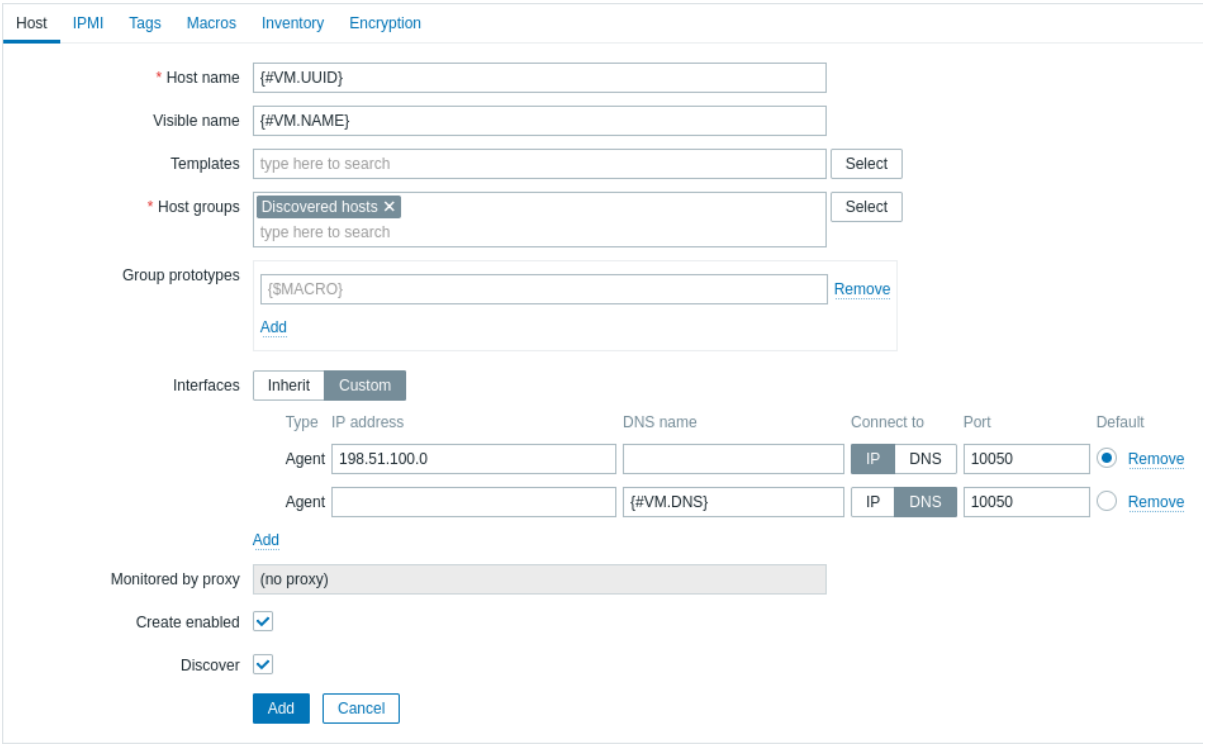
Чтобы создать прототип узла сети:

1. Перейдите в *Сбор данных → Узлы сети*.

2. Нажмите *Обнаружение* для нужного узла сети, чтобы перейти к списку правил низкоуровневого обнаружения, настроенных для этого узла сети.

3. Нажмите *Прототипы узлов сети* для нужного правила обнаружения.

4. Нажмите кнопку *Создать прототип узла сети* в правом верхнем углу.




Прототипы узлов сети имеют те же параметры, что и обычные узлы сети; однако следующие параметры поддерживают другую или дополнительную настройку:

| Parameter | Description |
|---|---|
| *Host name* | Этот параметр должен содержать как минимум один макрос низкоуровневого обнаружения, чтобы обеспечить уникальные имена узлов сети для создаваемых узлов сети. |
| *Visible name* | Поддерживаются макросы низкоуровневого обнаружения. |
| *Group prototypes* | Позволяет указывать прототипы групп узлов сети с использованием макросов низкоуровневого обнаружения.<br>На основе указанных прототипов групп будут обнаружены, созданы и привязаны к создаваемым узлам сети группы узлов сети; обнаруженные группы, которые уже были созданы другими правилами низкоуровневого обнаружения, также будут привязаны к создаваемым узлам сети. Однако обнаруженные группы узлов сети, которые совпадают с созданными вручную группами узлов сети, не будут привязаны к создаваемым узлам сети. |
| *Interfaces* | Укажите, должны ли обнаруженные узлы сети наследовать IP-адрес от узла сети, которому принадлежит правило обнаружения (по умолчанию), или получать пользовательские интерфейсы.<br>Поддерживаются макросы низкоуровневого обнаружения и пользовательские макросы. |
| *Create enabled* | Установите статус обнаруженных узлов сети; если флажок снят, узлы сети будут созданы как отключенные. |
| *Discover* | Укажите, будут ли узлы сети создаваться из прототипа узла сети; если флажок снят, узлы сети не будут создаваться из прототипа узла сети (если только этот параметр не будет переопределен в правиле низкоуровневого обнаружения). |

**Note:**
Макросы низкоуровневого обнаружения также поддерживаются для значений тегов и значений пользовательских макросов прототипа узла сети.<br> *Карты значений* не поддерживаются для прототипов узлов сети.

Пример настройки прототипа узла сети см. в разделе *Мониторинг виртуальных машин*.

Интерфейсы узла сети

Чтобы добавить пользовательские интерфейсы, переключите селектор *Интерфейсы (Interfaces)* из положения «Наследовать (Inherit)» в «Пользовательское (Custom)». Нажмите Add (Добавить) и выберите тип интерфейса — Zabbix агент, SNMP, JMX или IPMI.

**Note:**
Если выбрано *Пользовательское (Custom)*, но интерфейс не указан, узел сети будет создан без интерфейсов.<br> Если выбрано *Наследовать (Inherit)*, а прототип узла сети принадлежит шаблону, то все обнаруженные узлы сети унаследуют интерфейс узла сети от того узла сети, к которому присоединён шаблон.

Если указано несколько пользовательских интерфейсов, то основной интерфейс можно задать в столбце *По умолчанию (Default)*.

Для примера того, как настраивать пользовательский интерфейс узла сети, смотрите *Пример настройки мониторинга VMware*.

**Warning:**
Узел сети будет создан, только если интерфейс узла сети содержит корректные данные.

Обнаруженные узлы сети

В списке узлов сети обнаруженные узлы сети помечаются префиксом с именем правила обнаружения, которое их создало.

Обнаруженные узлы сети наследуют большинство параметров от прототипов узлов сети как *только для чтения*. Настраивать можно только следующие параметры обнаруженного узла сети:

- *Шаблоны* - привязывать новые шаблоны или отвязывать добавленные вручную. Шаблоны, унаследованные от прототипа узла сети, отвязать нельзя.
- *Описание* - добавлять или редактировать описание узла сети.
- *Состояние* - включать или отключать узел сети.
- *Теги* - добавлять новые теги или удалять добавленные вручную. Теги, унаследованные от прототипа узла сети, удалить нельзя. Обратите внимание, что теги не могут иметь дубликатов (теги с одинаковыми именем и значением). Если в прототип узла сети добавлен новый тег и он совпадает с тегом, добавленным вручную на обнаруженном узле сети, вручную добавленный тег будет заменен унаследованным тегом во время обнаружения.
- *Макросы* - добавлять новые макросы узла сети или удалять добавленные вручную, а также унаследованные от прототипа узла сети; изменять значения макросов и типы.

**Note:**
Обнаруженные узлы сети также наследуют пользовательские макросы от узла сети, на котором настроено правило обнаружения. Эти унаследованные макросы можно удалять, а их значения и типы можно изменять.

Обнаруженные узлы сети также могут обнаруживать другие узлы сети. Например, если у прототипа узла сети (используемого для обнаружения гипервизоров) есть привязанный к нему шаблон, который включает правило низкоуровневого обнаружения со своим собственным прототипом узла сети (используемым для обнаружения виртуальных машин), Zabbix обнаружит и гипервизоры, и их виртуальные машины. Чтобы дополнительно обнаруживать контейнеры на этих ВМ, можно создать правила низкоуровневого обнаружения или привязать новый шаблон с прототипами узлов сети к обнаруженным ВМ либо предварительно настроить прототип узла сети ВМ, привязав шаблон, который сам включает прототипы узлов сети.

Обнаруженные узлы сети можно удалять вручную. Однако обратите внимание, что они будут обнаружены снова, если для них включено обнаружение.

Узлы сети, которые больше не обнаруживаются, могут быть:

- автоматически отключены (на основе значения *Disable lost resources* правила обнаружения)
- автоматически удалены (на основе значения *Delete lost resources* правила обнаружения).

## 5 Прототипы обнаружения

Обзор

Прототипы обнаружения — это **вложенные правила низкоуровневого обнаружения** внутри «родительского» правила обнаружения, которые позволяют создавать многоуровневое обнаружение объектов с собственными элементами данных, триггерами и т. д. Например, вы можете захотеть обнаружить все экземпляры базы данных на сервере баз данных, затем обнаружить табличные пространства для каждого экземпляра, а затем обнаружить таблицы для каждого табличного пространства.

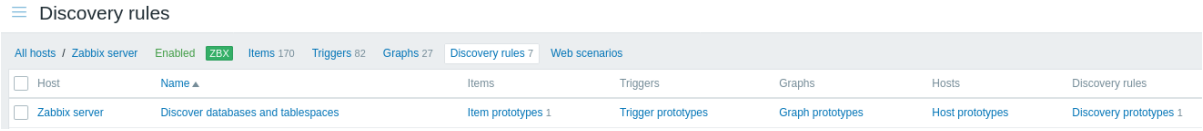
Прототипы обнаружения имеют собственные прототипы элементов данных, триггеров, графиков, узлов сети и прототипы обнаружения. Вложенный прототип обнаружения будет использовать то же JSON-значение, что и родительское правило, если вы укажете тип *Nested*.

Уровни вложенности для прототипов обнаружения не ограничены.

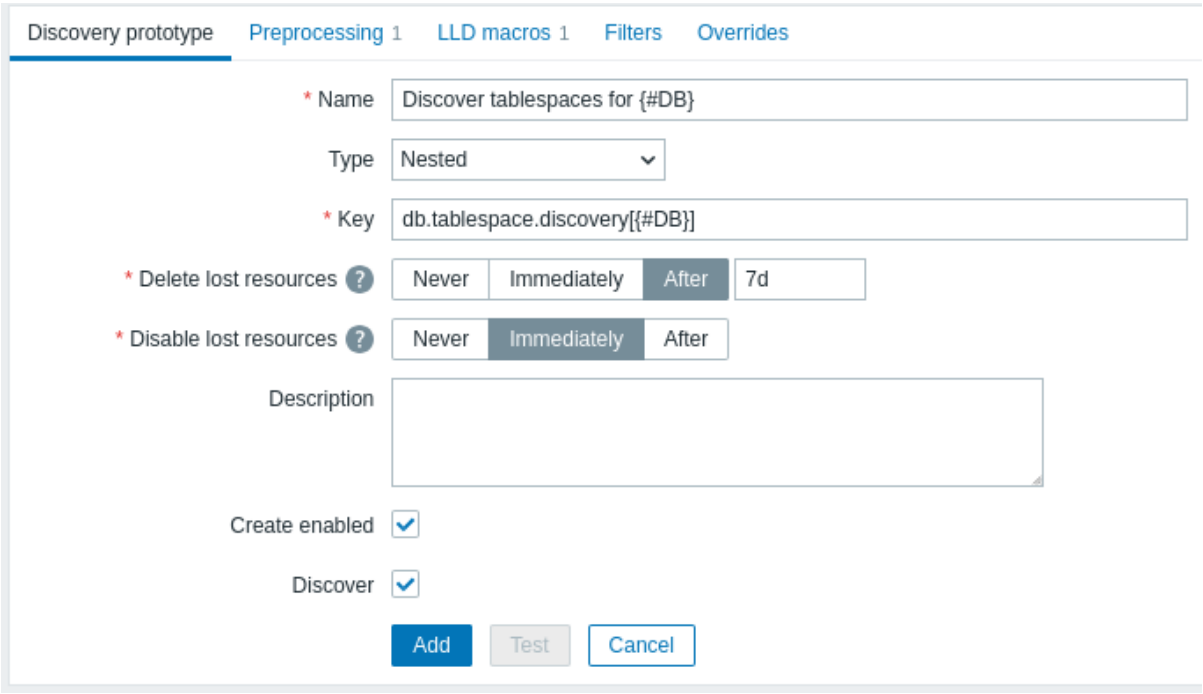
Конфигурация

Чтобы создать прототип обнаружения:

- Нажмите *Прототипы обнаружения* в строке существующего правила обнаружения




- Нажмите *Создать прототип обнаружения*




Поля конфигурации этой формы общие с обычным обнаружением на низком уровне.

Если в открывшейся форме прототипа обнаружения в качестве *Type* выбрать "Nested", то правила обнаружения (из прототипа обнаружения) будут генерироваться на основе JSON-объекта из того же JSON-значения, что и у родительского правила обнаружения. Например, если исходный JSON — [<object A>, <object B>] и существует один вложенный прототип правила обнаружения, то будут сгенерированы два правила обнаружения на основе данных object A и object B соответственно.

В этом случае прототип обнаружения активируется одновременно с родительским правилом. Таким образом, вложенное правило может использовать предварительную обработку, чтобы работать с другим "срезом" тех же данных, уже полученных родителем.

LLD-макросы из родительского LLD-правила доступны для вложенных правил обнаружения.

Вложенные правила LLD на обнаруженных узлах сети

*Вложенное* правило низкоуровневого обнаружения можно использовать в шаблоне узла сети, назначенном прототипу узла сети. Если на обнаруженном узле сети существует *вложенное* правило обнаружения, то JSON-объект, использованный

для обнаружения узла сети, также отправляется всем правилам LLD вложенного типа на этом узле сети. Подробнее см. пример.

Макросы LLD из правила обнаружения, которое создало узел сети, доступны для вложенных правил обнаружения.

Пример

Давайте рассмотрим возможное применение прототипов обнаружения на основе получения следующего примера многоуровневого 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" }
    ]
  }
]
```

Случай 1

Обнаружение экземпляров базы данных на сервере баз данных, а затем обнаружение табличных пространств для каждого экземпляра.

1. У вас есть как минимум один узел сети, связанный с обнаружением сервера баз данных.
2. Создайте правило LLD для этого узла сети с именем *Discover databases and tablespaces*.
3. Перейдите на вкладку *LLD Macros* в этом правиле, добавьте макрос `{#DB}=$.database`.
4. Добавьте прототип элемента данных для этого правила с именем *Active connections to {#DB}* (Type: Agent, Key: `db.connections[{#DB}]`).
5. Элементы данных, связанные с каждой базой данных, обнаруживаются:

```
Active connections to db1, Active connections to db2, Active connections to db3.
```

6. Создайте прототип обнаружения для этого правила с именем *Discover tablespaces for {#DB}* (Type: Nested, Key: `db.tablespace.discovery[{#DB}]`).
7. Перейдите на вкладку *Preprocessing* этого прототипа обнаружения и добавьте шаг `JSONPath=$.tablespaces`.
8. Перейдите на вкладку *LLD Macros* этого прототипа обнаружения, добавьте макрос `{#TSNAME}=$.name`.
9. Создайте прототип элемента данных для этого прототипа обнаружения с именем *Size of tablespace {#TSNAME} for {#DB}* (Type: Agent, Key: `db.ts.size[{#DB}, {#TSNAME}]`).

10. Элементы данных, связанные с каждым табличным пространством каждой базы данных, обнаруживаются:

```
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.
```

с ключами `db.ts.size[db1,ts1]`, `db.ts.size[db1,ts2]`, ... `db.ts.size[db3,ts3]`.

Случай 2

Обнаружение экземпляров базы данных на сервере базы данных путем представления их как обнаруженных узлов сети, а затем обнаружение табличных пространств для каждого экземпляра.

1. У вас есть как минимум один узел сети (корневой узел), связанный с обнаружением сервера базы данных.
2. Создайте шаблон для обнаружения табличных пространств для каждой базы данных.
3. Создайте в этом шаблоне элемент данных с именем *Active connections to {#DB}* (Тип: Agent, Ключ: `db.connections[{#DB}]`).
4. Создайте правило LLD для этого шаблона с именем *Discover tablespaces* (Тип: Nested).
5. Перейдите на вкладку *Preprocessing* этого правила и добавьте шаг `JSONPath=$.tablespaces`.
6. Перейдите на вкладку *LLD Macros* этого правила и добавьте макрос `{#TSNAME}=$.name`.
7. Создайте прототип элемента данных для этого правила с именем *Size of tablespace {#TSNAME} for {#DB}* (Тип: Agent, Ключ: `db.ts.size[{#DB}, {#TSNAME}]`).
8. Вернитесь к корневому узлу сети и создайте правило LLD для этого узла с именем *Discover databases and tablespaces*.
9. Перейдите на вкладку *LLD Macros* этого правила и добавьте макрос `{#DB}=$.database`.
10. Добавьте прототип узла сети для этого правила с именем *Host for database {#DB}*.
11. Перейдите на вкладку *Macros* этого прототипа узла сети и добавьте макрос `{$DB}={#DB}` (для имени и ключа элемента данных из шага 3).
12. Свяжите шаблон из шага 2 с этим прототипом узла сети.
13. Обнаруженные узлы сети содержат обнаруженные элементы данных, связанные с каждой базой данных и ее табличными пространствами:

| Узел сети | Элементы данных |
|---|---|
| *Host for database db1* | Active connections to db1 |
| | Size of tablespace ts1 for db1 |
| | Size of tablespace ts2 for db1 |
| | Size of tablespace ts3 for db1 |
| *Host for database db2* | Active connections to db2 |
| | Size of tablespace ts1 for db2 |
| | Size of tablespace ts2 for db2 |
| | Size of tablespace ts3 for db2 |
| *Host for database db3* | Active connections to db3 |
| | Size of tablespace ts1 for db3 |
| | Size of tablespace ts2 for db3 |
| | Size of tablespace ts3 for db3 |

#### 6 Примечания по низкоуровневому обнаружению

Использование макросов LLD в контекстах пользовательских макросов

Макросы LLD могут использоваться внутри контекста пользовательских макросов, например, в прототипах триггеров.

Несколько правил LLD для того же элемента данных

Можно определить несколько правил низкоуровневого обнаружения, использующих тот же элемент данных для обнаружения.

Чтобы сделать это, вам необходимо определить параметр агента Alias, чтобы разрешить использовать отличающиеся ключи элементов данных обнаружений в различных правилах обнаружения, например: `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` и т.д.

Ограничения данных для возвращаемых значений

Если значение получается непосредственно Zabbix сервером, то ограничений на данные JSON правил низкоуровневого обнаружения нет, так как возвращаемые значения обрабатываются без сохранения в базе данных.

Также нет ограничений для пользовательских правил низкоуровневого обнаружения. Однако, если пользовательские данные LLD опрашиваются с помощью пользовательских параметров, то применяется ограничение на значение, возвращаемое пользовательским параметром.

Если же данные поступают через Zabbix прокси, он должен сохранять эти данные в своей базе данных. В этом случае применяются ограничения баз данных.

### 7 Правила обнаружения

В этом разделе приведены практические примеры настройки правил обнаружения для различных типов сущностей, таких как файловые системы, сетевые интерфейсы, службы, датчики, объекты SNMP, Java-приложения и базы данных.

Каждый пример показывает, как может быть структурировано правило обнаружения, включая используемый ключ элемента данных или запрос, JSON-вывод с макросами LLD и то, как эти результаты используются для создания прототипов элементов данных, триггеров и графиков.

Чтобы ознакомиться с конкретным типом правила обнаружения, используйте боковую панель для перехода на соответствующую страницу.

#### 1. Обнаружение примонтированных файловых систем

Обзор

Существует возможность обнаружения примонтированных файловых систем и их свойств:

- имя точки монтирования
- тип файловой системы
- размер файловой системы
- статистика по файловам дескрипторам — inode
- опции монтирования

Чтобы это сделать, можно использовать комбинацию из:

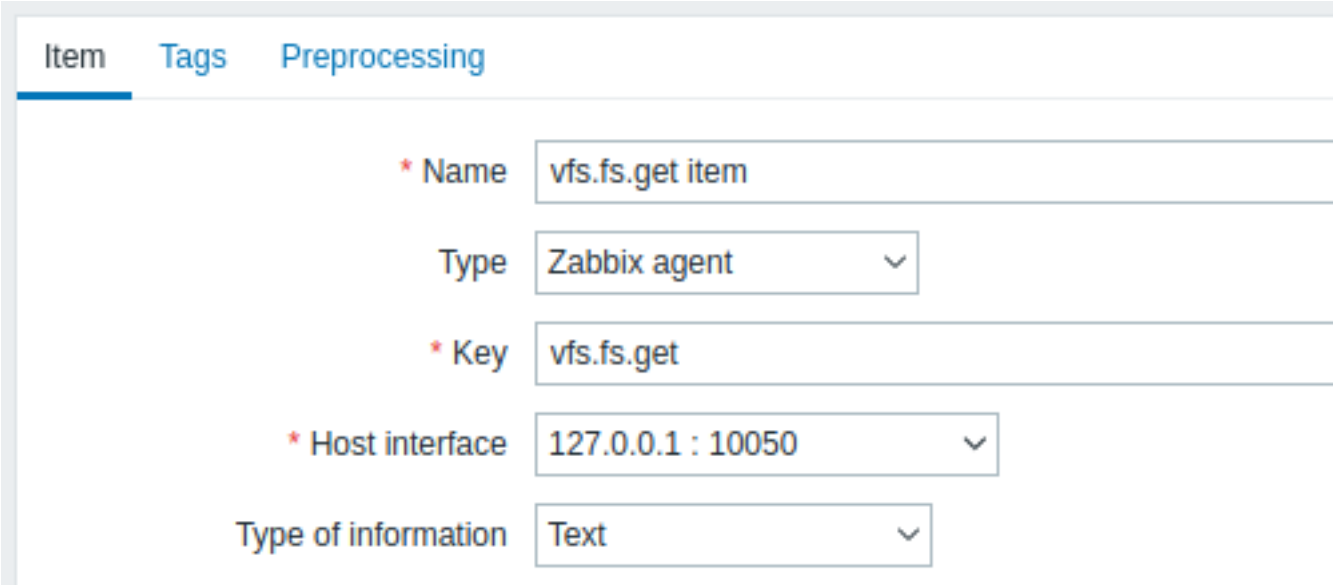
- элемента данных агента `vfs.fs.get` в качестве основного элемента данных
- зависимых от него элемента данных для правила низкоуровневого обнаружения и прототипов элементов данных

Настройка

Основной элемент данных

Создайте элемент данных агента Zabbix, используя следующий ключ:

```
vfs.fs.get
```



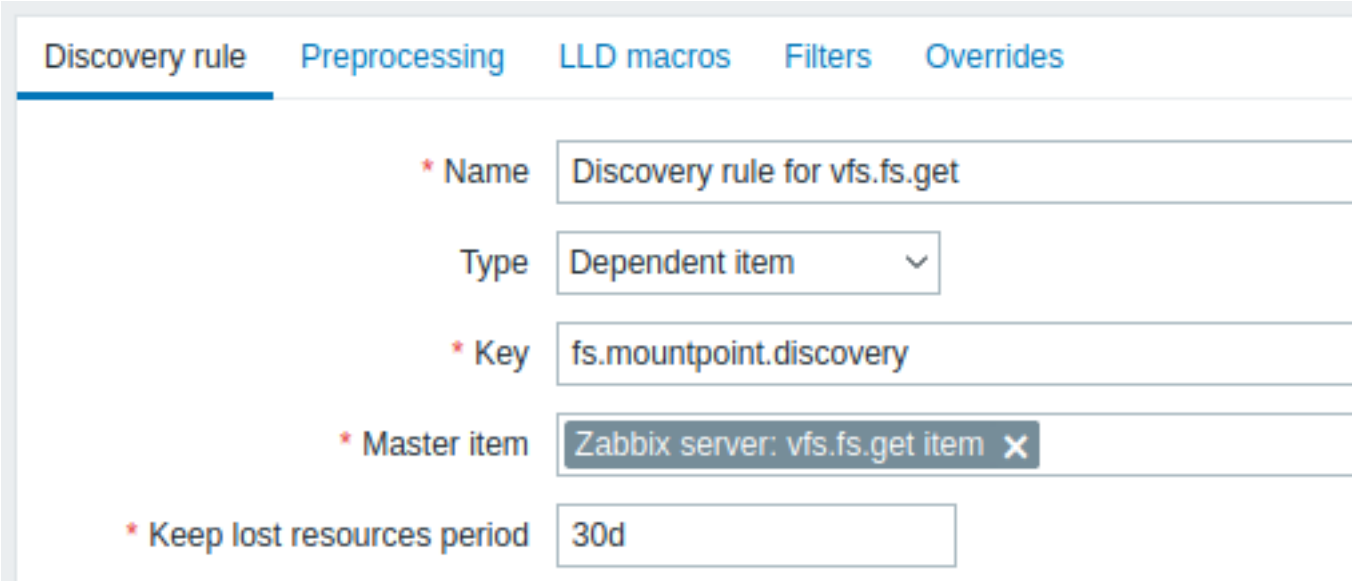

Установите тип информации в значение ”Text” для потенциально больших JSON-данных.

Данные, возвращаемые этим элементом данных, будут содержать примерно следующее для смонтированной файловой системы:

```
[
  {
    "fsname": "/",
    "fstype": "ext4",
    "bytes": {
      "total": 249405239296,
      "free": 24069537792,
      "used": 212595294208,
      "pfree": 10.170306,
      "pused": 89.829694
    },
    "inodes": {
      "total": 15532032,
      "free": 12656665,
      "used": 2875367,
      "pfree": 81.487503,
      "pused": 18.512497
    },
    "options": "rw,noatime,errors=remount-ro"
  }
]
```

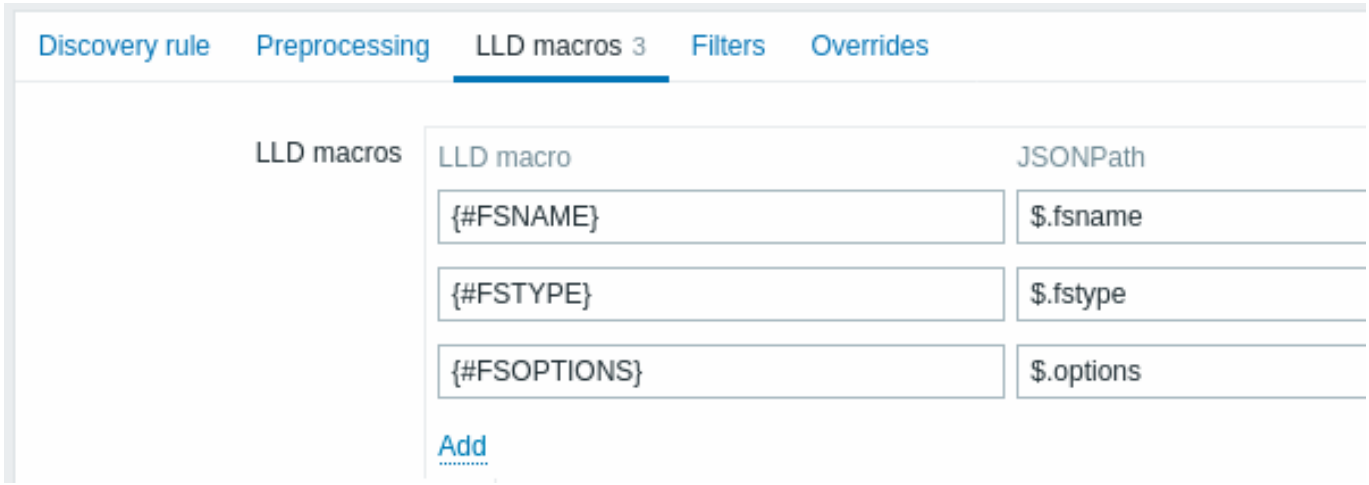
Зависимое правило LLD

Создайте правило обнаружения на низком уровне типа *Зависимый элемент данных*:

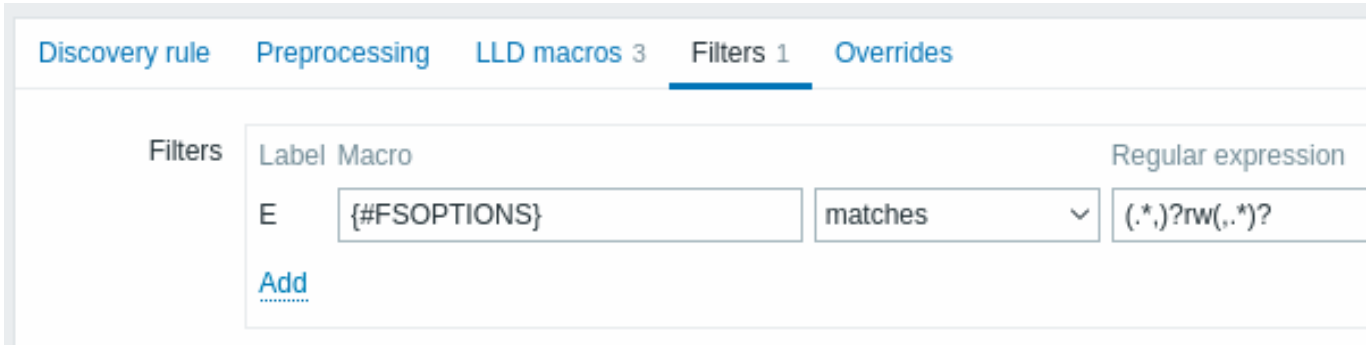



В качестве основного элемента данных выберите элемент `vfs.fs.get`, который мы создали.

На вкладке *LLD macros* задайте пользовательские макросы с соответствующим JSONPath:

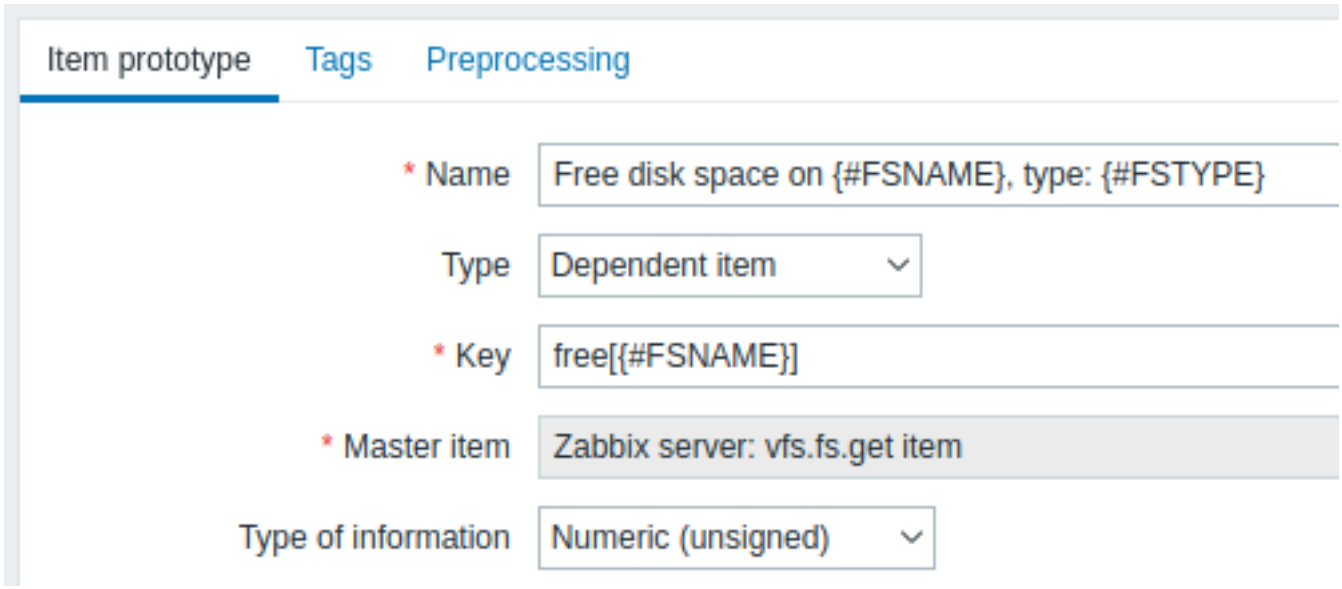



На вкладке *Filters* можно добавить регулярное выражение, которое отфильтрует только файловые системы с **чтением и записью**:




Прототип зависимого элемента данных

Создайте прототип элемента данных с типом ”Dependent item” в этом правиле LLD. В качестве мастер-элемента для этого прототипа выберите элемент `vfs.fs.get`, который мы создали.




Обратите внимание на использование пользовательских макросов в имени и ключе прототипа элемента данных:

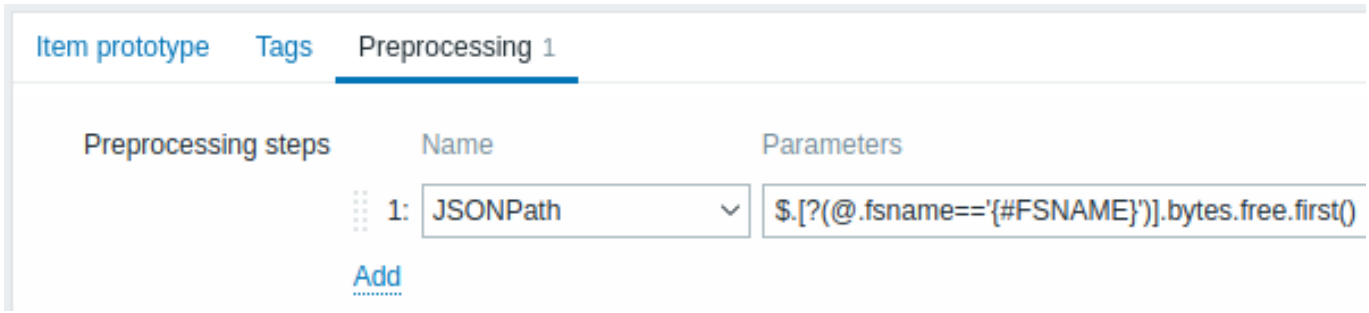
- *Name*: Free disk space on {#FSNAME}, type: {#FSTYPE}
- *Key*: Free[{#FSNAME}]

В качестве типа информации используйте:

- *Numeric (unsigned)* для метрик вроде ’free’, ’total’, ’used’
- *Numeric (float)* для метрик вроде ’pfree’, ’pused’ (percentage)

На вкладке *Preprocessing* прототипа элемента данных выберите *JSONPath* и используйте следующее выражение JSONPath в качестве параметра:

```
$.[?(@.fsname=='{#FSNAME}')].bytes.free.first()
```




Когда обнаружение начнется, для каждой точки монтирования будет создан один элемент данных. Этот элемент данных будет возвращать количество свободных байтов для указанной точки монтирования.

**2. Обнаружение сетевых интерфейсов**

Подобно тому как обнаруживаются файловые системы, можно обнаруживать и сетевые интерфейсы.

Ключ элемента данных

Ключ элемента данных, используемый для правила обнаружения, такой:

`net.if.discovery`

Поддерживаемые макросы

Вы можете использовать макрос {#IFNAME} в фильтре правила обнаружения и прототипах элементов данных, триггеров и графиков.

Примеры прототипов элементов данных, которые вы можете захотеть создать на основе «net.if.discovery»:

- ”net.if.in[{#IFNAME},bytes]”,
- ”net.if.out[{#IFNAME},bytes]”.

Обратите внимание, что на платформе Windows также возвращается {#IFGUID}.

**3. Обнаружение CPU и ядер CPU**

Подобно тому как обнаруживаются файловые системы, можно также обнаруживать CPU и ядра CPU.

Ключ элемента данных

Ключ элемента данных для использования в правиле обнаружения:

`system.cpu.discovery`

Поддерживаемые макросы

Этот ключ обнаружения возвращает два макроса — {#CPU.NUMBER} и {#CPU.STATUS}, определяющие порядковый номер CPU и статус соответственно. Он перечисляет процессоры от 0 до N - 1, где N — общее количество процессоров.

```
[
    {
        "{#CPU.NUMBER}": 0,
        "{#CPU.STATUS}": "online"
    },
    {
        "{#CPU.NUMBER}": 1,
        "{#CPU.STATUS}": "offline"
    },
    {
        "{#CPU.NUMBER}": 2,
        "{#CPU.STATUS}": "unknown" /* «unknown»       Windows */
    },
    {
        "{#CPU.NUMBER}": 3,
        "{#CPU.STATUS}": "online"
```

```
    }
]
```

Обратите внимание что фактически невозможно провести чёткое различие между физическими процессорами, ядрами и гиперпотоками. {#CPU.STATUS} в Linux, UNIX и системах BSD возвращают статус процессора, который может быть либо «online», либо «offline». В системах Windows этот же макрос может представлять третье значение — «unknown» — что указывает на то, что процессор был обнаружен, но информация о нём ещё не собрана.

Обнаружение CPU полагается на процесс сбора данных агента, чтобы оставаться в соответствии с данными, предоставленными сборщиком, и экономить ресурсы на получение данных. Это приводит к тому, что этот ключ элемента данных не работает с флагом командной строки test (-t) бинарного файла агента, который будет возвращать статус NOT_SUPPORTED и сопроводительное сообщение, указывающее, что процесс сборщика не запущен.

Прототипы элементов данных, которые могут быть созданы на основе обнаружения CPU, включают, для примера:

- system.cpu.util[{#CPU.NUMBER},< >,< >]
- system.hw.cpu[{#CPU.NUMBER},< >]

Подробное описание ключа элемента данных см. в разделе ключи элементов данных агента Zabbix.

Примечания по обнаруженным элементам данных счетчиков производительности на системах NUMA

Некоторые счетчики производительности Windows могут быть недоступны для некоторых логических процессоров в системах NUMA.

Например, элементы данных, обнаруженные с помощью прототипа элемента данных с ключом ниже, могут работать только для первого узла NUMA. Элементы данных для других узлов NUMA могут находиться в неподдерживаемом состоянии.

```
perf_counter[\Processor({#CPU.NUMBER})\% Processor Time,60]
```

Кроме того, правило обнаружения с ключом "system.cpu.discovery" не предоставляет никаких значений макросов низкоуровневого обнаружения для подстановки в такие счетчики производительности, как:

```
perf_counter["\Processor Information(<       NUMA>,<     CPU      NUMA>)\% Processor Time",60]
```

В то же время элементы данных system.cpu.util используют разные счетчики производительности для мониторинга процента загрузки CPU в зависимости от количества групп процессоров и количества процессоров. Также существует редкая ошибка, связанная с группами процессоров в Windows.

#### 4. Обнаружение SNMP OID'ов

Обзор

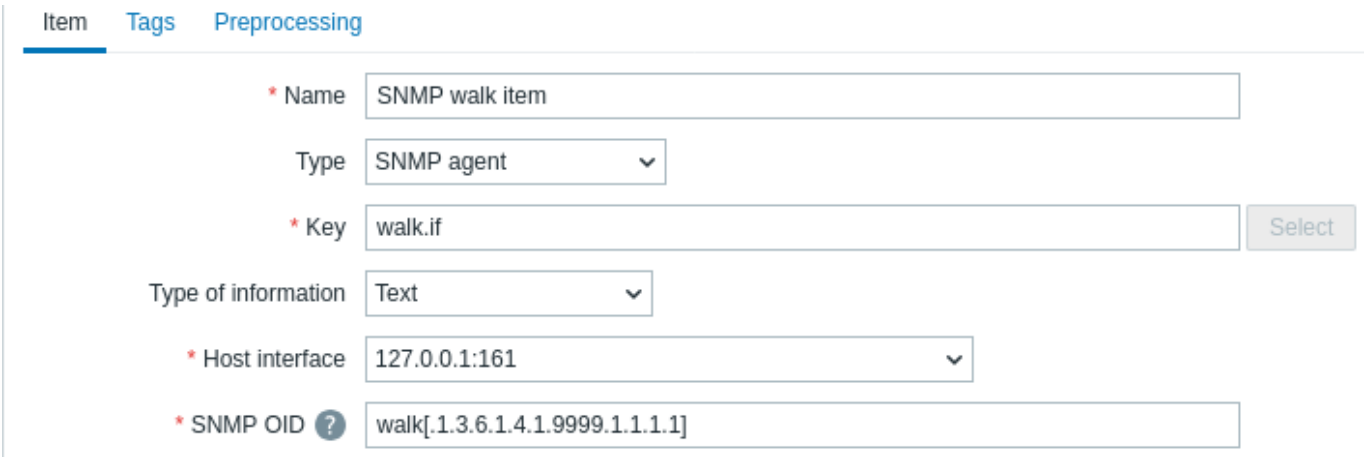
В этом разделе мы будем выполнять SNMP обнаружение на коммутаторе.

Этот метод обнаружения SNMP OID'ов поддерживается, начиная с Zabbix сервера/прокси версии 6.4.

Пример настройки

1. Создайте элемент данных с типом SNMP агент и ключом наподобие следующего:

```
walk[.1.3.6.1.4.1.9999.1.1.1.1]
```



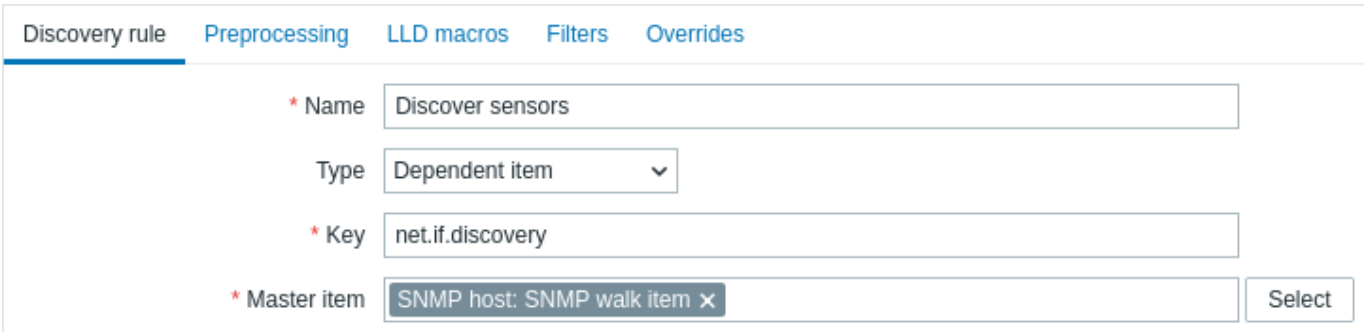

Этот элемент данных выполнит один обход таблиц SNMP и вернёт все элементы таблиц за один запрос, в формате, который соответствует выводу утилиты snmpwalk с опциями форматирования -Oe -Ot -On.

Вернётся следующее многострочное текстовое значение:

```
.1.3.6.1.4.1.9999.1.1.1.1.1.1 = STRING: "Temperature Sensor"
.1.3.6.1.4.1.9999.1.1.1.1.2.1 = STRING: "temp"
.1.3.6.1.4.1.9999.1.1.1.1.3.1 = 100
.1.3.6.1.4.1.9999.1.1.1.1.1.2 = STRING: "Humidity Sensor"
.1.3.6.1.4.1.9999.1.1.1.1.2.2 = STRING: "humidity"
.1.3.6.1.4.1.9999.1.1.1.1.3.2 = 200
```

2. Создайте правило обнаружения:

- В поле *Имя (Name)* введите наглядное имя правила обнаружения (например: «Discover sensors (Обнаружение датчиков)»).
- В поле *Тип (Type)* выберите «Зависимый элемент данных (Dependent item)».
- В поле *Ключ (Key)* введите наглядный ключ (например: «net.if.discovery»).
- В поле *Основной элемент данных (Master item)* выберите «SNMP walk item».




3. На вкладке *Предобработка (Preprocessing)* добавьте шаг предобработки «SNMP walk в JSON (SNMP walk to JSON)» в выпадающем списке *Имя (Name)* с тремя параметрами:

- *Имя поля (Field name)*: "{#SENSORNAME}"; *Префикс OID (OID prefix)*: ".1.3.6.1.4.1.9999.1.1.1.1.1": *Формат (Format)*: «Без изменений (Unchanged)».
- *Имя поля (Field name)*: "{#SENSORTYPE}"; *Префикс OID (OID prefix)*: ".1.3.6.1.4.1.9999.1.1.1.1.2": *Формат (Format)*: «Без изменений (Unchanged)».
- *Имя поля (Field name)*: "{#SENSORVALUE}"; *Префикс OID (OID prefix)*: ".1.3.6.1.4.1.9999.1.1.1.1.3": *Формат (Format)*: «Без изменений (Unchanged)».

После предобработки правило обнаружения вернёт JSON-массив наборов макросов.

Например:

```
[
    {
        "{#SNMPINDEX}": "1",
        "{#SENSORNAME}": "Temperature Sensor",
        "{#SENSORTYPE}": "temp",
        "{#SENSORVALUE}": "100"
    },
    {
        "{#SNMPINDEX}": "2",
        "{#SENSORNAME}": "Humidity Sensor",
        "{#SENSORTYPE}": "humidity",
        "{#SENSORVALUE}": "200"
    }
]
```

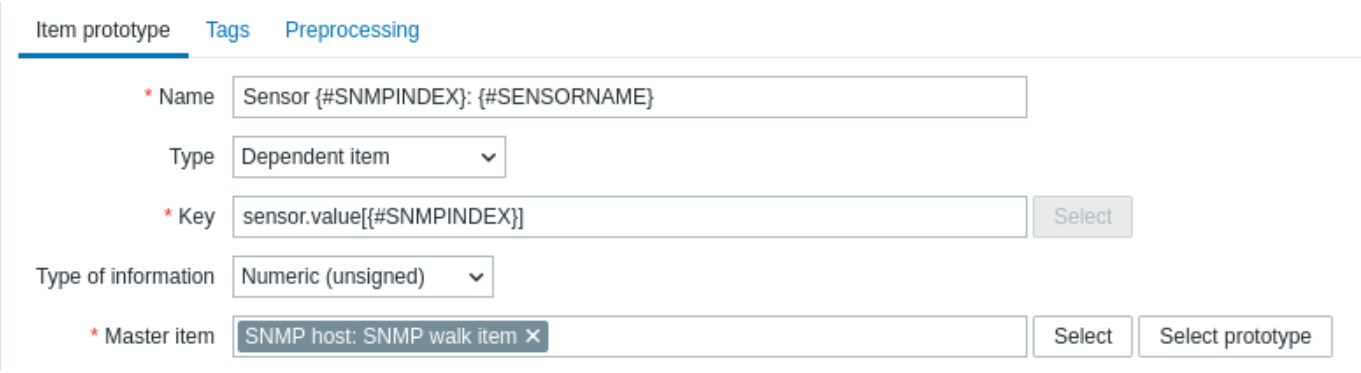
Каждый объект представляет один обнаруженный датчик и предоставляет макросы — такие, как {#SNMPINDEX}, {#SENSORNAME}, {#SENSORTYPE} и {#SENSORVALUE}.

Они сгруппированы по SNMP индексу, который является числовым суффиксом в конце каждого OID'а (например: .1, .2) — этот индекс уникально идентифицирует каждую строку в таблице SNMP и автоматически извлекается как значение макроса {#SNMPINDEX}.

4. Под правилом обнаружения создайте один или более прототипов элементов данных (с элементом данных правила обнаружения в качестве основного элемента данных).

Например, зависимый элемент данных для значения датчика:

- В поле *Имя (Name)* введите: «Sensor {#SNMPINDEX}: {#SENSORNAME}».
- В поле *Тип (Type)* выберите: «Зависимый элемент данных (Dependent item)».
- В поле *Ключ (Key)* введите: «sensor.value[{#SNMPINDEX}]».
- В поле *Основной элемент данных (Master item)* выберите: «SNMP walk item».




На вкладке *Предобработка (Preprocessing)* добавьте шаг предобработки «Значение SNMP walk (SNMP walk value)» в поле имени с OID'ом «.1.3.6.1.4.1.9999.1.1.1.1.3.{#SNMPINDEX}» в поле *Параметр (Parameter)*. *Формат (Format)*: «Без изменений (Unchanged)».

Будут обнаружены следующие элементы данных:

| Имя | Ключ | OID, из которого извлечено значение | Значение элемента данных |
|---|---|---|---|
| Sensor 1: Temperature Sensor | sensor.value[1] | .1.3.6.1.4.1.9999.1.1.1.1.3.1 | 100 |
| Sensor 2: Humidity Sensor | sensor.value[2] | .1.3.6.1.4.1.9999.1.1.1.1.3.2 | 200 |

При отрабатывании правила обнаружения создаются элементы данных — такие, как `sensor.value[1]`, `sensor.value[2]`.

Каждый зависимый элемент данных с помощью предобработки извлекает своё значение из результата SNMP walk основного элемента данных, без выполнения отдельных самостоятельных SNMP запросов.

5. В прототипах триггеров ссылайтесь на прототипы зависимых элементов данных, используя те же макросы из правила обнаружения. Пример:

```
{Template_Sensor:sensor.value[{#SNMPINDEX}].last()} > 75
```

Это создаёт триггер для каждого обнаруженного датчика (например, sensor.value[1], sensor.value[2]), срабатывающий, если последнее значение (температура или влажность) превышает 75.

6. Включайте зависимые элементы данных для каждого обнаруженного объекта. Пример ключа элемента данных для графика:

```
sensor.value[{#SNMPINDEX}]
```

Создаётся по графику на каждый `{#SNMPINDEX}`, выстраивая график температуры и влажности с течением времени.

Такая настройка выполняет только один запрос SNMP walk на цикл опроса, независимо от количества обнаруженных элементов данных. Все зависимые элементы данных извлекают свои значения из результата основного SNMP walk с помощью предобработки, значительно уменьшая SNMP трафик и нагрузку.

Динамические индексы с walk[]

Динамические индексы (например, индексы интерфейсов) могут сдвигаться при изменениях конфигурации оборудования. Чтобы приспособиться к такому поведению, основное правило обнаружения SNMP walk создаётся с ключом таким как:

```
walk[1.3.6.1.2.1.2.2.1.10]
```

После предобработки SNMP walk в JSON результат может быть наподобие:

```
[
    {
        "{#SNMPINDEX}": "2",
        "{#VALUE}": "123456"
    },
    {
        "{#SNMPINDEX}": "3",
        "{#VALUE}": "654321"
    }
]
```

Прототип зависимого элемента данных использует макрос {#SNMPINDEX}, чтобы сконструировать свой ключ:

```
net.if.in[{#SNMPINDEX}]
```

Предобработка для этого прототипа включает шаг «Значение SNMP walk (SNMP walk value» в поле имени с OID'ом «1.3.6.1.2.1.2.2.1.10.{#SNMPINDEX}» в поле *Параметр (Parameter)*. *Формат (Format)*: «Без изменений (Unchanged)».

Во время выполнения создаются реальные элементы данных — такие, как `net.if.in[2]` и `net.if.in[3]`. Если индекс данного интерфейса меняется (например, если индекс 2 заменяется в таблице SNMP на 5), то при следующем срабатывании правила обнаружения:

- Старый зависимый элемент данных net.if.in[2] помечается как «потерянный (lost)» или удаляется, и никакие новые данные для него не собираются.
- Создаётся новый элемент данных net.if.in[5], с пустой историей.
- Данные истории от net.if.in[2] автоматически не переносятся к net.if.in[5].

Пример прототипа триггера:

```
{Template_Interface:net.if.in[{#SNMPINDEX}].last()} > 1000000000
```

Пример прототипа графика включает элементы данных:

```
net.if.in[{#SNMPINDEX}]
net.if.out[{#SNMPINDEX}]
```

Такая настройка обеспечивает надёжность мониторинга таблиц с динамическими индексами, в то же время минимизируя SNMP трафик, — требуется только обход SNMP на цикл опроса, с прототипами зависимых элементов данных, извлекающих необходимые значения.

Обнаруженные объекты

При работе сервера он будет создавать реальные зависимые элементы данных, триггеры и графики на основе значений, возвращаемых правилом обнаружения SNMP.

#### 5. Обнаружение SNMP OID'ов (прежнее)

Обзор

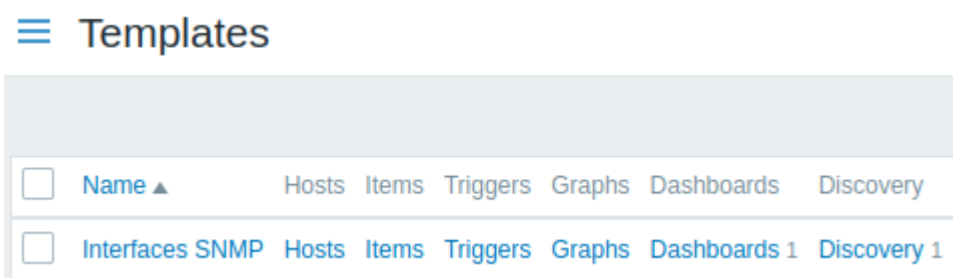
В этом разделе мы будем выполнять SNMP обнаружение на свитче.

Ключ элемента данных

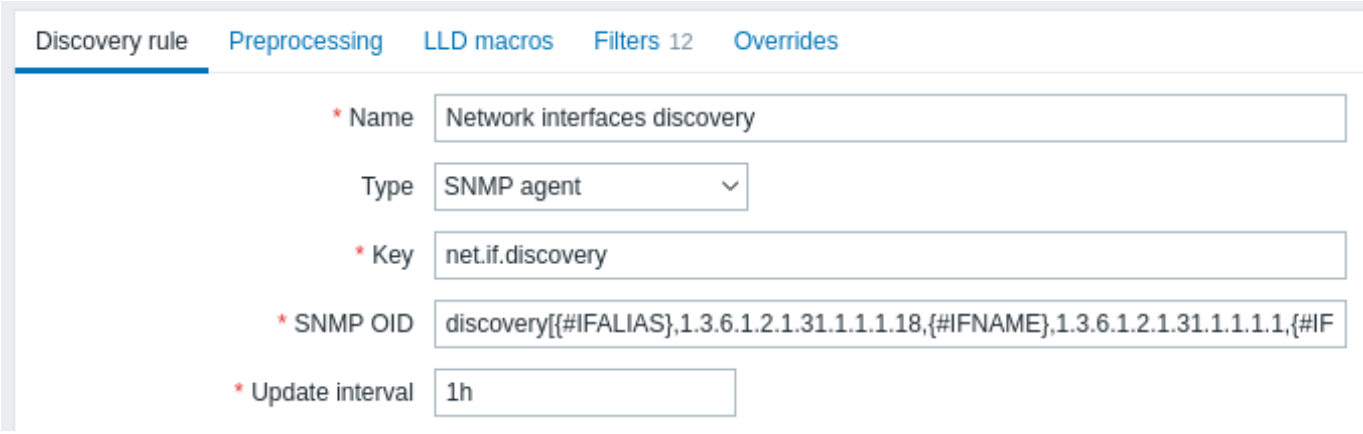
В отличие от обнаружения файловой системы и сетевого интерфейса, элемент данных не обязательно должен иметь ключ "snmp.discovery" - достаточно, чтобы типом элемента данных был SNMP agent.

Чтобы настроить правило обнаружения, выполните следующие действия:

- Перейдите в: *Сбор данных → Шаблоны*
- Нажмите *Обнаружение* в строке нужного шаблона



- Нажмите *Создать правило обнаружения* в правом верхнем углу экрана
- Заполните форму правила обнаружения, указав необходимые сведения, как показано на снимке экрана ниже




Все обязательные поля ввода отмечены красной звездочкой.

OID, которые нужно обнаружить, задаются в поле SNMP OID в следующем формате: `discovery[{#MACRO1}, oid1, {#MACRO2}, oid2, …,]`

где *{#MACRO1}*, *{#MACRO2}* ... - это допустимые имена lld macro, а *oid1*, *oid2*... - это OID, способные возвращать значимые значения для этих макросов. Встроенный макрос *{#SNMPINDEX}*, содержащий индекс обнаруженного OID, применяется к обнаруженным сущностям. Обнаруженные сущности группируются по значению макроса *{#SNMPINDEX}*.

Ниже приведен пример, в котором на коммутаторе выполняется несколько snmpwalk:

```
$ snmpwalk -v 2c -c public 192.168.1.1 IF-MIB::ifDescr
IF-MIB::ifDescr.1 = STRING: WAN
IF-MIB::ifDescr.2 = STRING: LAN1
IF-MIB::ifDescr.3 = STRING: LAN2

$ snmpwalk -v 2c -c public 192.168.1.1 IF-MIB::ifPhysAddress
IF-MIB::ifPhysAddress.1 = STRING: 8:0:27:90:7a:75
IF-MIB::ifPhysAddress.2 = STRING: 8:0:27:90:7a:76
IF-MIB::ifPhysAddress.3 = STRING: 8:0:27:2b:af:9e
```

А SNMP OID задан как: `discovery[{#IFDESCR}, ifDescr, {#IFPHYSADDRESS}, ifPhysAddress]`

Теперь это правило обнаружит сущности, для которых макросы {#IFDESCR} имеют значения **WAN**, **LAN1** и **LAN2**, макросы {#IFPHYSADDRESS} - **8:0:27:90:7a:75**, **8:0:27:90:7a:76** и **8:0:27:2b:af:9e**, а макросы {#SNMPINDEX} - индексы обнаруженных OID **1**, **2** и **3**:

```
[
    {
        "{#SNMPINDEX}": "1",
        "{#IFDESCR}": "WAN",
        "{#IFPHYSADDRESS}": "8:0:27:90:7a:75"
    },
    {
        "{#SNMPINDEX}": "2",
        "{#IFDESCR}": "LAN1",
        "{#IFPHYSADDRESS}": "8:0:27:90:7a:76"
    },
    {
        "{#SNMPINDEX}": "3",
        "{#IFDESCR}": "LAN2",
        "{#IFPHYSADDRESS}": "8:0:27:2b:af:9e"
    }
]
```

Если у сущности нет указанного OID, соответствующий макрос для этой сущности будет опущен. Например, если у нас есть следующие данные:

```
ifDescr.1 "Interface #1"
ifDescr.2 "Interface #2"
ifDescr.4 "Interface #4"
```

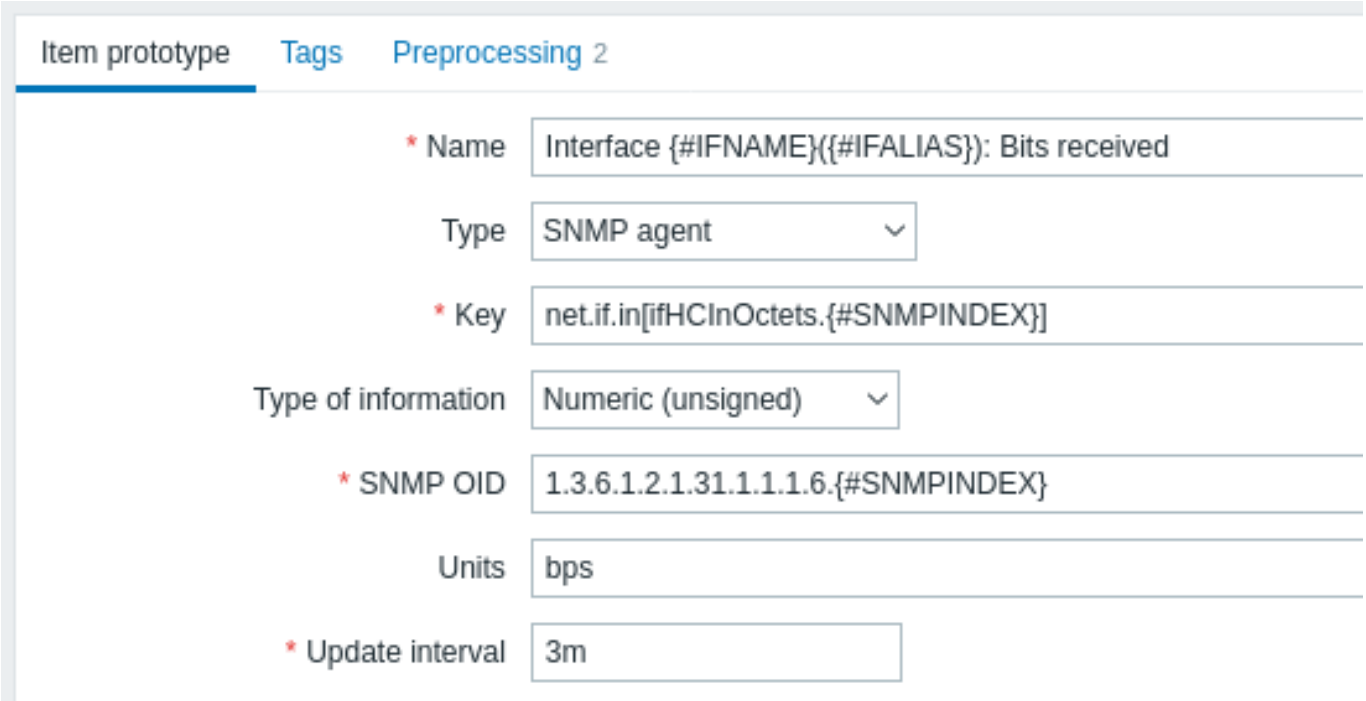
```
ifAlias.1 "eth0"
ifAlias.2 "eth1"
ifAlias.3 "eth2"
ifAlias.5 "eth4"
```

Тогда в этом случае SNMP discovery `discovery[{#IFDESCR}, ifDescr, {#IFALIAS}, ifAlias]` вернет следующую структуру:

```
[
    {
        "{#SNMPINDEX}": 1,
        "{#IFDESCR}": "Interface #1",
        "{#IFALIAS}": "eth0"
    },
    {
        "{#SNMPINDEX}": 2,
        "{#IFDESCR}": "Interface #2",
        "{#IFALIAS}": "eth1"
    },
    {
        "{#SNMPINDEX}": 3,
        "{#IFALIAS}": "eth2"
    },
    {
        "{#SNMPINDEX}": 4,
        "{#IFDESCR}": "Interface #4"
    },
    {
        "{#SNMPINDEX}": 5,
        "{#IFALIAS}": "eth4"
    }
]
```

Прототипы элементов данных

Следующий снимок экрана иллюстрирует, как мы можем использовать эти макросы в прототипах элементов данных:




Мы можем создать столько прототипов элементов данных, сколько нам необходимо:

## Item prototypes

All templates / Linux SNMP  Discovery list / Network interfaces discovery  Item prototypes 9  Trigger prototypes 4  Graph prototypes 1  Host prototypes

| Name ▲ | Key | Interval | History | Trends | Type | Create enabled |
|---|---|---|---|---|---|---|
| Interface {#IFNAME}({#IFALIAS}): Bits received | net.if.in[ifHCInOctets.{#SNMPINDEX}] | 3m | 7d | 365d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Bits sent | net.if.out[ifHCOutOctets.{#SNMPINDEX}] | 3m | 7d | 365d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Inbound packets discarded | net.if.in.discards[ifInDiscards.{#SNMPINDEX}] | 3m | 7d | 365d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Inbound packets with errors | net.if.in.errors[ifInErrors.{#SNMPINDEX}] | 3m | 7d | 365d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Interface type | net.if.type[ifType.{#SNMPINDEX}] | 1h | 7d | 0d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Operational status | net.if.status[ifOperStatus.{#SNMPINDEX}] | 1m | 7d | 0 | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Outbound packets discarded | net.if.out.discards[ifOutDiscards.{#SNMPINDEX}] | 3m | 7d | 365d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Outbound packets with errors | net.if.out.errors[ifOutErrors.{#SNMPINDEX}] | 3m | 7d | 365d | SNMP agent | Yes |
| Interface {#IFNAME}({#IFALIAS}): Speed | net.if.speed[ifHighSpeed.{#SNMPINDEX}] | 5m | 7d | 0d | SNMP agent | Yes |

Прототипы триггеров

Следующий снимок экрана иллюстрирует, как мы можем использовать эти макросы в прототипах триггеров:

Trigger prototype  Tags  Dependencies

* Name: Interface {#IFNAME}({#IFALIAS}): Link down

Event name: Interface {#IFNAME}({#IFALIAS}): Link down

Operational data: Current state: {ITEM.LASTVALUE1}

Severity: Not classified | Information | Warning | Average | High | Disaster

* Problem expression:
```
{$IFCONTROL:"{#IFNAME}"}=1 and last(/SNMP
host/net.if.status[ifOperStatus.{#SNMPINDEX}])=2 and
(last(/SNMP host/net.if.status[ifOperStatus.
{#SNMPINDEX}],#1)<>last(/SNMP
host/net.if.status[ifOperStatus.{#SNMPINDEX}],#2))
```
Expression constructor

OK event generation: Expression | Recovery expression | None

* Recovery expression:
```
last(/SNMP host/net.if.status[ifOperStatus.{#SNMPINDEX}])<>2
or {$IFCONTROL:"{#IFNAME}"}=0
```

Trigger prototypes

All templates / Linux SNMP  Discovery list / Network interfaces discovery  Item prototypes 9  Trigger prototypes 4  Graph prototypes 1  Host prototypes

| Severity | Name ▲ | Operational data | Expression | Create enabled |
|---|---|---|---|---|
| Information | Interface {#IFNAME}({#IFALIAS}): Ethernet has changed to lower speed than it was before<br>**Depends on:**<br>Linux SNMP: Interface {#IFNAME}({#IFALIAS}): Link down | Current reported speed: {ITEM.LASTVALUE1} | Problem: **change**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])<0 and **last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>0 and ( **last**(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=6 or **last**(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=7 or **last**(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=11 or **last**(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=62 or **last**(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=69 or **last**(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=117 ) and (**last**(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])<>2)<br>Recovery: (**change**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>0 and **last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}],#2)>0) or (**last**(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])=2) | Yes |
| Warning | Interface {#IFNAME}({#IFALIAS}): High bandwidth usage<br>**Depends on:**<br>Linux SNMP: Interface {#IFNAME}({#IFALIAS}): Link down | In: {ITEM.LASTVALUE1}, out: {ITEM.LASTVALUE3}, speed: {ITEM.LASTVALUE2} | Problem: (**avg**(/Linux SNMP/net.if.in[ifHCInOctets.{#SNMPINDEX}],15m)>({$IF.UTIL.MAX:"{#IFNAME}"}/100)***last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}]) or **avg**(/Linux SNMP/net.if.out[ifHCOutOctets.{#SNMPINDEX}],15m)>({$IF.UTIL.MAX:"{#IFNAME}"}/100)***last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])) and **last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>0<br>Recovery: **avg**(/Linux SNMP/net.if.in[ifHCInOctets.{#SNMPINDEX}],15m)<(({$IF.UTIL.MAX:"{#IFNAME}"}-3)/100)***last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}]) and **avg**(/Linux SNMP/net.if.out[ifHCOutOctets.{#SNMPINDEX}],15m)<(({$IF.UTIL.MAX:"{#IFNAME}"}-3)/100)***last**(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}]) | Yes |
| Warning | Interface {#IFNAME}({#IFALIAS}): High error rate<br>**Depends on:**<br>Linux SNMP: Interface {#IFNAME}({#IFALIAS}): Link down | errors in: {ITEM.LASTVALUE1}, errors out: {ITEM.LASTVALUE2} | Problem: **min**(/Linux SNMP/net.if.in.errors[ifInErrors.{#SNMPINDEX}],5m)>{$IF.ERRORS.WARN:"{#IFNAME}"} or **min**(/Linux SNMP/net.if.out.errors[ifOutErrors.{#SNMPINDEX}],5m)>{$IF.ERRORS.WARN:"{#IFNAME}"}<br>Recovery: **max**(/Linux SNMP/net.if.in.errors[ifInErrors.{#SNMPINDEX}],5m)<{$IF.ERRORS.WARN:"{#IFNAME}"}*0.8 and **max**(/Linux SNMP/net.if.out.errors[ifOutErrors.{#SNMPINDEX}],5m)<{$IF.ERRORS.WARN:"{#IFNAME}"}*0.8 | Yes |
| Average | Interface {#IFNAME}({#IFALIAS}): Link down | Current state: {ITEM.LASTVALUE1} | Problem: {$IFCONTROL:"{#IFNAME}"}=1 and **last**(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}],#1)<>**last**(/Linux SNMP/ne | Yes |

Прототипы графиков

Следующий снимок экрана иллюстрирует, как мы можем использовать эти макросы в прототипах графиков:

Graph prototype  Preview

* Name: Interface {#IFNAME}({#IFALIAS}): Network traffic
* Width: 900
* Height: 200
Graph type: Normal
Show legend: ☑
Show working time: ☑
Show triggers: ☑
Percentile line (left): ☐
Percentile line (right): ☐
Y axis MIN value: Calculated
Y axis MAX value: Calculated

* Items

| | Name | Function | Draw style | Y axis side | Color |
|---|---|---|---|---|---|
| 1: | SNMP host: Interface {#IFNAME}({#IFALIAS}): Bits received | avg | Gradient line | Left | |
| 2: | SNMP host: Interface {#IFNAME}({#IFALIAS}): Bits sent | avg | Bold line | Left | |
| 3: | SNMP host: Interface {#IFNAME}({#IFALIAS}): Outbound packets with errors | avg | Line | Right | |
| 4: | SNMP host: Interface {#IFNAME}({#IFALIAS}): Inbound packets with errors | avg | Line | Right | |
| 5: | SNMP host: Interface {#IFNAME}({#IFALIAS}): Outbound packets discarded | avg | Line | Right | |
| 6: | SNMP host: Interface {#IFNAME}({#IFALIAS}): Inbound packets discarded | avg | Line | Right | |

Add  Add prototype

## Graph prototypes

All templates / Linux SNMP | Discovery list / Network interfaces discovery | Item prototypes 9 | Trigger prototypes 4 | Graph prototypes 1 | Host prototypes

| Name ▲ | Width | Height |
|---|---|---|
| Interface {#IFNAME}({#IFALIAS}): Network traffic | 900 | 200 |

Общий вид нашего правила обнаружения:

All templates / Linux SNMP | Items 26 | Triggers 10 | Graphs 5 | Dashboards 2 | Discovery rules 5 | Web scenarios

| Template | Name ▲ | Items | Triggers | Graphs |
|---|---|---|---|---|
| Linux SNMP | Network interfaces discovery | Item prototypes 9 | Trigger prototypes 4 | Graph prototypes 1 |

Обнаруженные объекты

Во время своей работы сервер создаст реальные элементы данных, триггеры и графики на основе значений, которые вернёт правило SNMP обнаружения. В настройках узла сети они будут снабжены префиксом золотистого цвета, который является ссылкой на породившее их правило обнаружения.

## Items

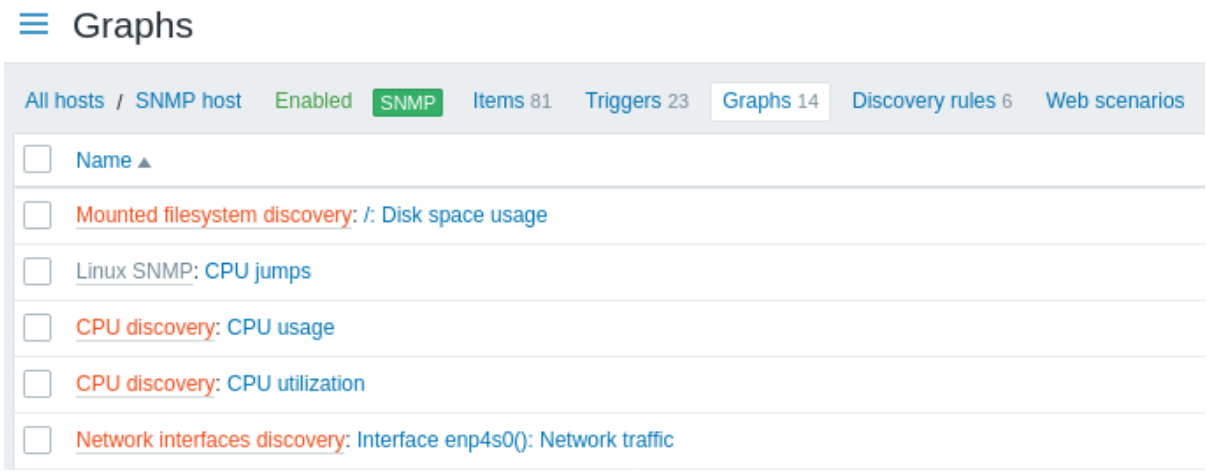
All hosts / SNMP host | Enabled | SNMP | Items 81 | Triggers 23 | Graphs 14 | Discovery rules 6 | Web scenarios

| Name ▲ | Triggers | Key | Interval | History | Trends | Type | Status |
|---|---|---|---|---|---|---|---|
| Network interfaces discovery: Interface enp4s0(): Bits received | Triggers 1 | net.if.in[ifHCInOctets.2] | 3m | 7d | 365d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Bits sent | Triggers 1 | net.if.out[ifHCOutOctets.2] | 3m | 7d | 365d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Inbound packets discarded | | net.if.in.discards[ifInDiscards.2] | 3m | 7d | 365d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Inbound packets with errors | Triggers 1 | net.if.in.errors[ifInErrors.2] | 3m | 7d | 365d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Interface type | Triggers 1 | net.if.type[ifType.2] | 1h | 7d | 0d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Operational status | Triggers 2 | net.if.status[ifOperStatus.2] | 1m | 7d | 0 | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Outbound packets discarded | | net.if.out.discards[ifOutDiscards.2] | 3m | 7d | 365d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Outbound packets with errors | Triggers 1 | net.if.out.errors[ifOutErrors.2] | 3m | 7d | 365d | SNMP agent | Enabled |
| Network interfaces discovery: Interface enp4s0(): Speed | Triggers 2 | net.if.speed[ifHighSpeed.2] | 5m | 7d | 0d | SNMP agent | Enabled |

## Triggers

All hosts / SNMP host | Enabled | SNMP | Items 81 | Triggers 23 | Graphs 14 | Discovery rules 6 | Web scenarios

| Severity | Value | Name ▲ | Operational data | Expression |
|---|---|---|---|---|
| Information | OK | Network interfaces discovery: Interface enp4s0(): Ethernet has changed to lower speed than it was before<br>**Depends on:**<br>SNMP host: Interface enp4s0(): Link down | Current reported speed: {ITEM.LASTVALUE1} | Problem: **change**(/SNMP host/net.if.speed[ifHighSpeed.2])<0 and **last**(/SNMP host/net.if.speed[ifHighSpeed.2])>0 and ( **last**(/SNMP host/net.if.type[ifType.2])=6 or **last**(/SNMP host/net.if.type[ifType.2])=7 or **last**(/SNMP host/net.if.type[ifType.2])=11 or **last**(/SNMP host/net.if.type[ifType.2])=62 or **last**(/SNMP host/net.if.type[ifType.2])=69 or **last**(/SNMP host/net.if.type[ifType.2])=117 ) and (**last**(/SNMP host/net.if.status[ifOperStatus.2])<>2)<br>Recovery: (**change**(/SNMP host/net.if.speed[ifHighSpeed.2])>0 and **last**(/SNMP host/net.if.speed[ifHighSpeed.2],#2)>0) or (**last**(/SNMP host/net.if.status[ifOperStatus.2])=2) |
| Warning | OK | Network interfaces discovery: Interface enp4s0(): High bandwidth usage<br>**Depends on:**<br>SNMP host: Interface enp4s0(): Link down | In: {ITEM.LASTVALUE1}, out: {ITEM.LASTVALUE3}, speed: {ITEM.LASTVALUE2} | Problem: (**avg**(/SNMP host/net.if.in[ifHCInOctets.2],15m)>({$IF.UTIL.MAX:"enp4s0"}/100)***last**(/SNMP host/net.if.speed[ifHighSpeed.2]) or **avg**(/SNMP host/net.if.out[ifHCOutOctets.2],15m)>({$IF.UTIL.MAX:"enp4s0"}/100)***last**(/SNMP host/net.if.speed[ifHighSpeed.2])) and **last**(/SNMP host/net.if.speed[ifHighSpeed.2])>0<br>Recovery: **avg**(/SNMP host/net.if.in[ifHCInOctets.2],15m)<(({$IF.UTIL.MAX:"enp4s0"}-3)/100)***last**(/SNMP host/net.if.speed[ifHighSpeed.2]) and **avg**(/SNMP host/net.if.out[ifHCOutOctets.2],15m)<(({$IF.UTIL.MAX:"enp4s0"}-3)/100)***last**(/SNMP host/net.if.speed[ifHighSpeed.2]) |
| Warning | OK | Network interfaces discovery: Interface enp4s0(): High error rate<br>**Depends on:**<br>SNMP host: Interface enp4s0(): Link down | errors in: {ITEM.LASTVALUE1}, errors out: {ITEM.LASTVALUE2} | Problem: **min**(/SNMP host/net.if.in.errors[ifInErrors.2],5m)>{$IF.ERRORS.WARN:"enp4s0"} or **min**(/SNMP host/net.if.out.errors[ifOutErrors.2],5m)>{$IF.ERRORS.WARN:"enp4s0"}<br>Recovery: **max**(/SNMP host/net.if.in.errors[ifInErrors.2],5m)<{$IF.ERRORS.WARN:"enp4s0"}*0.8 and **max**(/SNMP host/net.if.out.errors[ifOutErrors.2],5m)<{$IF.ERRORS.WARN:"enp4s0"}*0.8 |
| Average | OK | Network interfaces discovery: Interface enp4s0(): Link down | Current state: {ITEM.LASTVALUE1} | Problem: {$IFCONTROL:"enp4s0"}=1 and **last**(/SNMP host/net.if.status[ifOperStatus.2])=2 and (**last**(/SNMP host/net.if.status[ifOperStatus.2],#1)<>**last**(/SNMP host/net.if.status[ifOperStatus.2],#2))<br>Recovery: **last**(/SNMP host/net.if.status[ifOperStatus.2])<>2 or {$IFCONTROL:"enp4s0"}=0 |




### 6. Обнаружение JMX объектов

Обзор

Существует возможность обнаруживать все JMX MBean-ы или атрибуты MBean, либо указывать шаблон для обнаружения этих объектов.

Для настройки правила обнаружения обязательно нужно понимать различие между объектами MBean и атрибутами MBean. MBean — это объект, который может представлять устройство, приложение или любой другой ресурс, требующий наблюдения.

Например, существует MBean, представляющий веб-сервер. Его атрибутами являются количество соединений, количество потоков, тайм-аут запроса, файловый кэш http, использование памяти и т.д. Выражая эту мысль понятным человеческим языком, мы можем определить кофеварку как MBean, имеющий следующие атрибуты для мониторинга: количество воды на чашку, средний расход воды за определённый период времени, количество кофейных зёрен, необходимое на чашку, время перезаправки кофейными зёрнами и водой и т. д.

Ключ элемента данных

В настройках правила обнаружения, в поле *Тип* выберите **JMX агент**.

Для обнаружения JMX объектов поддерживаются два ключа элементов данных — jmx.discovery[] и jmx.get[]:

| **Ключ элемента данных** | | |
|---|---|---|
| Возвращаемое значение | Параметры | Комментарий |
| **jmx.discovery**[<режим обнаружения>,<имя объекта>,<уникальное короткое описание>] | | |

| Ключ элемента данных | | | |
|---|---|---|---|
| | Этот элемент данных возвращает массив JSON с LLD макросами, описывающими объекты MBean или их атрибуты. | **режим обнаружения** — один из следующих: *attributes* (вернуть атрибуты JMX MBean, по умолчанию) или *beans* (вернуть объекты JMX MBean)<br>**имя объекта** — шаблон имени объекта (см. документацию [en]), идентифицирующий имена MBean-ов, которые надо вернуть (по умолчанию пусто, вернёт все зарегистрированные bean-ы)<br>**уникальное короткое описание** — уникальное описание, позволяющее на том же узле сети задать несколько элементов данных JMX с тем же режимом обнаружения и именем объекта (опционально) | Примеры:<br>→ jmx.discovery — вернуть все атрибуты JMX MBean-ов<br>→ jmx.discovery[beans] — вернуть все JMX MBean-ы<br>→ jmx.discovery[attributes,"*:type=GarbageCollector,name — вернуть все атрибуты сборщика мусора (garbage collector-а)<br>→ jmx.discovery[beans,"*:type=GarbageCollector,name=* — вернуть все сборщики мустора<br><br>Имеются некоторые ограничения относительно того, какие свойства MBean-ов этот элемент данных может вернуть, вызванные ограниченным набором символов, который поддерживается при генерации имён макросов (поддерживаемые символы могут быть описаны следующим регулярным выражением: «A-Z0-9_\.»). Поэтому для обнаружения, скажем, свойств MBean-ов, содержащих слова через дефис (знак минус) или не-ASCII символы, вам нужно использовать `jmx.get[]`. |
| **jmx.get**[<режим обнаружения>,<имя объекта>,<уникальное короткое описание>] | | | |
| | Этот элемент данных возвращает массив JSON с объектами MBean или их атрибутами.<br><br>По сравнению с `jmx.discovery[]`, он не определяет LLD макросы. | **режим обнаружения** — один из следующих: *attributes* (вернуть атрибуты JMX MBean, по умолчанию) или *beans* (вернуть объекты JMX MBean)<br>**имя объекта** — шаблон имени объекта (см. документацию [en]), идентифицирующий имена MBean-ов, которые надо вернуть (по умолчанию пусто, вернёт все зарегистрированные bean-ы)<br>**уникальное короткое описание** — уникальное описание, позволяющее на том же узле сети задать несколько элементов данных JMX с тем же режимом обнаружения и именем объекта (опционально) | При использовании этого элемента данных необходимо определить настраиваемые макросы низкоуровневого обнаружения, указывающие на значения, извлекаемые из возвращаемого JSON-а с помощью JSONPath. |

**Attention:**
Если не передано никаких параметров, из JMX запрашиваются все атрибуты MBean-ов. Отсутствие параметров для JMX обнаружения или попытка получить все атрибуты для широкого диапазона наподобие `*:type=*,name=*` может привести к потенциальным проблемам с производительностью.

Использование jmx.discovery

Этот элемент данных возвращает объект JSON с макросами низкоуровневого обнаружения, описывающими объекты MBean или их атрибуты. Например, при обнаружении атрибутов MBean attributes (отформатировано для наглядности):

```
[
    {
        "{#JMXVALUE}":"0",
        "{#JMXTYPE}":"java.lang.Long",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,CollectionCount",
        "{#JMXATTR}":"CollectionCount"
    },
    {
        "{#JMXVALUE}":"0",
        "{#JMXTYPE}":"java.lang.Long",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,CollectionTime",
        "{#JMXATTR}":"CollectionTime"
    },
    {
        "{#JMXVALUE}":"true",
        "{#JMXTYPE}":"java.lang.Boolean",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,Valid",
        "{#JMXATTR}":"Valid"
    },
    {
        "{#JMXVALUE}":"PS Scavenge",
        "{#JMXTYPE}":"java.lang.String",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,Name",
        "{#JMXATTR}":"Name"
    },
    {
        "{#JMXVALUE}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXTYPE}":"javax.management.ObjectName",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXDESC}":"java.lang:type=GarbageCollector,name=PS Scavenge,ObjectName",
        "{#JMXATTR}":"ObjectName"
    }
]
```

При обнаружении объектов MBeans (отформатировано для наглядности):

```
[
    {
        "{#JMXDOMAIN}":"java.lang",
        "{#JMXTYPE}":"GarbageCollector",
        "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
        "{#JMXNAME}":"PS Scavenge"
    }
]
```

Поддерживаемые макросы

Для использования в фильтре правил обнаружения и прототипах элементов данных, триггеров и графиков поддерживаются следующие макросы :

| Макрос | Описание |
|---|---|
| Обнаружение атрибутов MBean | |
| {#JMXVALUE} | Значение атрибута. |
| {#JMXTYPE} | Тип атрибута. |
| {#JMXOBJ} | Имя объекта. |
| {#JMXDESC} | Имя объекта вместе с именем атрибута. |

| Макрос | Описание |
|---|---|
| {#JMXATTR} | Имя атрибута. |
| Обнаружение объектов MBeans | |
| {#JMXDOMAIN} | Домен MBean. (*зарезервированное имя Zabbix*) |
| {#JMXOBJ} | Имя объекта. (*зарезервированное имя Zabbix*) |
| {#JMX<свойство ключа>} | Свойства MBean (наподобие {#JMXTYPE}, {#JMXNAME}) (смотрите Ограничения ниже). |

Ограничения

Имеются некоторые ограничения, связанные с алгоритмом создания имён LLD макросов из имён свойств MBean:

- имена атрибутов приводятся к верхнему регистру
- имена атрибутов игнорируются (LLD макросы не генерируются), если они состоят из символов, не поддерживаемых для имён LLD макросов. Поддерживаемые символы могут быть описаны следующим регулярным выражением: `A-Z0-9_\.`
- если атрибуты называются «obj» или «domain», они будут проигнорированы, т.к. перекрываются со значениями зарезервированных в Zabbix свойств {#JMXOBJ} и {#JMXDOMAIN}

Пожалуйста, рассмотрите данный пример jmx.discovery (с режимом «beans»). MBean имеет следующие определённые для него свойства (некоторые из них будут игнорироваться; см. ниже):

```
name=test
  =Type
attributes []=1,2,3
Name=NameOfTheTest
domAin=some
```

В результате JMX обнаружения, будут сгенерированы следующие LLD макросы:

- {#JMXDOMAIN} — внутренний макрос Zabbix, описывающий домен MBean
- {#JMXOBJ} — внутренний макрос Zabbix, описывающий объект MBean
- {#JMXNAME} — создан из свойства «name»

Проигнорированные свойства:

- тип : его имя содержит неподдерживаемые символы (не-ASCII)
- attributes[] : его имя содержит неподдерживаемые символы (квадратные скобки не поддерживаются)
- Name : уже определено (name=test)
- domAin : зарезервированное имя Zabbix

Примеры

Давайте обратимся ещё к двум более практическим примерам создания правил LLD с применением MBean. Для лучшего понимания разницы между правилом LLD, собирающем объекты MBeans, и правилом LLD, собирающем атрибуты MBean, взгляните на следующую таблицу:

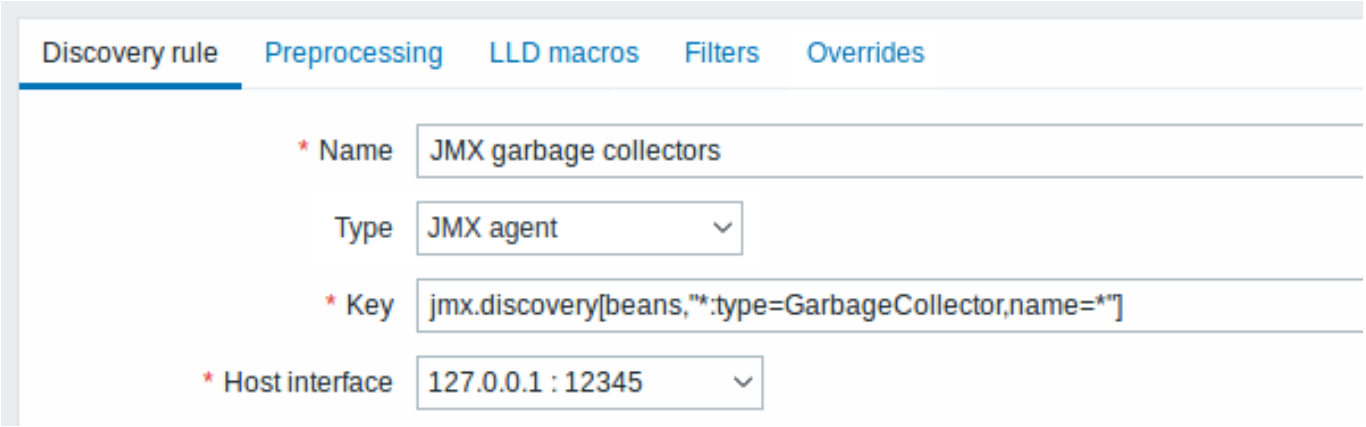
| **MBean1** | **MBean2** | **MBean3** |
|---|---|---|
| MBean1Attribute1 | MBean2Attribute1 | MBean3Attribute1 |
| MBean1Attribute2 | MBean2Attribute2 | MBean3Attribute2 |
| MBean1Attribute3 | MBean2Attribute3 | MBean3Attribute3 |

Пример 1: Обнаружение MBeans

Это правило вернёт 3 объекта: верхнюю строку со столбцами: MBean1, MBean2, MBean3.

Для более подробной информации об объектах, пожалуйста, обратитесь к таблице поддерживаемые макросы, раздел *Обнаружение объектов MBean*.

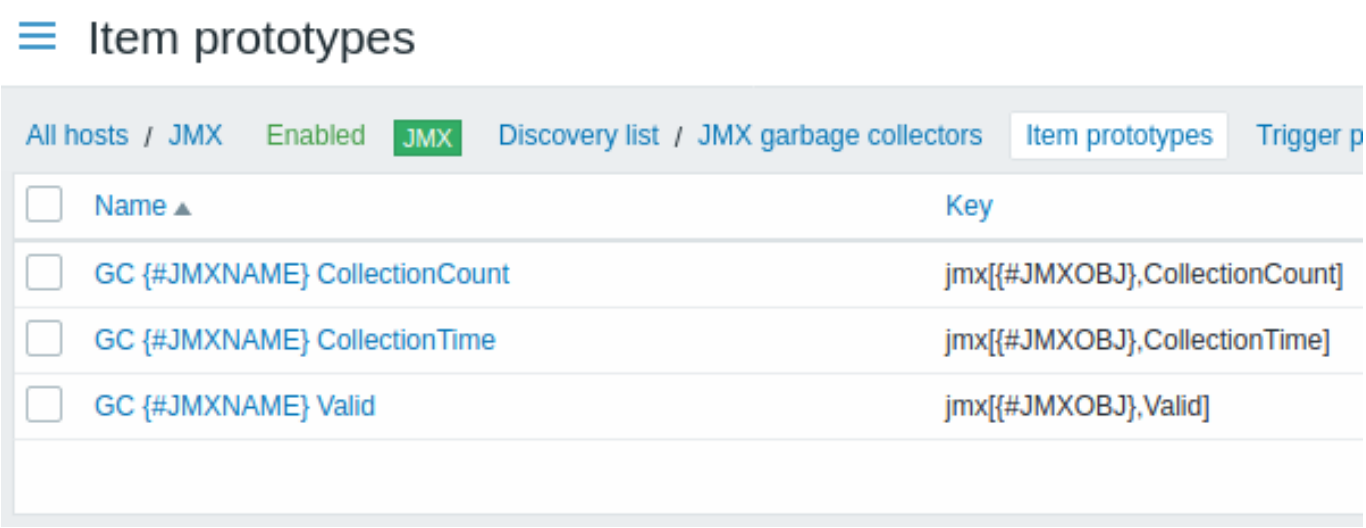
Конфигурация правила обнаружения для сбора объектов MBeans (без атрибутов) выглядит наподобие следующего:




Используемый здесь ключ:

```
jmx.discovery[beans,"*:type=GarbageCollector,name=*"]
```

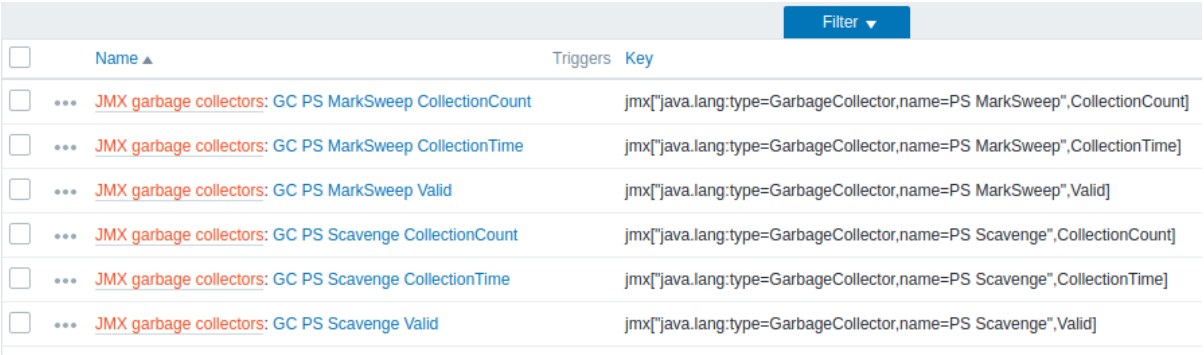
Будут обнаружены все сборщики мусора (garbage collectors) без атрибутов. Поскольку сборщики мусора имеют одинаковый набор атрибутов, мы можем использовать желаемые атрибуты в прототипах элементов данных следующим образом:




Используемые здесь ключи:

```
jmx[{#JMXOBJ},CollectionCount]
jmx[{#JMXOBJ},CollectionTime]
jmx[{#JMXOBJ},Valid]
```

Правило LLD обнаружения приведёт к чему-то, близкому к этому (элементы данных обнаружены для двух сборщиков мусора):

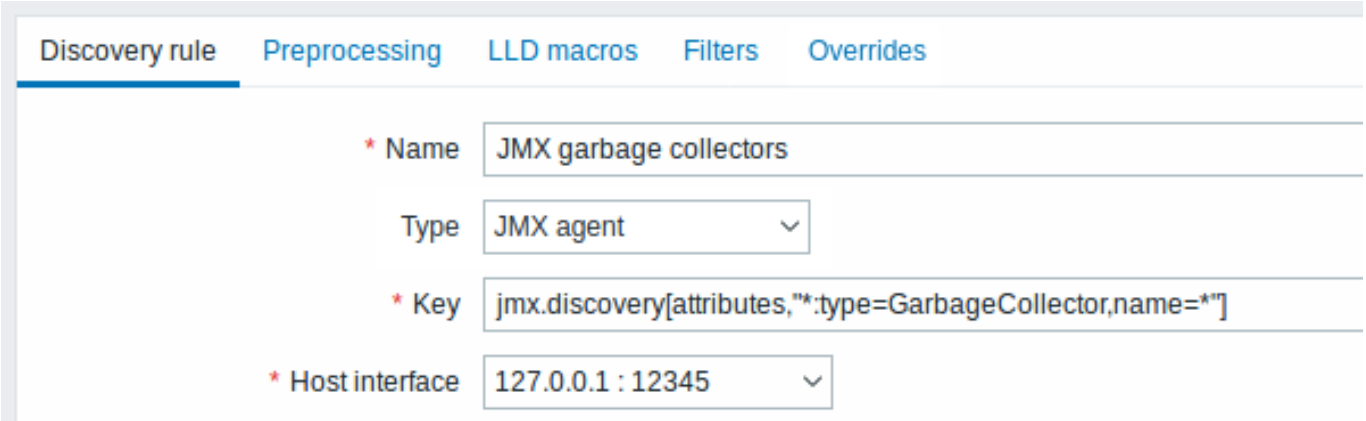



Пример 2: Обнаружение атрибутов MBean

Этот объект вернёт 9 объектов со следующими полями: MBean1Attribute1, MBean2Attribute1, MBean3Attribute1, MBean1Attribute2, MBean2Attribute2, MBean3Attribute2, MBean1Attribute3, MBean2Attribute3, MBean3Attribute3.

Для более подробной информации об объектах, пожалуйста, обратитесь к таблице поддеживаемых макросов, раздел *Обнаружение атрибутов MBean*.

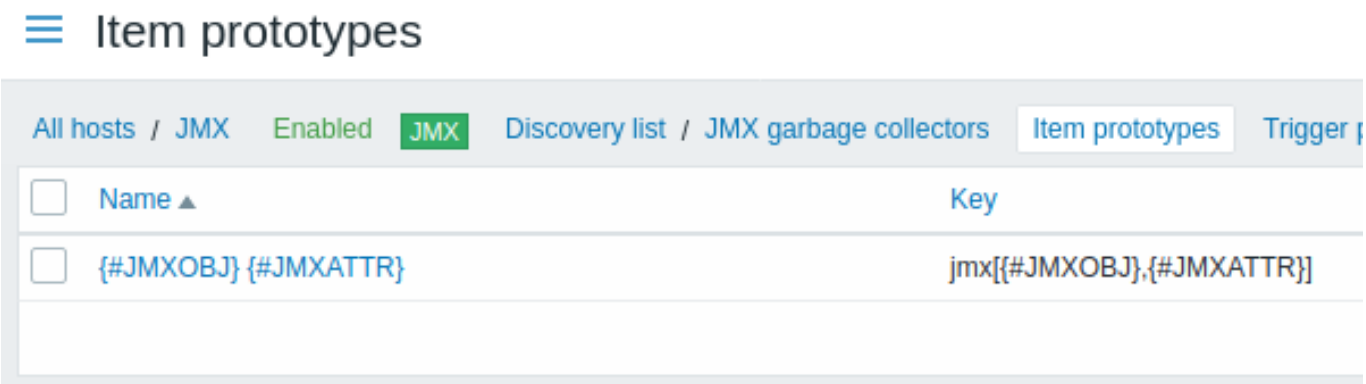
Конфигурация правила обнаружения, собирающего атрибуты MBean, выглядит наподобие следующего:




Используемый здесь ключ:

```
jmx.discovery[attributes,"*:type=GarbageCollector,name=*"]
```

Будут обнаружены все сборщики мусора с одним атрибутом.




В данном конкретном случае будет создан элемент данных из прототипа для каждого атрибута MBean. Главный недостаток этой конфигурации — что невозможно создание триггера из прототипа триггера, поскольку имеется только один прототип элемента данных для всех атрибутов. Поэтому такая настройка может использоваться для сбора данных, но не рекомендуется для автоматического мониторинга.

Использование jmx.get

Элемент данных `jmx.get[]` похож на `jmx.discovery[]`, но он не превращает свойства объектов Java в имена макросов низкоуровневого обнаружения и поэтому может возвращать значения без ограничений, связанных с генерацией имён LLD макросов — таких как дефисы или не-ASCII символы.

При использовании `jmx.get[]` для обнаружения, макросы низкоуровневого обнаружения могут быть определены отдельно на вкладке настраиваемых LLD макросов настроек правила обнаружения, используя JSONPath для указания на нужные значения.

Обнаружение объектов MBean

Элемент данных обнаружения: `jmx.get[beans,"com.example:type=*,*"]`

Ответ:

```
[
    {
        "object": "com.example:type=Hello,data-src=data-base,  =    ",
        "domain": "com.example",
        "properties": {
            "data-src": "data-base",
            "  ": "    ",
            "type": "Hello"
        }
    },
    {
        "object": "com.example:type=Atomic",
        "domain": "com.example",
```

```
        "properties": {
            "type": "Atomic"
        }
    }
]
```

Обнаружение атрибутов MBean

Элемент данных обнаружения: `jmx.get[attributes,"com.example:type=*,*"]`

Ответ:

```
[
    {
        "object": "com.example:type=*",
        "domain": "com.example",
        "properties": {
            "type": "Simple"
        }
    },
    {
        "object": "com.zabbix:type=yes,domain=zabbix.com,data-source=/dev/rand,  =    ,obj=true",
        "domain": "com.zabbix",
        "properties": {
            "type": "Hello",
            "domain": "com.example",
            "data-source": "/dev/rand",
            "  ": "    ",
            "obj": true
        }
    }
]
```

#### 7. Обнаружение датчиков IPMI

Обзор

Существует возможность автоматического обнаружения датчиков IPMI.

Чтобы это сделать, можно использовать комбинацию из:

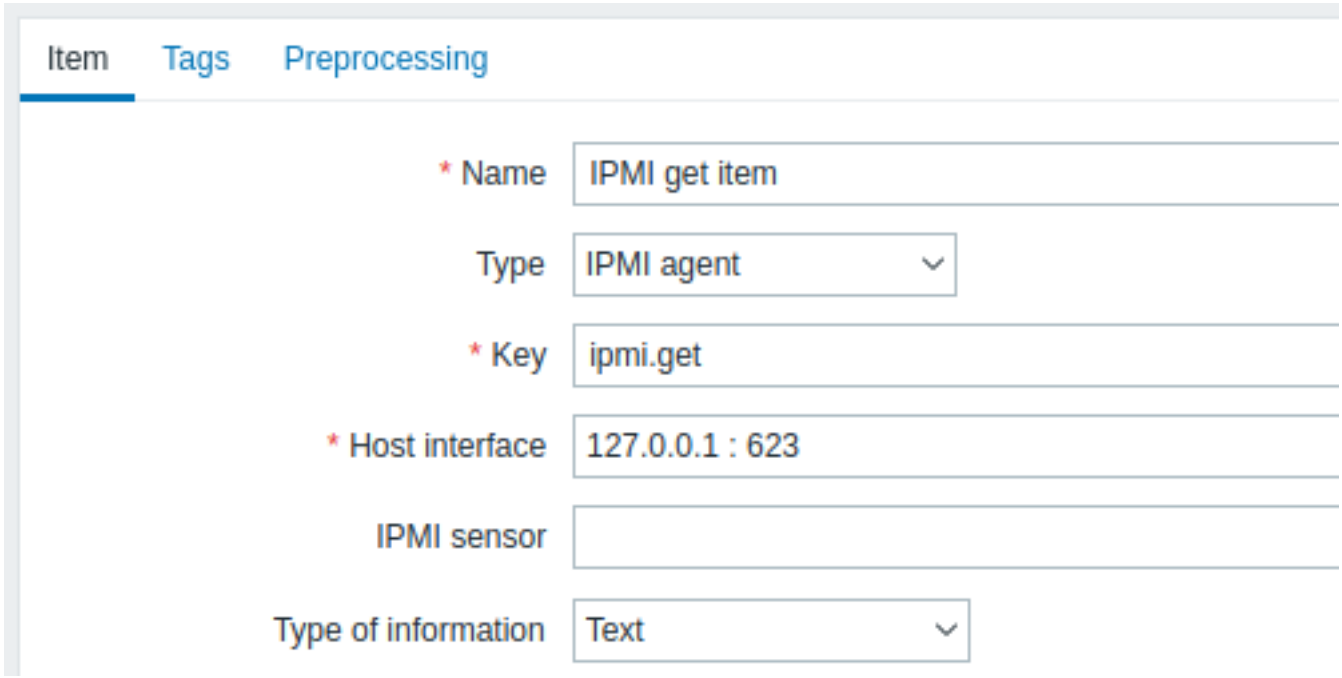
- элемента данных IPMI `ipmi.get` в качестве основного элемента данных
- зависимых от него правила низкоуровневого обнаружения и прототипов элементов данных

Настройка

Основной элемент данных

Создайте элемент данных IPMI, используя следующий ключ:

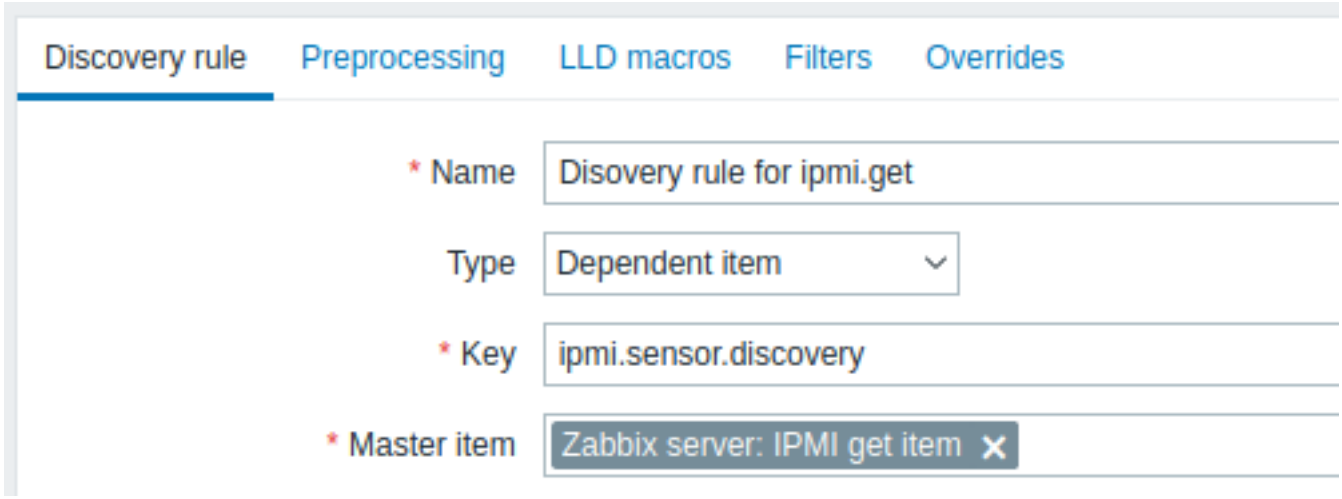
`ipmi.get`




Выставьте тип информации в «Текст», чтобы иметь возможность обрабатывать потенциально большие данные JSON.

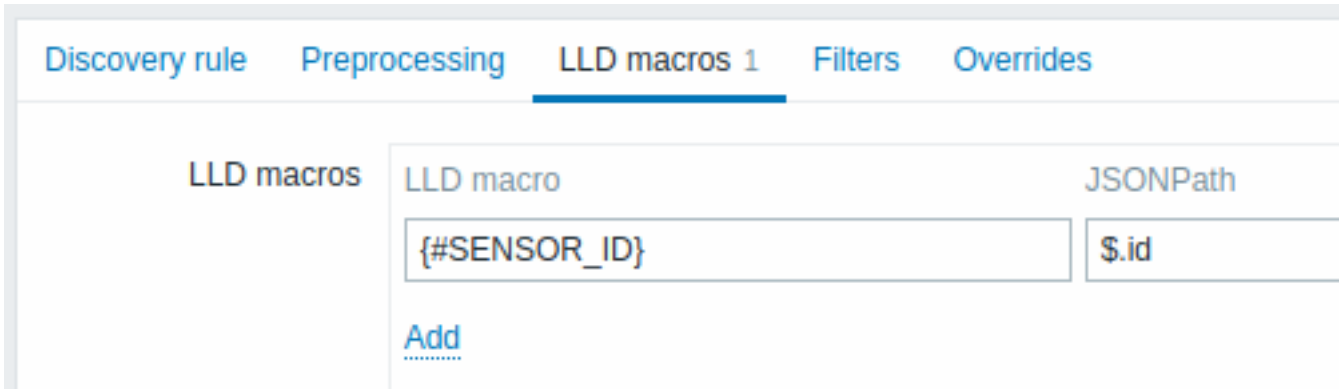
Зависимое правило LLD

Создайте правило низкоуровневого обнаружения, имеющее тип «Зависимый элемент данных»:



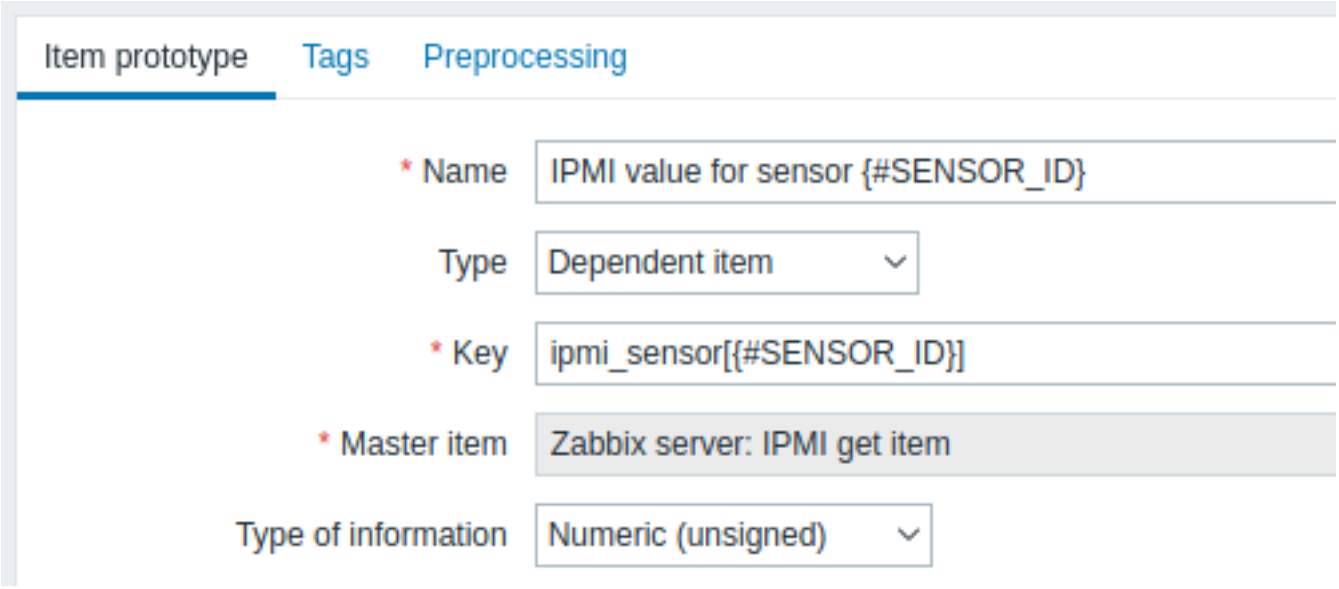

В качестве основного элемента данных выберите созданный ранее элемент данных `ipmi.get`.

На вкладке «LLD макросы» определите настраиваемый макрос с соответствующим JSONPath:




Зависимый прототип элемента данных

Создайте для этого правила LLD прототип элемента данных с типом «Зависимый элемент данных». В качестве основного элемента данных для этого прототипа выберите созданный ранее элемент данных `ipmi.get`.



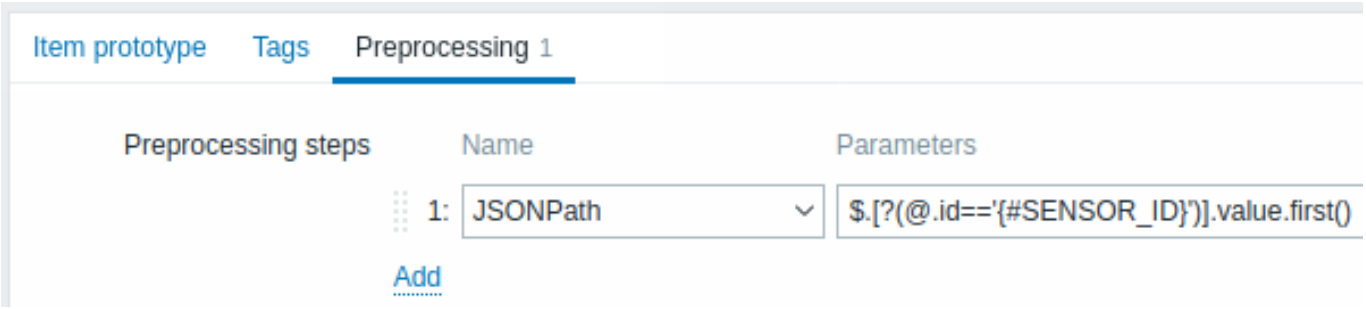

Обратите внимание на использование макроса {#SENSOR_ID} в имени и ключе прототипа элемента данных:

- *Имя*: IPMI value for sensor {#SENSOR_ID}
- *Ключ*: ipmi_sensor[{#SENSOR_ID}]

В качестве типа информации, выберите *Числовой (целое положительное)*.

На вкладке «Предобработка» прототипа элемента данных выберите «JSONPath» и используйте следующее выражение JSONPath как параметр:

```
$.[?(@.id=='{#SENSOR_ID}')].value.first()
```




При работе обнаружения будет создан один элемент данных на каждый датчик IPMI. Этот элемент данных будет возвращать целочисленное значение для данного датчика.

#### 8. Обнаружение служб systemd

Обзор

Существует возможность обнаружения модулей (units) systemd (службы, по умолчанию) с помощью Zabbix.

Ключ элемента данных

Элемент данных, который нужно использовать в правиле обнаружения, — это

`systemd.unit.discovery`

> **Attention:**
> Этот ключ элемента данных поддерживается только в Zabbix agent 2.

Этот элемент данных возвращает JSON с информацией о юнитах systemd, например:

```
[{
    "{#UNIT.NAME}": "mysqld.service",
    "{#UNIT.DESCRIPTION}": "MySQL Server",
    "{#UNIT.LOADSTATE}": "loaded",
    "{#UNIT.ACTIVESTATE}": "active",
    "{#UNIT.SUBSTATE}": "running",
```

```
    "{#UNIT.FOLLOWED}": "",
    "{#UNIT.PATH}": "/org/freedesktop/systemd1/unit/mysqld_2eservice",
    "{#UNIT.JOBID}": 0,
    "{#UNIT.JOBTYPE}": "",
    "{#UNIT.JOBPATH}": "/",
    "{#UNIT.UNITFILESTATE}": "enabled"
    "{#UNIT.SERVICETYPE}": "simple"
}, {
    "{#UNIT.NAME}": "systemd-journald.socket",
    "{#UNIT.DESCRIPTION}": "Journal Socket",
    "{#UNIT.LOADSTATE}": "loaded",
    "{#UNIT.ACTIVESTATE}": "active",
    "{#UNIT.SUBSTATE}": "running",
    "{#UNIT.FOLLOWED}": "",
    "{#UNIT.PATH}": "/org/freedesktop/systemd1/unit/systemd_2djournald_2esocket",
    "{#UNIT.JOBID}": 0,
    "{#UNIT.JOBTYPE}": "",
    "{#UNIT.JOBPATH}": "/",
    "{#UNIT.UNITFILESTATE}": "enabled"
}]
```

Обнаружение деактивированных модулей systemd

Также возможно обнаруживать и **деактивированные** (disabled) модули systemd. В этом случае в результирующем JSON-е возвращаются три макроса:

- {#UNIT.PATH}
- {#UNIT.ACTIVESTATE}
- {#UNIT.UNITFILESTATE}.

**Attention:**
Чтобы из прототипов создавались элементы данных и триггеры для деактивированных модулей systemd, убедитесь, что подправлены (или удалены) запрещающие фильтры LLD для {#UNIT.ACTIVESTATE} и {#UNIT.UNITFILESTATE}.

Поддерживаемые макросы

Следующие макросы поддерживаются для использования в правиле обнаружения filter и прототипах элементов данных, триггеров и графиков:

| Macro | Description |
|---|---|
| {#UNIT.NAME} | Имя основной единицы. |
| {#UNIT.DESCRIPTION} | Человекочитаемое описание. |
| {#UNIT.LOADSTATE} | Состояние загрузки (то есть был ли файл единицы успешно загружен) |
| {#UNIT.ACTIVESTATE} | Активное состояние (то есть запущена ли единица в данный момент или нет) |
| {#UNIT.SUBSTATE} | Подсостояние (более детализированная версия активного состояния, специфичная для типа единицы, в отличие от активного состояния) |
| {#UNIT.FOLLOWED} | Единица, состояние которой отслеживается этой единицей, если такая есть; иначе пустая строка. |
| {#UNIT.PATH} | Путь объекта единицы. |
| {#UNIT.JOBID} | Числовой ID задания, если для единицы задания есть задание в очереди; 0 в противном случае. |
| {#UNIT.JOBTYPE} | Тип задания. |
| {#UNIT.JOBPATH} | Путь объекта задания. |
| {#UNIT.UNITFILESTATE} | Состояние установки файла единицы. |
| {#UNIT.SERVICETYPE} | Тип единицы службы (например, `simple`, `forking`, `oneshot`, `idle` и т. д.). Этот макрос возвращается только если единица является службой. |

Прототипы элементов данных

Прототипы элементов данных, которые могут быть созданы на основе обнаружения служб systemd, включают, например:

- Имя элемента данных: {#UNIT.DESCRIPTION} `active state info`; ключ элемента данных: `systemd.unit.info["{#UNIT`
- Имя элемента данных: {#UNIT.DESCRIPTION} `load state info`; ключ элемента данных: `systemd.unit.info["{#UNIT.NA`

### 9. Обнаружение служб Windows

Обзор

Подобно тому как обнаруживаются файловые системы, можно обнаруживать и службы Windows.

Ключ элемента данных

Ключ элемента данных для использования в правиле обнаружения:

`service.discovery`

Поддерживаемые макросы

Следующие макросы поддерживаются для использования в фильтре правила обнаружения и прототипах элементов данных, триггеров и графиков:

| Макрос | Описание |
|---|---|
| {#SERVICE.NAME} | Имя службы. |
| {#SERVICE.DISPLAYNAME} | Отображаемое имя службы |
| {#SERVICE.DESCRIPTION} | Описание службы. |
| {#SERVICE.STATE} | Числовое значение состояния службы.<br>Возможные значения смотрите в описании элемента данных service.info. |
| {#SERVICE.STATENAME} | Наименование состояния службы.<br>Возможные значения смотрите в описании элемента данных service.info. |
| {#SERVICE.PATH} | Путь службы. |
| {#SERVICE.USER} | Пользователь службы. |
| {#SERVICE.STARTUP} | Числовое значение состояния запуска службы.<br>Возможные значения смотрите в описании элемента данных service.info. |
| {#SERVICE.STARTUPNAME} | Наименование состояния запуска службы.<br>Возможные значения смотрите в описании элемента данных service.info. |
| {#SERVICE.STARTUPTRIGGER} | Числовое значение, обозначающее имеющийся тип запуска службы:<br>0 — не имеет триггеров запуска<br>1 — имеет триггеры запуска<br>Полезно обнаруживать службы с такими типами запуска как *Automatic (trigger start)*, *Automatic delayed (trigger start)* и *Manual (trigger start).* |

На основе обнаружения служб Windows можно создать прототипы элементов данных наподобие следующего:

`service.info[{#SERVICE.NAME},< >]`

где принимает следующие значения: *state*, *displayname*, *path*, *user*, *startup* или *description*.

Например, чтобы извлечь отображаемое имя службы, вы можете использовать элемент данных «service.info[{#SERVICE.NAME},displa
Если значение не указано («service.info[{#SERVICE.NAME}]»), по умолчанию используется параметр *state*.

### 10. Обнаружение экземпляров счётчиков производительности Windows

Обзор

Существует возможность обнаружения экземпляров объектов счётчиков производительности Windows. Это полезно для счётчиков производительности с несколькими экземплярами.

Ключ элемента данных

Для настройки правила обнаружения используйте следующий элемент данных:

- `perf_instance.discovery[ ]`

Обратите внимание, что имя объекта может быть локализованным. Например:

```
perf_instance.discovery[Processor] #       -    .
perf_instance.discovery[Processador] #       -    .
```

В качестве альтернативы, чтобы гарантировать, что имя объекта указано на английском языке, независимо от локализации ОС, используйте следующий элемент данных:

- `perf_instance_en.discovery[ ]`

Например:

```
perf_instance_en.discovery[Processor]
perf_instance_en.discovery[Memory]
```

Поддерживаемые макросы

Процесс обнаружения вернёт все экземпляры указанного объекта в макросе {#INSTANCE}:

```
[
  {"{#INSTANCE}":"0"},
  {"{#INSTANCE}":"1"},
  {"{#INSTANCE}":"_Total"}
]
```

Этот макрос может использоваться в прототипах элементов данных perf_counter[] и perf_counter_en[].

Например, если в правиле обнаружения используется ключ элемента данных perf_instance.discovery[Processor], то вы можете создать прототип элемента данных:

```
perf_counter["\Processor({#INSTANCE})\% Processor Time"]
```

Примечания:

- Если указанный объект не найден или не поддерживает экземпляры переменных, то элемент данных обнаружения станет НЕПОДДЕРЖИВАЕМЫМ.
- Если указанный объект поддерживает экземпляры переменных, но в данный момент не имеет никаких экземпляров, то будет возвращён пустой массив JSON.
- Дубликаты экземпляров они будут пропущены.

#### 11. Обнаружение с использованием запросов WMI

Обзор

WMI — мощный интерфейс в Windows, который можно использовать для получения различной информации о компонентах, службах, состоянии и установленном программном обеспечении Windows.

Его можно использовать для обнаружения физических дисков и сбора их данных производительности, обнаружения сетевых интерфейсов, обнаружения гостевых систем Hyper-V, мониторинга служб Windows и множества других задач в ОС Windows.

Этот тип низкоуровневого обнаружения выполняется с использованием запросов WQL, результаты которых автоматически преобразуются в объект JSON, подходящий для низкоуровневого обнаружения.

Ключ элемента данных

Элемент данных, используемый в правиле обнаружения:

```
wmi.getall[<       _   >,<    >]
```

Этот элемент данных преобразует результат запроса в массив JSON. Например:

```
select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'
```

может вернуть что-то наподобие этого:

```
[
    {
        "DeviceID" : "\\.\PHYSICALDRIVE0",
        "BytesPerSector" : 512,
        "Capabilities" : [
            3,
            4
        ],
        "CapabilityDescriptions" : [
            "Random Access",
            "Supports Writing"
        ],
        "Caption" : "VBOX HARDDISK ATA Device",
        "ConfigManagerErrorCode" : 0,
        "ConfigManagerUserConfig" : "False",
```

```
        "CreationClassName" : "Win32_DiskDrive",
        "Description" : "Disk drive",
        "FirmwareRevision" : "1.0",
        "Index" : 0,
        "InterfaceType" : "IDE"
    },
    {
        "DeviceID" : "\\.\PHYSICALDRIVE1",
        "BytesPerSector" : 512,
        "Capabilities" : [
            3,
            4
        ],
        "CapabilityDescriptions" : [
            "Random Access",
            "Supports Writing"
        ],
        "Caption" : "VBOX HARDDISK ATA Device",
        "ConfigManagerErrorCode" : 0,
        "ConfigManagerUserConfig" : "False",
        "CreationClassName" : "Win32_DiskDrive",
        "Description" : "Disk drive",
        "FirmwareRevision" : "1.0",
        "Index" : 1,
        "InterfaceType" : "IDE"
    }
]
```

Макросы низкоуровневого обнаружения

Несмотря на то, что макросы низкоуровневого обнаружения не создаются в возвращаемом JSON, эти макросы могут быть определены пользователем в дополнительном шаге с использованием функциональности настраиваемых LLD макросов с JSONPath, которые будут указывать на обнаруженные значения в полученном JSON.

Эти макросы можно затем использовать для создания прототипов элементов данных, триггеров, графиков и т.п.

#### 12. Обнаружение с использованием запросов ODBC SQL

Обзор

Этот тип низкоуровневого обнаружения осуществляется с использованием SQL запросов, результаты которых автоматически преобразуются в объект JSON, пригодный для низкоуровневого обнаружения.

Ключ элемента данных

SQL-запросы выполняются с использованием типа элемента данных "Database monitor". Поэтому большинство инструкций на странице ODBC monitoring применимы для того, чтобы получить рабочее правило обнаружения "Database monitor".

В правилах обнаружения "Database monitor" можно использовать два ключа элемента данных:

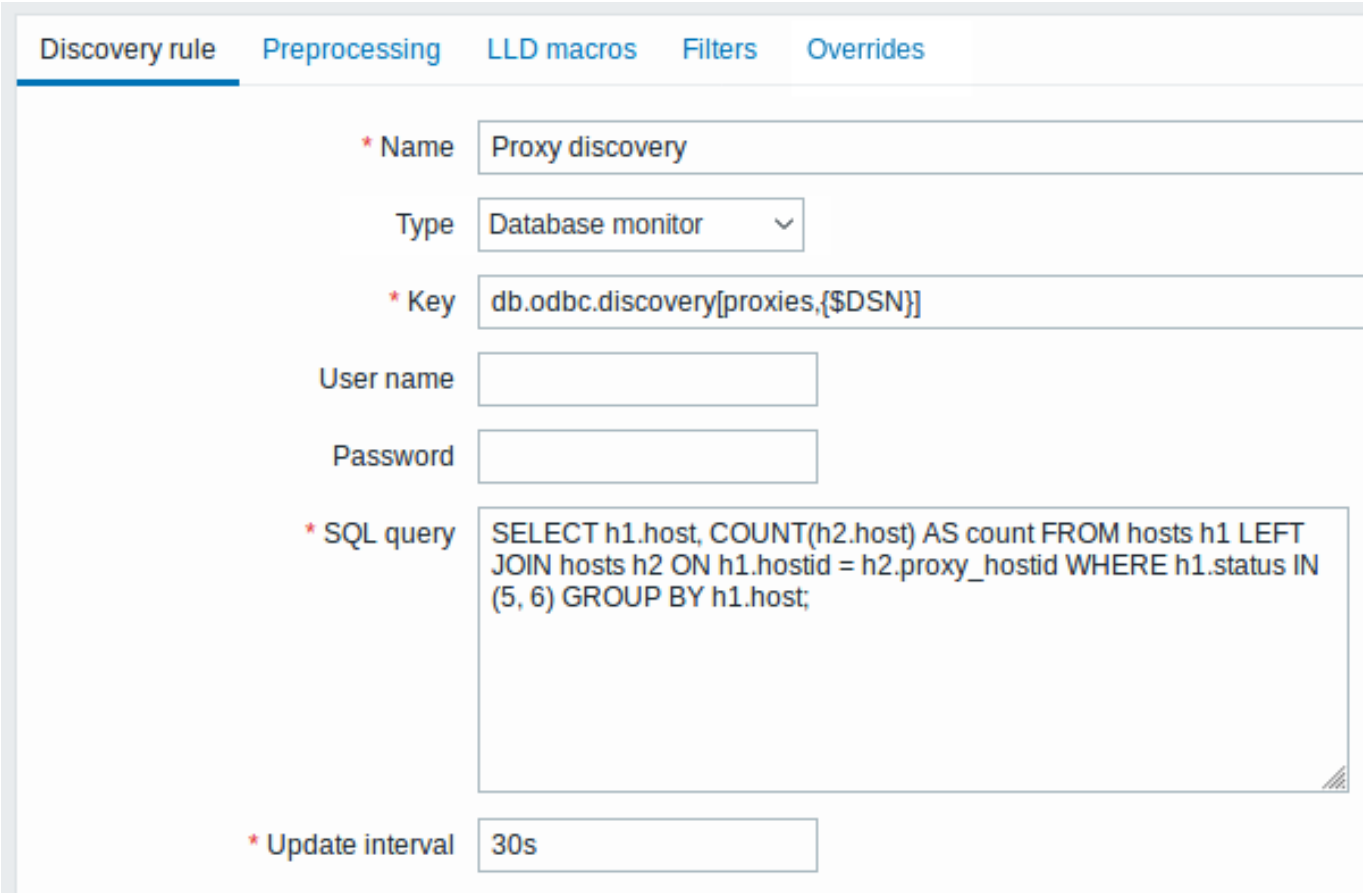
- **db.odbc.discovery**[<unique short description>,<dsn>,<connection string>] - этот элемент преобразует результат SQL-запроса в JSON-массив, превращая имена столбцов из результата запроса в имена макросов низкоуровневого обнаружения, сопоставленные со значениями обнаруженных полей. Эти макросы можно использовать при создании прототипов элемента данных, триггера и т. д. См. также: Using db.odbc.discovery.
- **db.odbc.get**[<unique short description>,<dsn>,<connection string>] - этот элемент преобразует результат SQL-запроса в JSON-массив, сохраняя исходные имена столбцов из результата запроса в качестве имен полей в JSON, сопоставленных со значениями обнаруженных данных. По сравнению с `db.odbc.discovery[]`, этот элемент не создает макросы низкоуровневого обнаружения в возвращаемом JSON, поэтому нет необходимости проверять, могут ли имена столбцов быть допустимыми именами макросов. Макросы низкоуровневого обнаружения можно определить дополнительно, если это требуется, с помощью функциональности custom LLD macro, используя JSONPath, указывающий на обнаруженные значения в возвращаемом JSON. См. также: Using db.odbc.get.

Использование db.odbc.discovery

Следующий пример демонстрирует, как SQL-запрос преобразуется в JSON с использованием низкоуровневого обнаружения прокси Zabbix на основе ODBC-запроса к базе данных Zabbix. Это полезно для автоматического создания внутренних

элементов данных ”zabbix[proxy,<name>,lastaccess]” internal items, чтобы отслеживать, какие прокси активны.

Начните с конфигурации правила обнаружения:




Все обязательные поля ввода отмечены красной звездочкой.

Здесь используется следующий прямой запрос к базе данных Zabbix, чтобы выбрать все прокси Zabbix вместе с количеством узлов сети, которые они контролируют. Количество узлов сети можно использовать, например, для исключения пустых прокси:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxyid
+---------+-------+
| host    | count |
+---------+-------+
| Japan 1 |     5 |
| Japan 2 |    12 |
| Latvia  |     3 |
+---------+-------+
3 rows in set (0.01 sec)
```

Внутренним механизмом элемента данных ”db.odbc.discovery[,{$DSN}]” результат этого запроса автоматически преобразуется в следующий JSON:

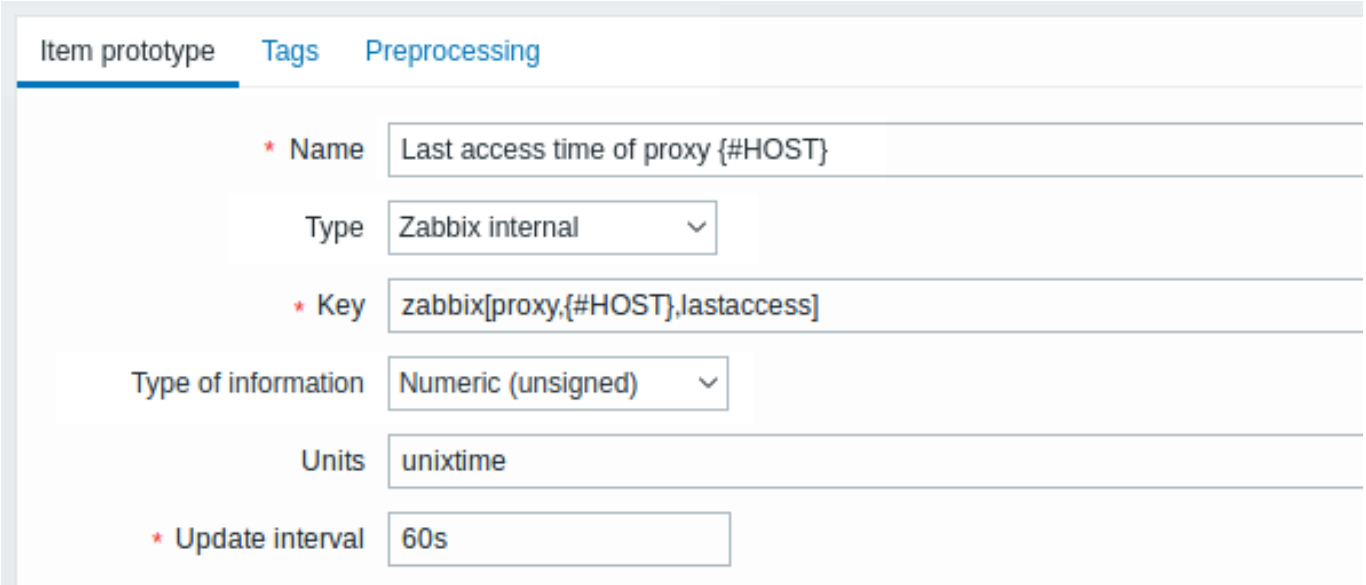
```
[
    {
        "{#HOST}": "Japan 1",
        "{#COUNT}": "5"
    },
    {
        "{#HOST}": "Japan 2",
        "{#COUNT}": "12"
    },
    {
        "{#HOST}": "Latvia",
        "{#COUNT}": "3"
    }
]
```

Как видно, имена столбцов становятся именами макросов, а выбранные строки - значениями этих макросов.

**Note:**
Если неочевидно, как имя столбца будет преобразовано в имя макроса, рекомендуется использовать псевдонимы столбцов, как в примере выше: "COUNT(h2.host) AS count".
Если имя столбца невозможно преобразовать в допустимое имя макроса, правило обнаружения перейдет в состояние "не поддерживается", а в сообщении об ошибке будет указан номер проблемного столбца. Если требуется дополнительная помощь, полученные имена столбцов можно увидеть при DebugLevel=4 в файле журнала сервера Zabbix:

```
$ grep db.odbc.discovery /tmp/zabbix_server.log
 ...
 23876:20150114:153410.856 In db_odbc_discovery() query:'SELECT h1.host, COUNT(h2.host) FROM hosts h1 I
 23876:20150114:153410.860 db_odbc_discovery() column[1]:'host'
 23876:20150114:153410.860 db_odbc_discovery() column[2]:'COUNT(h2.host)'
 23876:20150114:153410.860 End of db_odbc_discovery():NOTSUPPORTED
 23876:20150114:153410.860 Item [Zabbix server:db.odbc.discovery[proxies,{$DSN}]] error: Cannot convert
```

Теперь, когда мы понимаем, как SQL-запрос преобразуется в JSON-объект, мы можем использовать макрос {#HOST} в прототипах элементов данных:




После выполнения обнаружения для каждого прокси будет создан элемент данных:

Использование db.odbc.get

При использовании db.odbc.get[,{$DSN}] и следующего примера SQL:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxyid
+---------+-------+
| host    | count |
+---------+-------+
| Japan 1 |     5 |
| Japan 2 |    12 |
| Latvia  |     3 |
+---------+-------+
3 rows in set (0.01 sec)
```

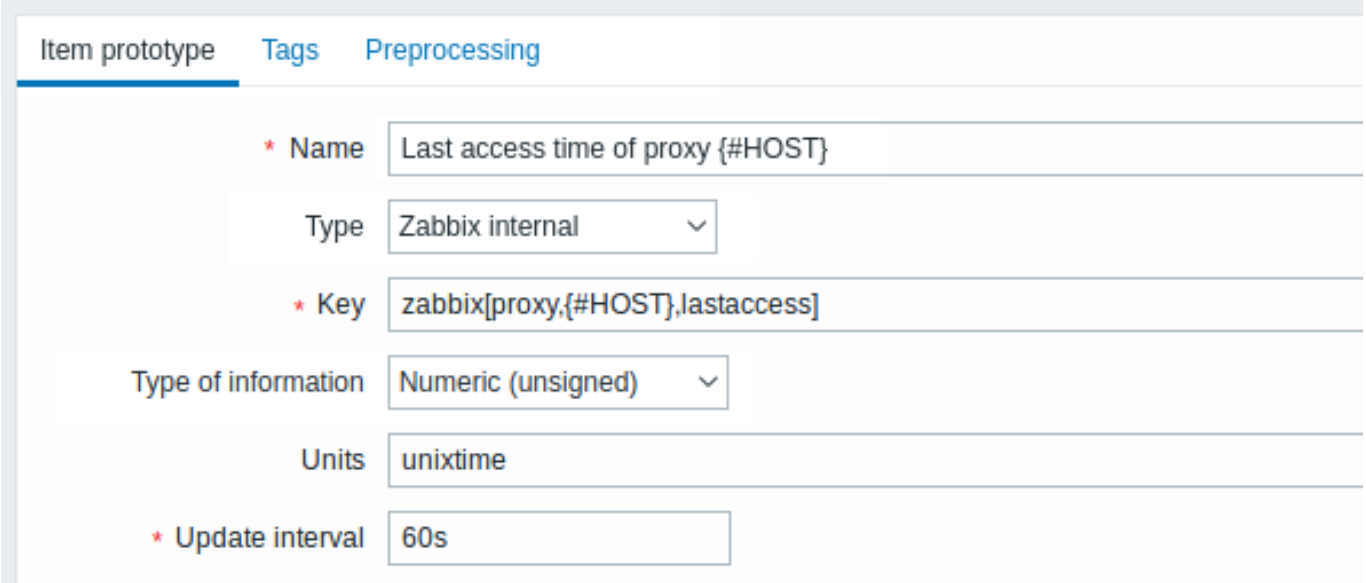
вернётся этот JSON:

```
[
    {
        "host": "Japan 1",
        "count": "5"
    },
    {
        "host": "Japan 2",
        "count": "12"
    },
    {
        "host": "Latvia",
        "count": "3"
    }
]
```

Как вы можете видеть, макросы низкоуровневого обнаружения здесь отсутствуют. Однако, можно создать настраиваемые макросы низкоуровневого обнаружения на вкладке LLD макросы правила обнаружения, используя JSONPath, например:

```
{#HOST} → $.host
```

Теперь, этот макрос {#HOST} можно использовать в прототипах элементов данных:




#### 13. Обнаружение с использованием данных Prometheus

Обзор

Данные, предоставленные в формате строк Prometheus, могут использоваться для низкоуровневого обнаружения.

См. проверки Prometheus для получения подробной информации о том, как реализован запрос данных Prometheus в Zabbix.

Настройка

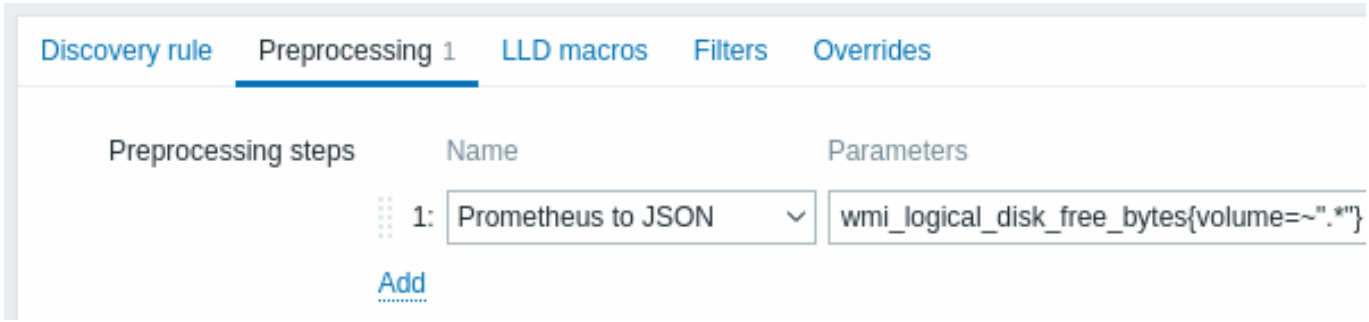
Правило низкоуровневого обнаружения должно быть создано как зависимый элемент данных от основного элемента данных HTTP, собирающего данные Prometheus.

Prometheus в JSON

В правиле обнаружения перейдите на вкладку «Предобработка (Preprocessing)» и выберите опцию *Prometheus в JSON (Prometheus to JSON)*. Для обнаружения необходимы данные в формате JSON, и опция предварительной обработки *Prometheus в JSON* отформатирует их именно таким образом со следующими атрибутами:

- название метрики (metric name)
- значение метрики (metric value)
- помощь (help; если имеется)
- тип (type; если имеется)
- метки (labels; если имеются)
- необработанная строка (raw line)

Например, запрос `wmi_logical_disk_free_bytes`:




из этих строк Prometheus:

```
# HELP wmi_logical_disk_free_bytes Free space in bytes (LogicalDisk.PercentFreeSpace)
# TYPE wmi_logical_disk_free_bytes gauge
wmi_logical_disk_free_bytes{volume="C:"} 3.5180249088e+11
wmi_logical_disk_free_bytes{volume="D:"} 2.627731456e+09
wmi_logical_disk_free_bytes{volume="HarddiskVolume4"} 4.59276288e+08
```

вернёт:

```
[
    {
        "name": "wmi_logical_disk_free_bytes",
        "help": "Free space in bytes (LogicalDisk.PercentFreeSpace)",
        "type": "gauge",
        "labels": {
            "volume": "C:"
         },
        "value": "3.5180249088e+11",
        "line_raw": "wmi_logical_disk_free_bytes{volume=\"C:\"} 3.5180249088e+11"
    },
    {
        "name": "wmi_logical_disk_free_bytes",
        "help": "Free space in bytes (LogicalDisk.PercentFreeSpace)",
        "type": "gauge",
        "labels": {
            "volume": "D:"
         },
        "value": "2.627731456e+09",
        "line_raw": "wmi_logical_disk_free_bytes{volume=\"D:\"} 2.627731456e+09"
    },
    {
        "name": "wmi_logical_disk_free_bytes",
        "help": "Free space in bytes (LogicalDisk.PercentFreeSpace)",
        "type": "gauge",
        "labels": {
            "volume": "HarddiskVolume4"
         },
        "value": "4.59276288e+08",
        "line_raw": "wmi_logical_disk_free_bytes{volume=\"HarddiskVolume4\"} 4.59276288e+08"
    }
]
```

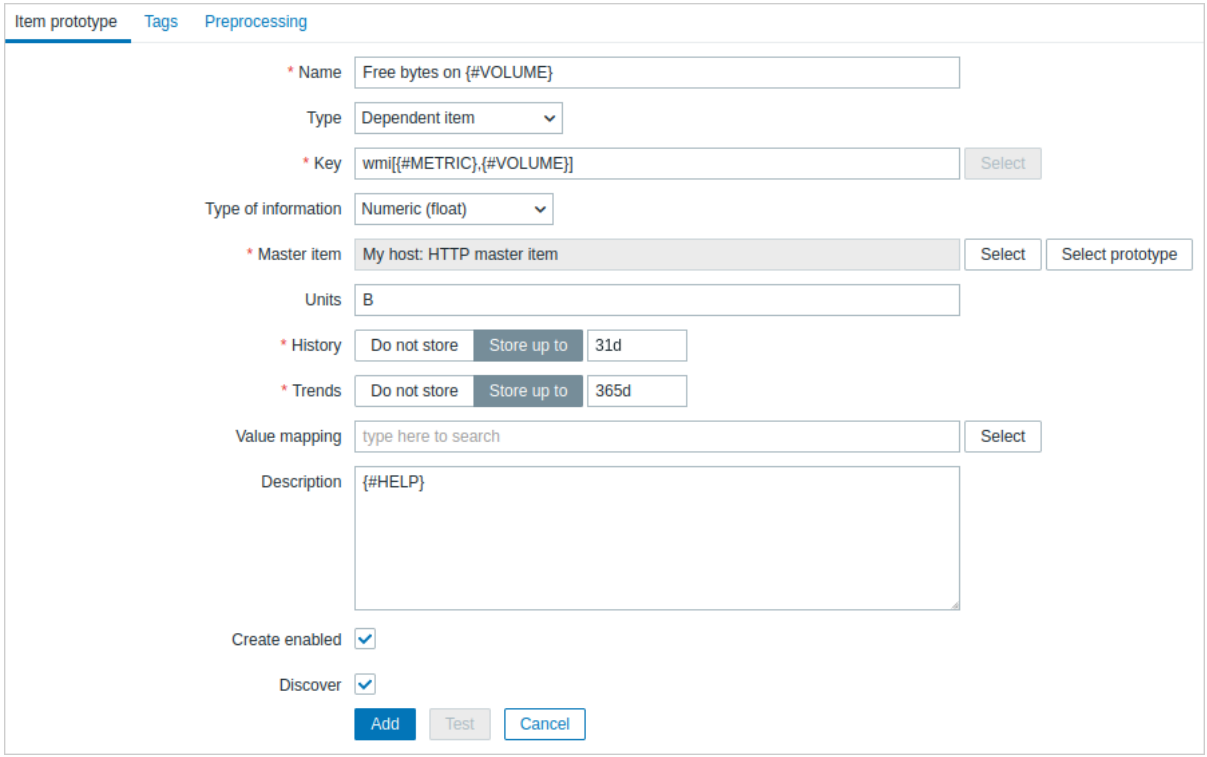
Сопоставление LLD макросов

Далее вам нужно перейти на вкладку «LLD макросы (LLD macros)» и настроить следующие сопоставления:

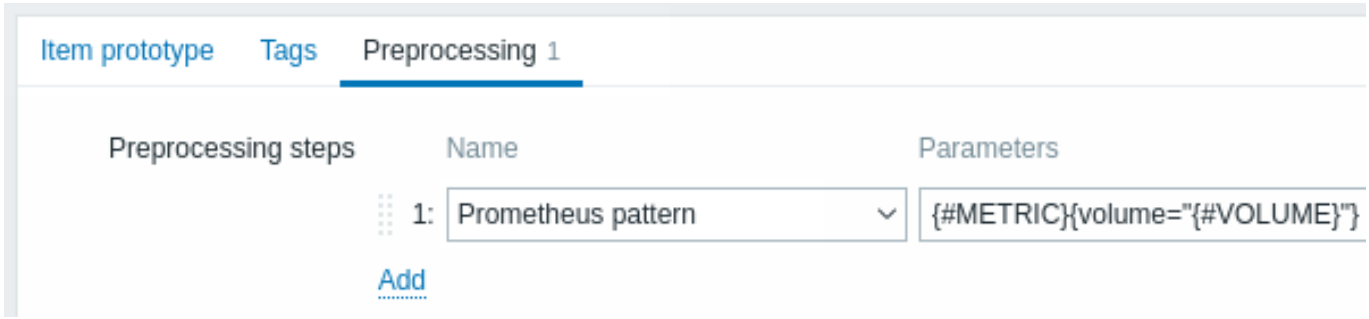
```
{#VOLUME}=$.labels['volume']
{#METRIC}=$['name']
{#HELP}=$['help']
```

Прототип элемента данных

Вы можете создать прототип элемента данных наподобие следующего:




с опциями предобработки:




#### 14. Обнаружение блочных устройств

Аналогично обнаружению файловых систем, также имеется возможность обнаружения блочных устройств и их типов .

Ключ элемента данных

Ключом элемента данных, который используется в правиле обнаружения, является

`vfs.dev.discovery`

Этот элемент данных поддерживается только на платформах Linux.

Вы можете создать правила обнаружения, используя этот элемент данных и:

- фильтр: **{#DEVNAME} совпадает** `sd[\D]$` — для обнаружения устройств с именами «sd0», «sd1», «sd2», ...
- фильтр: **{#DEVTYPE} совпадает** `disk` **И {#DEVNAME} не соответствует** `^loop.*` — для обнаружения типов дисковых устройств, чьи имена не начинаются с «loop»

Поддерживаемые макросы

Этот ключ обнаружения возвращает два макроса — {#DEVNAME} и {#DEVTYPE}, которые задают соответственно имя и тип блочного устройства, например:

```
[
   {
      "{#DEVNAME}":"loop1",
      "{#DEVTYPE}":"disk"
   },
   {
      "{#DEVNAME}":"dm-0",
```

```
        "{#DEVTYPE}":"disk"
    },
    {
        "{#DEVNAME}":"sda",
        "{#DEVTYPE}":"disk"
    },
    {
        "{#DEVNAME}":"sda1",
        "{#DEVTYPE}":"partition"
    }
]
```

Обнаружение блочных устройств позволяет использовать элементы данных `vfs.dev.read[]` и `vfs.dev.write[]`, чтобы создать прототипы элементов данных с использованием макроса {#DEVNAME}, например:

- "vfs.dev.read[{#DEVNAME},sps]"
- "vfs.dev.write[{#DEVNAME},sps]"

{#DEVTYPE} предназначен для фильтрации устройств.

### 15. Обнаружение интерфейсов узлов сети в Zabbix

Обзор

Существует возможность обнаружения всех интерфейсов, настроенных для узла сети в веб-интерфейсе Zabbix.

Ключ элемента данных

Элемент данных, который используется в правиле обнаружения, является внутренним элементом данных Zabbix:

`zabbix[host,discovery,interfaces]`

Этот элемент данных возвращает JSON с описаниями интерфейсов, включая:

- IP адрес / DNS имя хоста (в зависимости от опции «Подключение через» у узла сети)
- Номер порта
- Тип интерфейса (Zabbix агент, SNMP, JMX, IPMI)
- Является ли интерфейс интерфейсом по умолчанию или нет
- Активирована ли функция массового опроса — только для SNMP интерфейсов.

Например:

```
[{"{#IF.CONN}":"192.168.3.1","{#IF.IP}":"192.168.3.1","{#IF.DNS}":"","{#IF.PORT}":"10050","{#IF.TYPE}":"AG
```

При наличии нескольких интерфейсов, их записи сортируются в JSON по:

- типу интерфейса,
- умолчанию — интерфейс по умолчанию помещается до интерфейсов не по умолчанию,
- идентификатору (ID) интерфейса (в порядке возрастания).

Поддерживаемые макросы

Следующие макросы поддерживаются для использования в фильтре правила обнаружения и прототипах элементов данных, триггеров и графиков:

| Макрос | Описание |
|---|---|
| {#IF.CONN} | IP адрес или DNS имя хоста интерфейса. |
| {#IF.IP} | IP адрес интерфейса. |
| {#IF.DNS} | DNS имя хоста интерфейса. |
| {#IF.PORT} | Номер порта интерфейса. |
| {#IF.TYPE} | Тип интерфейса («AGENT», «SNMP», «JMX» или «IPMI»). |
| {#IF.DEFAULT} | Состояние умолчания у интерфейса:<br>0 — не является интерфейсом по умолчанию<br>1 — интерфейс по умолчанию |
| {#IF.SNMP.BULK} | Состояние массовой обработки SNMP у интерфейса:<br>0 — деактивировано<br>1 — активировано<br>Этот макрос возвращается, только если типом интерфейса является «SNMP». |

#### 8 Пользовательские правила LLD

Обзор

Также можно создать полностью пользовательское правило LLD, обнаруживающее любые типы сущностей — например, базы данных на сервере баз данных.

Для этого следует создать пользовательский элемент данных, который возвращает строку JSON, указывающую найденные объекты и, при необходимости, некоторые их свойства. Количество макросов на одну сущность не ограничено — в то время как встроенные правила обнаружения возвращают либо один, либо два макроса (например, два для обнаружения файловых систем), можно возвращать и больше.

Пример

Требуемый формат строки JSON лучше всего показать на примере. Предположим, что вы используете старый агент Zabbix 1.8 (тот, который не поддерживает ключ `vfs.fs.discovery`), но вам все еще нужно обнаруживать файловые системы. Ниже приведен простой Perl-скрипт для Linux, который обнаруживает смонтированные файловые системы и выводит данные в формате JSON, включая как имя файловой системы, так и ее тип. Один из способов использовать его — задать как UserParameter с ключом `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:**
Допустимые символы для имен макросов LLD: **0-9** , **A-Z** , _ , **.**. Строчные буквы в именах не поддерживаются.

Ниже показан пример его вывода (переформатирован для наглядности). JSON для пользовательских проверок обнаружения должен соответствовать тому же формату.

```
[
    { "{#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"  }
]
```

В предыдущем примере требуется, чтобы ключи совпадали с именами макросов LLD, используемыми в прототипах; альтернативный вариант — извлекать значения макросов LLD с помощью JSONPath {#FSNAME} → $.fsname и {#FSTYPE} → $.fstype, что позволяет использовать такой скрипт:

```
####!/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";
```

Ниже показан пример его вывода (переформатирован для наглядности).

```
[
    { "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"  }
]
```

Затем в поле *Filter* правила обнаружения можно указать `{#FSTYPE}` как макрос и `rootfs|ext3` как регулярное выражение.

**Note:**
Вам не обязательно использовать имена макросов вроде `FSNAME`/`FSTYPE` в пользовательских правилах LLD, вы можете использовать любые имена. Если используется JSONPath, то строка LLD будет элементом массива, который может быть объектом, но также может быть и другим массивом или значением.

Обратите внимание: при использовании пользовательского параметра возвращаемое значение ограничено 16 МБ. Подробнее см. ограничения на данные для возвращаемых значений LLD.

## 16. Распределённый мониторинг

**Обзор** Zabbix обеспечивает эффективный и надёжный способ мониторинга распределённой IT инфраструктуры, используя Zabbix прокси.

Прокси могут быть использованы для сбора данных локально от имени централизованного Zabbix сервера и последующего перенаправления данных на сервер.

Возможности прокси

Делая выбор между использованием прокси или нет, во внимание должны быть приняты несколько соображений.

| | Прокси |
|---|---|
| *Лёгкость* | **Да** |
| *Веб-интерфейс* | Нет |
| *Независимость работы* | **Да** |
| *Лёгкость обслуживания* | **Да** |
| *Автоматическое создание БД* | **Да**[1] |
| *Локальное администрирование* | Нет |
| *Готовность к встроенному аппаратному обеспечению* | **Да** |
| *Однонаправленные TCP соединения* | **Да** |
| *Централизованная конфигурация* | **Да** |
| *Генерирование оповещений* | Нет |

[1] Функция автоматического создания БД работает только с SQLite. Остальные поддерживаемые базы данных требуют ручного создания.

**Attention:**
Zabbix прокси не знает о периодах обслуживания; подробности смотрите в разделе Расчёт очередей во время обслуживания.

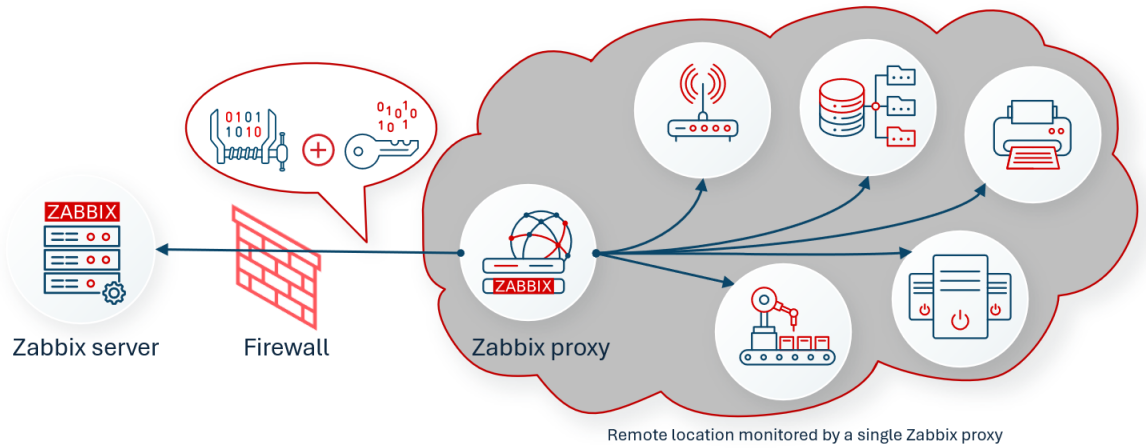
### 1. Прокси

**Обзор** Прокси Zabbix может собирать данные о производительности и доступности от имени сервера Zabbix. Таким образом, прокси может взять на себя часть нагрузки по сбору данных и разгрузить сервер Zabbix.

Кроме того, использование прокси — это самый простой способ реализовать централизованный и распределенный мониторинг, когда все агенты и прокси отправляют данные на один сервер Zabbix, а все данные собираются централизованно.

Прокси Zabbix можно использовать для:

- Мониторинга удаленных площадок
- Мониторинга площадок с ненадежной связью
- Разгрузки сервера Zabbix при мониторинге тысяч устройств
- Упрощения обслуживания распределенного мониторинга




Для работы прокси требуется только одно TCP-соединение с сервером Zabbix. Это упрощает обход межсетевого экрана, поскольку нужно настроить только одно правило.

**Attention:**
Прокси Zabbix должен использовать отдельную базу данных. Если указать базу данных сервера Zabbix, конфигурация будет нарушена.

Все данные, собранные прокси, сначала сохраняются локально, а затем передаются на сервер. Благодаря этому данные не теряются из-за временных проблем связи с сервером. Параметры *ProxyLocalBuffer* и *ProxyOfflineBuffer* в файле конфигурации прокси определяют, как долго данные хранятся локально.

**Attention:**
Может случиться так, что прокси, который получает последние изменения конфигурации напрямую из базы данных сервера Zabbix, имеет более актуальную конфигурацию, чем сервер Zabbix, чья конфигурация может обновляться не так быстро из-за значения CacheUpdateFrequency. В результате прокси может начать собирать данные и отправлять их на сервер Zabbix, который будет игнорировать эти данные.

Прокси Zabbix — это сборщик данных. Он не вычисляет триггеры, не обрабатывает события и не отправляет оповещения. Обзор возможностей прокси приведен в следующей таблице:

| 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** |
| Built-in web monitoring | | **Yes** |
| Item value preprocessing | | **Yes** |
| Network discovery | | **Yes** |
| Active agent autoregistration | | **Yes** |
| Low-level discovery | | **Yes** [2] |
| Remote commands | | **Yes** |
| Calculating triggers | | *No* |
| Processing events | | *No* |
| Event correlation | | *No* |
| Sending alerts | | *No* |

**Note:**
[1] Чтобы агент запрашивал активные проверки у прокси, а не у сервера, прокси должен быть указан в параметре **ServerActive** в файле конфигурации агента.<br> [2] Для LLD прокси Zabbix только собирает и предварительно обрабатывает данные, а затем отправляет их на сервер Zabbix для дальнейшей обработки.

Защита от перегрузки

Если Zabbix сервер был остановлен на какое-то время и прокси собрали много данных, а затем сервер запустился, он может оказаться перегруженным (использование кэша истории некоторое время остаётся на уровне 95-100%). Эта перегрузка может привести к падению производительности, где проверки обрабатываются медленнее, чем должны. Чтобы избежать проблем, возникающих из-за перегрузки кэша истории, была реализована защита от такого сценария.

Когда кэш истории сервера Zabbix полностью заполнен, доступ на запись в кэш истории ограничивается, что приводит к задержке процессов сбора данных сервера. Наиболее распространенный случай перегрузки кэша истории — после останова сервера, когда прокси-серверы пересылают собранные данные. Чтобы избежать этого, был добавлен троттлинг прокси (в настоящее время его нельзя отключить).

Сервер Zabbix остановит приём данных от прокси, когда использование кэша истории достигнет 80%. Вместо этого эти прокси будут помещены в список троттлинга. Это будет продолжаться до тех пор, пока использование кэша не упадёт до 60%. Теперь сервер начнёт принимать данные от прокси по очереди, определенной списком троттлинга. Это означает,

что первый прокси-сервер, который пытался выгрузить данные в течение периода троттлинга, будет обслужен первым, и до тех пор, пока он не завершит, сервер не будет принимать данные от других прокси.

Этот режим троттлинга будет продолжаться до тех пор, пока или использование кэша снова не достигнет 80%, или не упадет до 20%, или пока список троттлинга не опустеет. В первом случае сервер снова перестанет принимать данные прокси. В остальных двух случаях сервер начнёт работать как обычно, принимая данные со всех прокси.

Приведённую выше информацию можно проиллюстрировать следующей таблицей:

| Использование на запись кэша истории | Режим сервера Zabbix | Действие сервера Zabbix |
|---|---|---|
| Достигает 80% | Ожидание | Останавливает приём данных от прокси, но поддерживает *список троттлинга* (упорядоченный по приоритету список прокси-серверов, с которыми надо будет связаться позже). |
| Падает до 60% | Троттлинг | Начинает обработку по списку троттлинга, но всё ещё не принимает данные от прокси. |
| Падает до 20% | Обычный | Удаляет список троттлинга и начинает обычный приём данных от прокси. |

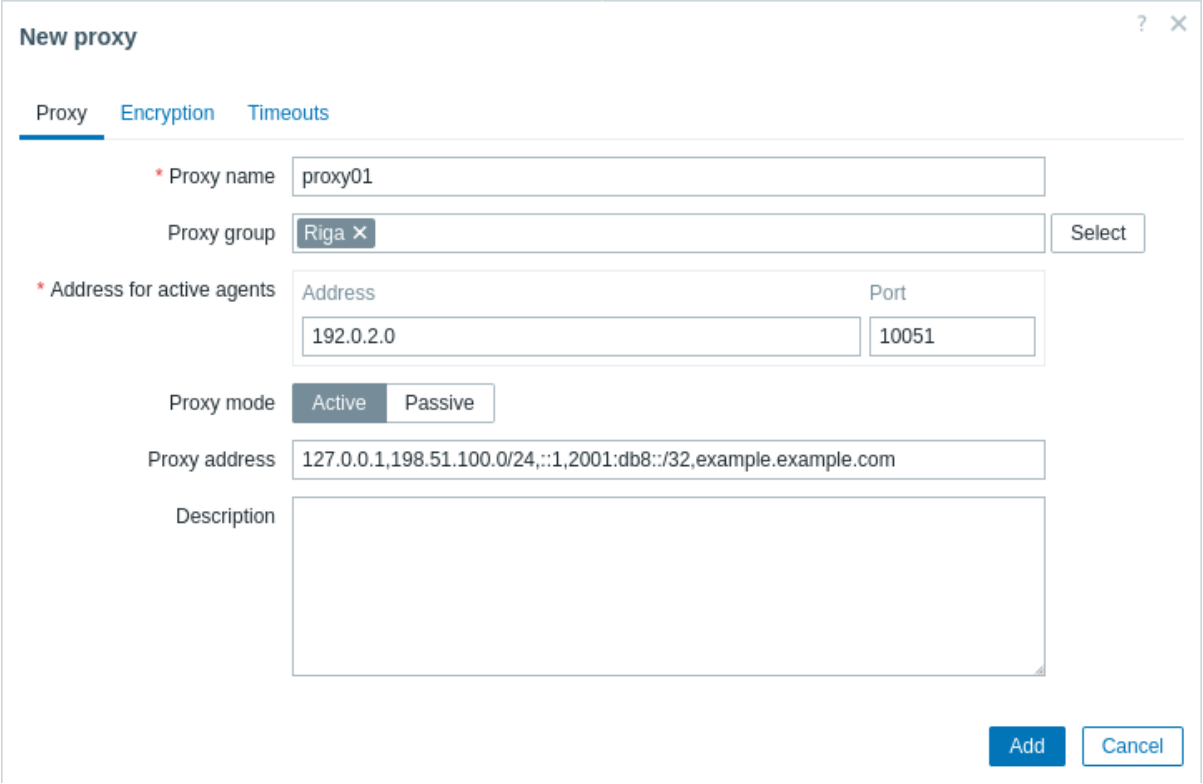
Вы можете использовать внутренний элемент данных zabbix[wcache,history,pused], чтобы скоррелировать это поведение сервера Zabbix с метрикой.

**Настройка** Если у вас есть установленный и настроенный прокси, то самое время настроить его в веб-интерфейсе Zabbix.

Добавление прокси

Чтобы настроить прокси в Zabbix веб-интерфейсе:

- Перейдите в: *Administration* → *Proxies*
- Нажмите *Create proxy*



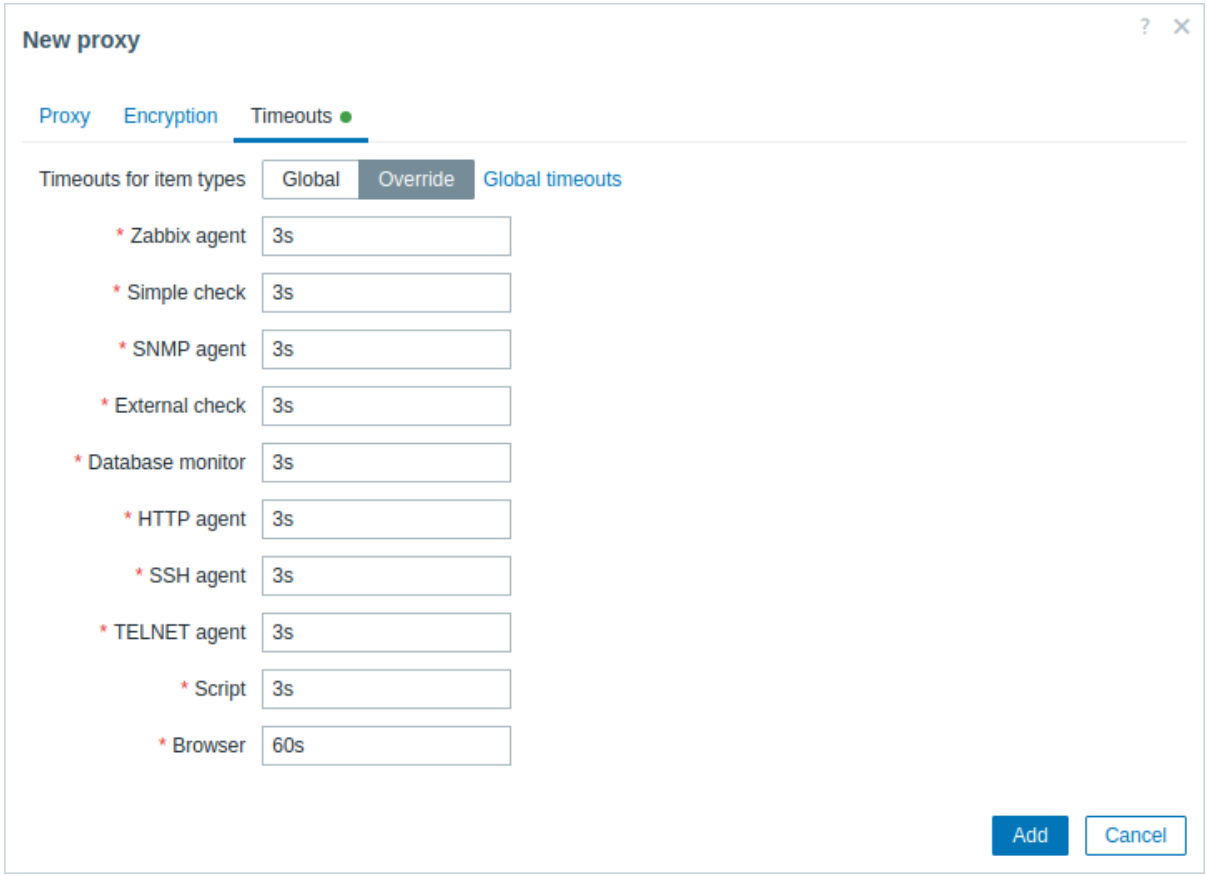

| Parameter | Description |
|---|---|
| *Proxy name* | Введите имя прокси. Оно должно совпадать с именем в параметре *Hostname* в файле конфигурации прокси. |
| *Proxy group* | Выберите одну группу прокси для балансировки нагрузки/высокой доступности. |

| Parameter | Description |
|---|---|
| *Address for active agents* | Введите адрес, к которому должны подключаться контролируемые активные агенты или отправители. Поддерживается **только** для агентов Zabbix 7.0 или более поздней версии.<br>Этот адрес используется для подключения как к активным, так и к пассивным прокси.<br>Это поле доступно только если в поле *Proxy group* выбрана группа прокси. |
| *Address* | IP-адрес/DNS-имя для подключения. |
| *Port* | Номер TCP-порта (по умолчанию 10051) для подключения. Поддерживаются пользовательские макросы. |
| *Proxy mode* | Выберите режим прокси.<br>**Active** - прокси будет подключаться к серверу Zabbix и запрашивать данные конфигурации<br>**Passive** - сервер Zabbix подключается к прокси<br>*Примечание*: без шифрования конфиденциальные данные конфигурации прокси могут стать доступны лицам, имеющим доступ к trapper-порту сервера Zabbix, при использовании активного прокси. Это возможно, поскольку любой может выдать себя за активный прокси и запросить данные конфигурации, если аутентификация не выполняется или адреса прокси не ограничены в поле *Proxy address*. |
| *Proxy address* | Если указано, запросы активного прокси принимаются только из этого списка IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен активного прокси Zabbix.<br>Это поле доступно только если в поле *Proxy mode* выбран активный прокси. Макросы не поддерживаются. |
| *Interface* | Введите сведения об интерфейсе для пассивного прокси.<br>Это поле доступно только если в поле Proxy mode выбран пассивный прокси. |
| *Address* | IP-адрес/DNS-имя пассивного прокси. |
| *Port* | Номер TCP-порта пассивного прокси (по умолчанию 10051). Поддерживаются пользовательские макросы. |
| *Description* | Введите описание прокси. |

Вкладка **Encryption** позволяет требовать зашифрованные соединения с прокси.

| Parameter | Description |
|---|---|
| *Connections to proxy* | Как сервер подключается к пассивному прокси: без шифрования (по умолчанию), с использованием PSK (предварительно разделяемого ключа) или сертификата. |
| *Connections from proxy* | Выберите, какие типы соединений разрешены от активного прокси. Можно выбрать несколько типов соединений одновременно (полезно для тестирования и переключения на другой тип соединения). По умолчанию - ”No encryption”. |
| *Issuer* | Разрешенный издатель сертификата. Сначала сертификат проверяется с CA (центром сертификации). Если он действителен и подписан CA, поле *Issuer* можно использовать для дополнительного ограничения разрешенного CA. Это поле необязательное и предназначено для использования, если ваша установка Zabbix использует сертификаты от нескольких CA. |
| *Subject* | Разрешенный субъект сертификата. Сначала сертификат проверяется с CA. Если он действителен и подписан CA, поле *Subject* можно использовать, чтобы разрешить только одно значение строки *Subject*. Если это поле пустое, принимается любой действительный сертификат, подписанный настроенным CA. |
| *PSK identity* | Строка идентификатора предварительно разделяемого ключа.<br>Не помещайте конфиденциальную информацию в PSK identity, она передается по сети без шифрования, чтобы сообщить получателю, какой PSK использовать. |
| *PSK* | Предварительно разделяемый ключ (hex-строка). Максимальная длина: 512 шестнадцатеричных цифр (PSK длиной 256 байт), если Zabbix использует библиотеку GnuTLS или OpenSSL, 64 шестнадцатеричные цифры (PSK длиной 32 байта), если Zabbix использует библиотеку mbed TLS (PolarSSL). Пример: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952 |

Вкладка **Timeouts** позволяет переопределять глобальные тайм-ауты для типов элементов данных, которые это поддерживают.




| Parameter | Description |
|---|---|
| *Timeouts for item types* | Выберите вариант тайм-аута:<br>**Global** - используется глобальный тайм-аут (отображается в неактивном поле *Timeout* для каждого типа элемента данных);<br>**Override** - используется пользовательский тайм-аут (задается в поле *Timeout* для каждого типа элемента данных). Допустимый диапазон: 1 - 600 с (по умолчанию: наследуется из глобальных тайм-аутов). Поддерживаются суффиксы времени, например 30s, 1m, и пользовательские макросы.<br><br>Нажатие ссылки *Global timeouts* позволяет настроить глобальные тайм-ауты. Обратите внимание, что ссылка *Global timeouts* видна только пользователям типа *Super admin* с правами на раздел веб-интерфейса *Administration* → *General*.<br><br>Поддерживаемые типы элементов данных:<br>- Zabbix agent (как пассивные, так и активные проверки)<br>- Simple check (кроме элементов `icmpping*`, `vmware.*`)<br>- SNMP agent (только для элементов SNMP `walk[OID]` и `get[OID]`)<br>- External check<br>- Database monitor<br>- HTTP agent<br>- SSH agent<br>- TELNET agent<br>- Script<br>- Browser<br><br>Обратите внимание, что тайм-ауты, заданные в режиме **Override**, будут иметь приоритет над глобальными, но будут переопределены индивидуальными тайм-аутами элемента данных, если они заданы в настройках элемента данных. |

**Note:**

Если основная версия прокси не совпадает с основной версией сервера, значок будет отображаться рядом с *Timeouts for item types*, а при наведении будет показано сообщение "Timeouts disabled because the proxy and server versions do not match". В таких случаях прокси будет использовать параметр `Timeout` из файла конфигурации прокси.

Форма редактирования существующего прокси содержит следующие дополнительные кнопки:

- *Refresh configuration* - обновить конфигурацию прокси
- *Clone* - создать новый прокси на основе свойств существующего прокси
- *Delete* - удалить прокси

Настройка узлов сети

Указать, что конкретные узлы сети должны наблюдаться через прокси или группу прокси, вы можете в диалоге настройки узла сети, используя поле *Наблюдение через (Monitored by)*.

Массовое обновление узлов сети является ещё одним способом указать, что узлы сети должны наблюдаться через прокси или группу прокси.

#### 1. Синхронизация настроек мониторинга

Обзор

На этой странице представлена подробная информация об обновлении настроек мониторинга для прокси, т. е. о том, как изменения, внесённые в конфигурацию мониторинга на сервере, синхронизируются с прокси-сервером.

Инкрементальное обновление

Обновление настроек прокси является инкрементальным. Во время синхронизации конфигурации обновляются только изменённые объекты (таким образом, если никакие объекты не менялись, то ничего и не пересылается). Такой подход позволяет сберечь ресурсы и установить меньший интервал (почти мгновенный) для обновлений конфигурации прокси.

Изменения конфигурации прокси отслеживаются при помощи номеров ревизий (revision numbers). В конфигурационные данные, пересылаемые на прокси, включаются только объекты с ревизиями больше, чем ревизия конфигурации прокси.

Объекты для синхронизации конфигурации — следующие:

| Объект | Подробности |
|---|---|
| *данные tls авторегистрации* | Все данные TLS авторегистрации. |
| *выражения* | Все выражения (регулярные выражения, таблицы выражений). |
| *глобальные настройки* | Глобальные настройки, определённые в таблице «config» |
| *узел сети* | Все свойства, интерфейсы, данные инвентаризации, элементы данных, их предобработка и параметры, веб-сценарии узла сети. |
| *макросы узла сети* | Все макросы, определённые на узле сети, а также идентификаторы всех шаблонов, присоединённых к нему. |
| *правила обнаружения прокси-сервера* | Правила обнаружения и проверки, назначенные на прокси-сервер. |

Это означает, что:

- Если на **узле сети** меняется элемент данных, засинхронизируется вся конфигурация этого узла сети.
- Если меняется **регулярное выражение**, будут засинхронизированы все регулярные выражения.

Исключением являются макросы узла сети, которые также пересылаются, если на узле сети что-либо поменялось.

Команда `-R config_cache_reload` на прокси-сервере также будет инициировать инкрементальное обновление.

Обратите внимание, что полная синхронизация конфигурации будет производиться при запуске/перезапуске прокси, переключении кластера HA, если поменялся токен сессии, либо если обновление конфигурации на прокси завершилось неудачей — например, если при получении конфигурационных данных соединение было разорвано.

Параметры конфигурации

Параметр **ProxyConfigFrequency** определяет, как часто конфигурация прокси-сервера синхронизируется с сервером (по умолчанию 10 секунд).

Обратите внимание, что ProxyConfigFrequency является:

- параметром сервера для пассивных прокси
- параметром прокси для активных прокси

На активных прокси ProxyConfigFrequency — это новый параметр, начиная с версии Zabbix 6.4, и его следует использовать вместо ныне устаревшего ConfigFrequency.

**Attention:**
Если используются и ProxyConfigFrequency, и ConfigFrequency, прокси-сервер зарегистрирует ошибку и завершит работу.

Расчёт очередей во время обслуживания

**Attention:**
Прокси-сервер Zabbix не знает о периодах обслуживания; подробности смотрите в Расчёт очередей во время обслуживания.

### 2. Распределение нагрузки и отказоустойчивость прокси

Обзор

Чтобы разрешить распределение нагрузки и отказоустойчивость, Zabbix прокси-серверы могут быть организованы в группы прокси.

Распределение нагрузки и отказоустойчивость прокси-серверов — это автоматическое перераспределение узлов сети между прокси-серверами в пределах группы прокси:

- Если прокси выключается, его узлы сети будут перемещены на другие прокси, таким образом поддерживая отказоустойчивость прокси.
- Если прокси-сервер имеет слишком высокое/низкое количество узлов сети по сравнению с другими прокси-серверами, его узлы сети будут перемещены на другие прокси-серверы, чтобы сбалансировать нагрузку на прокси.

Перераспределение узлов сети работает только между прокси-серверами в группе, которая удовлетворяет следующим условиям:

- Прокси-серверы имеют версию 7.0 или новее.
- Версия прокси-сервера соответствует версии Zabbix сервера. Если используется Zabbix агент (пассивный), то версия прокси должна соответствовать версии агента. Активные агенты требуют только версию Zabbix 7.0 или новее.
- Группа прокси находится в состоянии online.
- Узлы сети настроены на наблюдение через группу прокси, а не через отдельные прокси-серверы.

**Note:**
Работоспособность группы прокси может отслеживаться при помощи внутренних проверок любым узлом сети, назначенным на группу прокси. Однако, для отслеживания работоспособности отдельного прокси-сервера в группе, назначьте узел сети на этот прокси; в противном случае результаты могут быть нецелостными.

Перераспределение узлов сети

Распределение нагрузки и отказоустойчивость прокси управляются Zabbix сервером через процесс proxy group manager, который постоянно отслеживает состояние всех прокси-серверов в каждой группе прокси и распределение узлов сети на них.

Отказоустойчивость прокси в пределах группы обеспечивается через переключение прокси: при останове прокси его узлы сети немедленно перераспределяются на другие прокси. Также происхоит и распределение нагрузки, поскольку узлы сети переназначаются на прокси с наименьшим числом назначенных на них узлов сети.

Кроме того, распределение нагузки прокси активируется, когда количество узлов сети прокси-сервера отличается от среднего по группе как минимум на 10 узлов сети и в 2 раза (превышение либо недостаток узлов сети). Если дисбаланс сохраняется по истечении льготного периода (10 x задержка переключения), то группа прокси ставится в очередь на перераспределение узлов сети.

Процесс proxy group manager перераспределяет узлы сети, используя следующую логику:

1. Вычислить среднее число узлов сети на прокси-сервер.
2. Для прокси-серверов с превышением узлов сети — переместить превышающие узлы сети в пул неназначенных прокси-серверов.
3. Для прокси-серверов с недостатком узлов сети — посчитать, сколько узлов сети нужно для достижения баланса.
4. Убрать нужное число узлов сети от прокси-серверов с наибольшим количеством узлов сети.
5. Переместить неназначенные узлы сети на прокси-серверы с наименьшим количеством узлов сети.

Примеры перераспределения узлов сети:

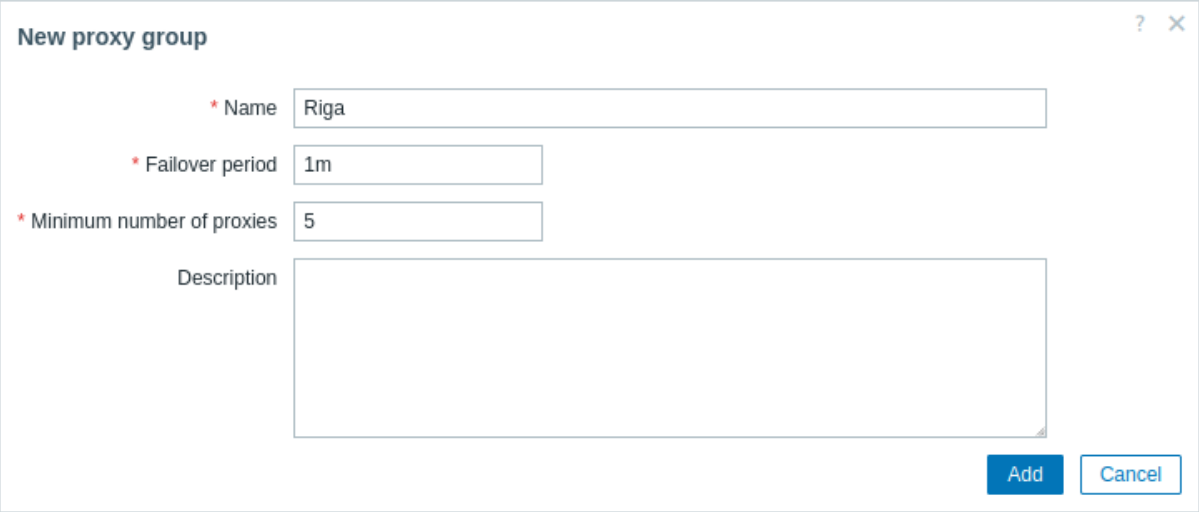
| Узлов сети на прокси-сервере | Среднее по группе | Переназначение узлов сети |
|---|---|---|
| 100 | 50 | Да |
| 60 | 50 | Нет |
| 40 | 50 | Нет |
| 25 | 50 | Да |
| 15 | 5 | Да |
| 10 | 5 | Нет |

**Attention:**
Наличие менее чем 10 узлов сети, наблюдаемых через группу прокси, может привести к неравномерному распределению узлов сети между прокси-серверами в этой группе.

Настройка группы прокси

Для настройки группы прокси в веб-интерфейсе Zabbix:

1. Перейдите к *Администрирование → Группы прокси (Administration → Proxy groups)*
2. Нажмите *Создать группу прокси (Create proxy group)*




| Параметр | Описание |
|---|---|
| *Имя (Name)* | Имя группы прокси-серверов. |
| *Период аварийного переключения (Failover period)* | Период в секундах, в течение которого прокси-сервер в группе прокси должен связаться с сервером Zabbix, чтобы считаться онлайн (по умолчанию: 1m; диапазон: 10s-15m). Если прокси не связывается в течение этого периода, состояние прокси меняется на *Офлайн (Offline)*, и его узлы сети немедленно перераспределяются по другим прокси. Распределение нагрузки прокси начинается по истечении этого 10-кратного периода. Поддерживаются суффиксы времени (например, 30s, 1m) и пользовательские макросы. |

| Параметр | Описание |
|---|---|
| *Минимальное количество прокси (Minimum number of proxies)* | Наименьшее число онлайн прокси-серверов, требующихся для поддержания группы прокси онлайн (по умолчанию: 1; диапазон: 1-1000).<br>Поддерживаются пользовательские макросы.<br><br>Это значение должно быть меньше общего числа прокси-серверов в группе. Например, в группе из 10 прокси, настройка минимума в 10 приведёт к тому, что группа перейдёт в офлайн при сбое любого из прокси-серверов. Обратите внимание, что онлайн-прокси в офлайн-группе продолжают нормально работать, но балансировка нагрузки и отказоустойчивость функционировать не будут. |
| *Описание (Description)* | Описание группы прокси. |
| *Прокси (Proxies)* | При редактировании группы с прокси-серверами отображается список — до пяти прокси-серверов (в виде ссылок или обычного текста, в зависимости от прав пользователя на прокси-серверы). |

Настройка распределения нагрузки прокси

Чтобы использовать распределение нагрузки между прокси-серверами, требуется в веб-интерфейсе Zabbix настроить группу прокси (смотрите выше) и убедиться, что узлы сети наблюдаются через группу прокси, а не отдельные прокси-серверы (можно использовать массовое обновление узлов сети, чтобы переместить их от прокси-сервера в группу прокси).

При использовании Zabbix агентов также настройте их следующим образом:

- Для пассивных проверок — перечислите все прокси-серверы в группе прокси в параметре Server.
- Для активных проверок — рекомендуется перечислить все прокси-серверы в группе прокси либо Zabbix сервер в параметре ServerActive. Обратите внимание, что с группой прокси работать в активном режиме будет только Zabbix агент версии 7.0 (или более поздней).

Если параметр `ServerActive` включает только один прокси-сервер из группы прокси (или Zabbix сервер), то агент всё ещё будет способен подключиться к правильному прокси-серверу. Когда служба агента запустится и подключится к указанному прокси-серверу, агент получит и закэширует полный список IP прокси-серверов и их текущую загрузку в группе. Затем активные проверки будут перенаправлены на правильный онлайн-прокси для этого узла сети, на основе текущего назначения «прокси-сервер — узел сети» в этой группе прокси.

**Warning:**
Наличие только одного прокси, указанного в параметре `ServerActive` Zabbix агента, может привести к потере данных мониторинга в случае, если агент запускается/перезапускается, пока указанный прокси недоступен.

**Note:**
При использовании утилиты Zabbix sender, запросы с данными также перенаправляются на правильный онлайн-прокси для узла сети, на основе текущего назначения «прокси-сервер — узел сети» в этой группе прокси. Однако, если вы отсылаете из одного файла значения нескольких узлов сети, используйте опцию `-g`, стобы предотвратить отсылку данных на неверный прокси.

Zabbix агент также должен быть способен подключаться через брандмауэр ко всем прокси-серверам в группе прокси. В противном случае активные проверки во время перенаправления либо переключения могут подвиснуть или завершиться неудачей. Например:

- Во время активной проверки прокси-сервер может перенаправить агента на другой прокси. Если этот другой прокси заблокирован брандмауэром, коммуникация подвиснет при ожидании ответа.
- В стабильных отказоустойчивых инсталляциях без недавних ребалансировок агенты могут никогда и не обращаться к резервным прокси. Если правила брандмауэра поменялись и не были протестированы, переключение может завершиться неудачей.

Тестирование балансировки нагрузки прокси

Чтобы протестировать балансировку нагрузки прокси:

1. Настройте группу прокси.
2. Убедитесь, что группа прокси находится в состоянии онлайн.
3. Убедитесь, что узлы сети наблюдаются через группу прокси, а не отдельные прокси-серверы (для перемещения узлов сети от прокси-сервера к группе прокси можно использовать массовое обновление узлов сети).
4. Подождите несколько секунд для обновления конфигурации и распределения узлов сети между прокси-серверами в группе прокси. Наблюдайте за изменениями, обновляя список узлов сети в *Администрирование (Administration)* → *Прокси (Proxies)*.

Важные замечания

- SNMP трапы не поддерживаются прокси-серверами в группе прокси.
- Проверки, зависящие от внешних настроек (например, скрипты для внешних проверок или настройки ODBC для проверок баз данных) должны иметь одинаковую конфигурацию на всех прокси-серверах в группе прокси.
- Проверки баз данных требуют дополнительных разрешений на объектах/серверах баз данных.
- Узлы сети VMware, наблюдаемые через группу прокси, будут случайным образом распределены между прокси-серверами в группе. Это приводит к тому, что каждый прокси кэширует все данные VMware, создавая в итоге дополнительную нагрузку на vCenter.
- Узлы сети, созданные на основе данных авторегистрации от прокси-сервера в группе прокси, настраиваются на наблюдение через эту группу прокси. Однако, узлы сети, созданные на основе данных сетевого обнаружения от прокси-сервера в группе прокси, настраиваются на наблюдение через этот прокси-сервер.

## 17. Шифрование

**Обзор** Zabbix поддерживает зашифрованные соединения между компонентами Zabbix с использованием протокола Transport Layer Security (TLS) версии 1.2 и 1.3 (в зависимости от криптографической библиотеки). Поддерживается шифрование на основе сертификатов и на основе предварительно общего ключа.

Шифрование можно настроить для соединений:

- Между сервером Zabbix, прокси Zabbix, агентом Zabbix, веб-службой Zabbix, утилитами zabbix_sender и zabbix_get
- С базой данных Zabbix из веб-интерфейса Zabbix и server/proxy
- Между веб-интерфейсом Zabbix и сервером Zabbix

Шифрование является необязательным и настраивается для отдельных компонентов:

- Некоторые прокси и агенты можно настроить на использование шифрования на основе сертификатов с сервером, тогда как другие могут использовать шифрование на основе предварительно общего ключа, а еще другие могут продолжать использовать незашифрованные соединения (как и раньше).
- Сервер (прокси) может использовать разные настройки шифрования для разных узлов сети.

Программы-демоны Zabbix используют один прослушивающий порт для зашифрованных и незашифрованных входящих соединений. Добавление шифрования не требует открытия новых портов на межсетевых экранах.

**Ограничения**

- Закрытые ключи хранятся в виде обычного текста в файлах, доступных для чтения компонентам Zabbix во время запуска.
- Предварительно общие ключи вводятся во веб-интерфейсе Zabbix и хранятся в базе данных Zabbix в виде обычного текста.
- Встроенное шифрование не защищает обмен данными между веб-сервером, на котором работает веб-интерфейс Zabbix, и веб-браузером пользователя.
- В настоящее время каждое зашифрованное соединение открывается с полным TLS-handshake, кэширование сеансов и tickets не реализованы.
- Добавление шифрования увеличивает время проверки элементов данных и выполнения действий в зависимости от задержки сети:
  - Например, если задержка пакета составляет 100 мс, то открытие TCP-соединения и отправка незашифрованного запроса занимает около 200 мс. При использовании шифрования для установления TLS-соединения добавляется около 1000 мс.
  - Может потребоваться увеличить значения тайм-аутов, иначе некоторые элементы данных и действия, выполняющие удаленные скрипты на агентах, могут работать с незашифрованными соединениями, но завершаться по тайм-ауту при использовании шифрования.
- Шифрование не поддерживается сетевым обнаружением. Проверки агента Zabbix, выполняемые в рамках сетевого обнаружения, будут незашифрованными, и если агент Zabbix настроен на отклонение незашифрованных соединений, такие проверки не будут успешными.

**Компиляция Zabbix с поддержкой шифрования** Для поддержки шифрования Zabbix должен быть скомпилирован и слинкован с одной из поддерживаемых криптобиблиотек:

- GnuTLS — с версии 3.1.18
- OpenSSL — версии 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x
- LibreSSL — протестировано с версиями 2.7.4, 2.8.2:
  - LibreSSL 2.6.x не поддерживается;

- LibreSSL поддерживается как совместимая замена для OpenSSL; новые функции API `tls_*()`, специфичные для LibreSSL, не используются. Компоненты Zabbix, скомпилированные с LibreSSL, не смогут использовать PSK, могут использоваться только сертификаты.

**Note:**
Вы можете найти более подробную информацию о настройке SSL для веб-интерфейса Zabbix по ссылке на наилучшие практики.

Библиотека выбирается при помощи соответствующей опции в скрипте «configure»:

- `--with-gnutls[=DIR]`
- `--with-openssl[=DIR]` (также используется и для LibreSSL)

Например, чтобы сконфигурировать исходные коды для сервера и агента с *OpenSSL*, вы можете использовать что-то вроде:

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

Разные компоненты Zabbix можно скомпилировать с различными криптобиблиотеками (например, сервер с *OpenSSL*, агент с *GnuTLS*).

**Attention:**
Если вы планируете использовать общие ключи (pre-shared keys, PSK), рассмотрите возможность использования библиотек *GnuTLS* или *OpenSSL 1.1.0* (или новее) с компонентами Zabbix, использующими PSK. Библиотеки *GnuTLS* и *OpenSSL 1.1.0* поддерживают наборы шифров PSK с совершенной прямой секретностью (Perfect forward secrecy). Более старые версии библиотеки *OpenSSL* (1.0.1, 1.0.2c) также поддерживают PSK, но доступные наборы шифров PSK не обеспечивают совершенную прямую секретность.

**Управление зашифрованными соединениями** Соединения в Zabbix могут использовать:

- отсутствие шифрования (по умолчанию)
- шифрование RSA на основе сертификатов
- шифрование на основе PSK

Имеется два важных параметра, которые используются, чтобы указать шифрование между компонентами Zabbix:

- TLSConnect — указывает, какое шифрование использовать для исходящих соединений (unencrypted, PSK или certificate);
- TLSAccept — указывает, какие виды соединений разрешены для входящих соединений (unencrypted, PSK или certificate). Можно указать одно или несколько значений.

`TLSConnect` используется в файлах конфигурации Zabbix прокси (в активном режиме, задаёт только подключения к серверу) и Zabbix агентов (для активных проверок). В веб-интерфейсе Zabbix эквивалентом параметру TLSConnect является поле *Подключения к узлу сети* на вкладке *Сбор данных → Узлы сети → <какой-то узел сети> → Шифрование (Data collection → Hosts → <somehost> → Encryption)* и поле *Подключения к прокси (Connections to proxy)* на вкладке *Администрирование → Прокси → <какой-то прокси> → Шифрование (Administration → Proxies → <some proxy> → Encryption)*. Если настроенный тип шифрования для соединения завершится неудачей, другие типы шифрования не будут опробованы

`TLSAccept` используется в файлах конфигурации Zabbix прокси (в пассивном режиме, задаёт только соединения от сервера) и Zabbix агента (для пассивных проверок). В веб-интерфейсе Zabbix эквивалентом параметра TLSAccept является поле *Соединения с узла сети (Connections from host)* на вкладке *Сбор данных → Узлы сети → <какой-то узел сети> → Шифрование (Data collection → Hosts → <somehost> → Encryption)* и поля *Соединения с прокси (Connections from proxy)* на вкладке *Администрирование → Прокси → <какой-то прокси> → Шифрование (Administration → Proxies → <some proxy> → Encryption)*.

Как правило, вы настраиваете только один тип шифрования для входящих подключений. Но вы можете захотеть переключить режим шифрования, например, с незашированного на основанный на сертификатах с минимальным временем простоя и с возможностью отката. Чтобы достичь этого:

- Выставьте `TLSAccept=unencrypted,cert` в файле конфигурации агента Zabbix и перезапустите агента.
- Проверьте подключение от zabbix_get к агенту с использованием сертификата. Если подключение работает, вы можете перенастроить шифрование у этого агента в веб-интерфейсе Zabbix на вкладке *Сбор данных → Узлы сети → <какой-то узел сети> → Шифрование (Data collection → Hosts → <some host> →Encryption)*, переключив настройку *Подключения к узлу сети (Connections to host)* на «Сертификат (Certificate)».
- Когда кэш конфигурации сервера обновится (и обновится конфигурация прокси, если узел сети наблюдается через прокси), тогда подключения к этому агенту будут зашифрованными.

- Если всё работает как ожидается, вы можете задать `TLSAccept=cert` в файле конфигурации агента и перезапустить Zabbix агента. Теперь агент будет принимать только зашифрованные подключения на основе сертификатов. Незашифрованные и основанные на PSK подключения будут отклонены.

Шифрование на сервере и прокси работает аналогичным образом. Если в веб-интерфейсе Zabbix в настройке узла сети *Соединения с узла сети (Connections from host)* задано равным «Сертификат (Certificate)», тогда от агента (активные проверки) и zabbix_sender (траппер элементы данных) будут приниматься только зашифрованные соединения на основе сертификатов.

Скорее всего, вы настроите входящие и исходящие соединения на использование одного типа шифрования или без шифрования вовсе. Но, технически, имеется возможность настроить шифрование асимметрично, например, шифрование на основе сертификатов для входящих подключений и на основе PSK для исходящих подключений.

Настройки шифрования по каждому узлу сети отображаются в веб-интерфейсе Zabbix *Сбор данных →Узлы сети (Data collection → Hosts)*, в столбце *Шифрование агента (Agent encryption)*. Примеры отображения настроек:

| Пример | Подключения К узлу сети | Разрешённые подключения ОТ узла сети | Отклонённые подключения С узла сети |
|---|---|---|---|
| NONE | Незашифровано | Незашифровано | Зашифровано, на основе сертификата и PSK |
| CERT NONE PSK CERT | Зашифровано на основе сертификата | Зашифровано на основе сертификата | Незашифровано и зашифровано на основе PSK |
| PSK NONE PSK CERT | Зашифровано на основе PSK | Зашифровано на основе PSK | Незашифровано и зашифровано на основе сертификата |
| PSK NONE PSK CERT | Зашифровано на основе PSK | Незашифровано и зашифровано на основе PSK | Зашифровано на основе сертификата |
| CERT NONE PSK CERT | Зашифровано на основе сертификата | Незашифровано, зашифровано на основе PSK или зашифровано на основе сертификата | - |

**Attention:**
По умолчанию используются незашифрованные подключения. Шифрование необходимо настраивать по каждому узлу сети и прокси отдельно.

**zabbix_get и zabbix_sender с шифрованием** Смотрите страницы помощи zabbix_get и zabbix_sender по использованию этих утилит при наличии шифрования.

**Алгоритмы шифрования** Алгоритмы по умолчанию настраиваются внутри в процессе запуска Zabbix.

Также поддерживаются и заданные пользователем алгоритмы шифрования для GnuTLS и OpenSSL. Пользователи могут настраивать алгоритмы в соответствии со своими политиками безопасности. Использование этой возможности опционально (встроенные алгоритмы по умолчанию всё ещё работают).

Для криптобиблиотек, скомпилированных с настройками по умолчанию, встроенные в Zabbix правила обычно дают в итоге следующие алгоритмы шифрования (от более высокого уровня приоритета к более низкому):

| Библиотека | Алгоритмы шифрования сертификатов | Алгоритмы шифрования PSK |
|---|---|---|
| *GnuTLS 3.1.18* | TLS_ECDHE_RSA_AES_128_GCM_SHA256<br>TLS_ECDHE_RSA_AES_128_CBC_SHA256<br>TLS_ECDHE_RSA_AES_128_CBC_SHA1<br>TLS_RSA_AES_128_GCM_SHA256<br>TLS_RSA_AES_128_CBC_SHA256<br>TLS_RSA_AES_128_CBC_SHA1 | TLS_ECDHE_PSK_AES_128_CBC_SHA256<br>TLS_ECDHE_PSK_AES_128_CBC_SHA1<br>TLS_PSK_AES_128_GCM_SHA256<br>TLS_PSK_AES_128_CBC_SHA256<br>TLS_PSK_AES_128_CBC_SHA1 |
| *OpenSSL 1.0.2c* | ECDHE-RSA-AES128-GCM-SHA256<br>ECDHE-RSA-AES128-SHA256<br>ECDHE-RSA-AES128-SHA<br>AES128-GCM-SHA256<br>AES128-SHA256<br>AES128-SHA | PSK-AES128-CBC-SHA |

| Библиотека | Алгоритмы шифрования сертификатов | Алгоритмы шифрования PSK |
|---|---|---|
| *OpenSSL 1.1.0* | ECDHE-RSA-AES128-GCM-SHA256<br>ECDHE-RSA-AES128-SHA256<br>ECDHE-RSA-AES128-SHA<br>AES128-GCM-SHA256<br>AES128-CCM8<br>AES128-CCM<br>AES128-SHA256<br>AES128-SHA | ECDHE-PSK-AES128-CBC-SHA256<br>ECDHE-PSK-AES128-CBC-SHA<br>PSK-AES128-GCM-SHA256<br>PSK-AES128-CCM8<br>PSK-AES128-CCM<br>PSK-AES128-CBC-SHA256<br>PSK-AES128-CBC-SHA |
| *OpenSSL 1.1.1d* | TLS_AES_256_GCM_SHA384<br>TLS_CHACHA20_POLY1305_SHA256<br>TLS_AES_128_GCM_SHA256<br>ECDHE-RSA-AES128-GCM-SHA256<br>ECDHE-RSA-AES128-SHA256<br>ECDHE-RSA-AES128-SHA<br>AES128-GCM-SHA256<br>AES128-CCM8<br>AES128-CCM<br>AES128-SHA256<br>AES128-SHA | TLS_CHACHA20_POLY1305_SHA256<br>TLS_AES_128_GCM_SHA256<br>ECDHE-PSK-AES128-CBC-SHA256<br>ECDHE-PSK-AES128-CBC-SHA<br>PSK-AES128-GCM-SHA256<br>PSK-AES128-CCM8<br>PSK-AES128-CCM<br>PSK-AES128-CBC-SHA256<br>PSK-AES128-CBC-SHA |

**Алгоритмы шифрования, заданные пользователем** Встроенные критерии выбора алгоритмов шифрования могут быть переопределены алгоритмами шифрования, заданными пользователем.

**Attention:**
Алгоритмы шифрования, заданные пользователем, — это возможность, предназначенная для продвинутых пользователей, которые понимают алгоритмы шифрования TLS, их безопасность и последствия ошибок, и которые разбираются в устранении неполадок с TLS.

Встроенные критерии выбора алгоритмов шифрования могут быть переопределены, используя следующие параметры:

| Область действия переопределения | Параметр | Значение | Описание |
|---|---|---|---|
| Выбор алгоритмов шифрования для сертификатов | TLSCipherCert13 | Корректные строки шифров OpenSSL 1.1.1 для протокола TLS 1.3 (их значения передаются функции OpenSSL `SSL_CTX_set_ciphersuites()`). | Критерии выбора алгоритмов шифрования при использовании сертификатов для TLS 1.3.<br><br>Только OpenSSL 1.1.1 или новее. |
| | TLSCipherCert | Корректные строки шифров OpenSSL для TLS 1.2 или корректные строки приоритетов GnuTLS. Их значения передаются функциям `SSL_CTX_set_cipher_list()` или `gnutls_priority_init()`, соответственно. | Критерии выбора алгоритмов шифрования при использовании сертификатов для TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL). |
| Выбор алгоритмов шифрования для PSK | TLSCipherPSK13 | Корректные строки шифров OpenSSL 1.1.1 для протокола TLS 1.3 (их значения передаются функции OpenSSL `SSL_CTX_set_ciphersuites()`). | Критерии выбора алгоритмов шифрования при использовании PSK для TLS 1.3<br><br>Только OpenSSL 1.1.1 или новее. |
| | TLSCipherPSK | Корректные строки шифров OpenSSL для TLS 1.2 или корректные строки приоритетов GnuTLS. Их значения передаются функциям `SSL_CTX_set_cipher_list()` или `gnutls_priority_init()`, соответственно. | Критерии выбора алгоритмов шифрования при использовании PSK для TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL). |

| Область действия переопределения | Параметр | Значение | Описание |
|---|---|---|---|
| Комбинированный список алгоритмов шифрования для сертификатов и PSK | TLSCipherAll13 | Корректные строки шифров OpenSSL 1.1.1 для протокола TLS 1.3 (их значения передаются функции OpenSSL `SSL_CTX_set_ciphersuites()`). | Критерии выбора алгоритмов шифрования для TLS 1.3<br><br>только OpenSSL 1.1.1 или новее. |
| | TLSCipherAll | Корректные строки шифров OpenSSL для TLS 1.2 или корректные строки приоритетов GnuTLS. Их значения передаются функциям `SSL_CTX_set_cipher_list()` или `gnutls_priority_init()`, соответственно. | Критерии выбора алгоритмов шифрования для TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL). |

Чтобы переопределить выбор алгоритмов шифрования в утилиах zabbix_get и zabbix_sender — используйте параметры командной строки:

- `--tls-cipher13`
- `--tls-cipher`

Эти параметры опциональны. Если параметр не указан, используется внутреннее значение по умолчанию. Если параметр указан, он не может быть пустым.

Если настройка значения параметров TLSCipher* в криптобиблиотеке завершается неудачей, то сервер, прокси или агент не запустятся, а в журнал будет записано сообщение об ошибке.

Крайне важно понимать, когда применим каждый из параметров.

Исходящие соединения

Простейший случай — это исходящие соединения:

- Для исходящих соединений с сертификатом — используйте TLSCipherCert13 или TLSCipherCert
- Для исходящих соединений с PSK — используйте TLSCipherPSK13 или TLSCipherPSK
- В случае утилит zabbix_get и zabbix_sender могут быть использованы параметры командной строки `--tls-cipher13` и `--tls-cipher` (само шифрование недвусмысленно указывается параметром `--tls-connect`)

Входящие соединения

Нменого сложнее с входящими соединениями, т.к. правила зависят от компонентов и конфигурации.

Для Zabbix **агента**:

| Настройка соединений агента | Настройка шифров |
|---|---|
| TLSConnect=cert | TLSCipherCert, TLSCipherCert13 |
| TLSConnect=psk | TLSCipherPSK, TLSCipherPSK13 |
| TLSAccept=cert | TLSCipherCert, TLSCipherCert13 |
| TLSAccept=psk | TLSCipherPSK, TLSCipherPSK13 |
| TLSAccept=cert,psk | TLSCipherAll, TLSCipherAll13 |

Для Zabbix **сервера** и **прокси**:

| Настройка соединений | Настройка шифров |
|---|---|
| Исходящие соединения, используя PSK | TLSCipherPSK, TLSCipherPSK13 |
| Исходящие соединения, используя сертификаты | TLSCipherAll, TLSCipherAll13 |
| Входящие соединения, используя PSK, если у сервера нет сертификата | TLSCipherPSK, TLSCipherPSK13 |
| Входящие соединения, используя PSK, если у сервера есть сертификат | TLSCipherAll, TLSCipherAll13 |

В двух таблицах выше можно увидеть некоторые шаблоны:

- TLSCipherAll и TLSCipherAll13 могут быть указаны, только если используется комбинированный список шифров на основе сертификатов **и** на основе PSK. Есть два случая, когда это имеет место: сервер (прокси) с настроенным сертификатом (шифры PSK всегда настраиваются на сервере и прокси, если криптобиблиотека поддерживает PSK), агент настраивается на приём как на основе сертификатов, так и PSK;
- в остальных случаях достаточно TLSCipherCert* и/или TLSCipherPSK*.

Следующие таблицы показывают встроенные по умолчанию значения `TLSCipher*`. Они могут являться хорошей отправной точкой для ваших собственных кастомизированных значений.

| Параметр | 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 |

| Параметр | 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] Значения по умолчанию отличаются для более старых версий OpenSSL (1.0.1, 1.0.2, 1.1.0), для LibreSSL и если OpenSSL скомпилирован без поддержки PSK.

**Примеры заданных пользователем алгоритмов шифрования**

Смотрите ниже следующие примеры заданных пользователем алгоритмов шифрования:

- Тестирование строк шифра и разрешение только алгоритмов PFS
- Переключение от AES128 на AES256

Тестирование строк шифра и разрешение только алгоритмов PFS

Чтобы видеть выбираемые алгоритмы шифрования, вам нужно выставить в файле конфигурации параметр 'DebugLevel=4' либо использовать опцию `-vv` для утилиты zabbix_sender.

Возможно, для получения желаемых алгоритмов шифрования придётся немного поэкспериментировать с параметрами `TLSCipher*`. Неудобно перезапускать Zabbix сервер, прокси или агента множество раз только для того, чтобы подогнать параметры `TLSCipher*`. Более удобным вариантом является использование утилиты zabbix_sender или команды `openssl`. Давайте рассмотрим оба.

**1.** Использование zabbix_sender.

Давайте сделаем тестовый файл конфигурации, например, `/home/zabbix/test.conf`, с таким же синтаксисом, как и файл `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
```

Для данного примера вам нужны корректные сертификаты Центра Авторизации (CA) и агента, а также общий ключ (PSK). Подправьте имена и пути файлов сертификатов и PSK в соответствии с вашей средой.

Если вы не используете сертификаты, а только общий ключ, тестовый файл может быть проще:

```
Hostname=nonexisting
ServerActive=nonexisting

TLSConnect=psk
TLSPSKIdentity=nonexisting
TLSPSKFile=/home/zabbix/agentd.psk
```

Выбираемые алгоритмы шифрования могуть быть видны путём запуска утилиты zabbix_sender (пример скомпилирован с 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
```

Здесь вы можете увидеть алгоритмы шифрования, выбранные по умолчанию. Эти значения по умолчанию выбираются, чтобы обеспечить совместимость с Zabbix агентами, работающими на системах с более старыми версиями OpenSSL (от 1.0.1).

С более новыми системами вы можете предпочесть усилить безопасность, разрешая только некоторые алгоритмы шифрования, скажем, только алгоритмы с PFS (Perfect Forward Secrecy, совершенная прямая секретность). Давайте попробуем разрешить только алгоритмы с PFS, используя параметры TLSCipher*.

**Attention:**
Результатом будет несовместимость с системами, использующими OpenSSL 1.0.1 и 1.0.2, если используется общий ключ PSK. Шифрование на основе сертификата должно работать.

Добавим две строки в файл конфигурации test.conf:

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
```

и проверим снова:

```
$ 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
```

Списки «certificate ciphersuites» и «PSK ciphersuites» поменялись — теперь они короче, чем раньше; как и ожидалось, содержат только алгоритмы TLS 1.3 и TLS 1.2 ECDHE-*.

**2.** TLSCipherAll и TLSCipherAll13 нельзя протестировать с помощью zabbix_sender; они не влияют на значение «certificate and PSK ciphersuites», показанное в примере выше. Чтобы подкорректировать TLSCipherAll и TLSCipherAll13, вам нужно экспериментировать с агентом, прокси или сервером.

Так, чтобы разрешить только алгоритмы PFS, вам, возможно, потребуется добавить до трёх параметров

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
TLSCipherAll=EECDH+aRSA+AES128:kECDHEPSK+AES128
```

в zabbix_agentd.conf, zabbix_proxy.conf и zabbix_server_conf, если каждый из них имеет сконфигурированный сертификат, а агент имеет ещё и PSK.

Если ваша среда Zabbix использует только шифрование на основе общего ключа и не использует сертификаты, то достаточно только одной строки:

```
TLSCipherPSK=kECDHEPSK+AES128
```

Теперь, когда вы понимаете, как это работает, вы можете проверить выбор алгоритма шифрования даже извне Zabbix, командой openssl. Давайте проверим значения всех трёх параметров 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
```

Вы можете предпочесть openssl ciphers с опцией -V для более подробного вывода:

```
$ 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
```

Аналогичным образом вы можете протестировать строки приоритета для 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
```

Переключение с AES128 на AES256

Zabbix использует AES128 как встроенное умолчание для данных. Предположим, что вы используете сертификаты и хотите переключиться на AES256, на OpenSSL 1.1.1.

Это может быть достигнуто путём добавления соответствующих параметров в `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:**
Несмотря на то, что будут использованы только алгоритмы шифрования на основе сертификатов, параметры `TLSCipherPSK*` также определены, чтобы не использовались их значения по умолчанию, для более широкой совместимости включающие менее безопасные шифры. Алгоритмы шифрования PSK нельзя полностью отключить для сервера/прокси.

И в `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
```

**1. Использование сертификатов**

Обзор

Zabbix может использовать RSA сертификаты в формате PEM, подписанные публичным или внутренним центром сертификации (certificate authority, CA).

Проверка сертификата выполняется с использованием заранее подготовленного сертификата CA. Опционально можно использовать списки отозванных сертификатов (Certificate Revocation Lists, CRL).

Каждый компонент Zabbix может иметь только один настроенный сертификат.

Для получения более подробной информации о том, как настроить и управлять внутренним CA, как генерировать и подписывать запросы на сертификаты, как отзывать сертификаты, — обратитесь к руководствам, таким как OpenSSL PKI Tutorial v2.0 [en] .

Тщательно продумывайте и тестируйте ваши расширения сертификатов. Для получения более подробной информации смотри Ограничения при использовании расширений сертификатов X.509 v3.

Параметры настройки сертификатов

| Параметр | Обязателен | Описание |
|---|---|---|
| *TLSCAFile* | да | Абсолютный путь к файлу, который содержит сертификаты CA верхнего уровня для верификации сертификата узла.<br>При наличии цепочки сертификатов с несколькими членами, они должны быть упорядочены: сначала сертификаты CA нижних уровней, за ними следуют сертификаты CA более высокого уровня.<br>Сертификаты из нескольких CA можно включать в один файл. |
| *TLSCRLFile* | нет | Абсолютный путь к файлу, который содержит Списки отозванных сертификатов (CRL). |
| *TLSCertFile* | да | Абсолютный путь к файлу, который содержит сертификат.<br>В случае цепочки сертификатов с несколькими членами они должны быть упорядочены: сначала сертификат сервера, прокси или агента, затем сертификаты CA нижних уровней и в завершение — сертификаты CA более высокого уровня. |
| *TLSKeyFile* | да | Абсолютный путь к файлу, который содержит закрытый ключ (private key).<br>Убедитесь, что этот файл доступен для чтения только пользователю Zabbix, выставив соответствующие права доступа. |
| *TLSServerCertIssuer* | нет | Разрешённый эмитент сертификата сервера. |
| *TLSServerCertSubject* | нет | Разрешённый субъект сертификата сервера. |

Примеры конфигурации

После настройки необходимых сертификатов настройте компоненты Zabbix для использования шифрования на основе сертификатов.

Ниже приведены подробные шаги по настройке:

- сервер Zabbix
- прокси Zabbix
- агент Zabbix

Zabbix сервер

1. Подготовьте файл сертификата CA.

Для проверки сертификатов хостов Zabbix сервер должен иметь доступ к файлу, содержащему самоподписанные сертификаты корневого CA верхнего уровня. Например, если нужны сертификаты от двух независимых корневых CA, поместите их в файл `/home/zabbix/zabbix_ca_file.crt`:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
            ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ...
        X509v3 extensions:
            X509v3 Key Usage: critical
```

```
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE
            ...
-----BEGIN CERTIFICATE-----
MIID2jCCAsKgAwIBAgIBATANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
....
9wEzdN8uTrqoyU78gi12npLj08LegRKjb5hFTVmO
-----END CERTIFICATE-----
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 1 (0x1)
    Signature Algorithm: sha1WithRSAEncryption
        Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
            ...
        Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
            ....
        X509v3 extensions:
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Basic Constraints: critical
                CA:TRUE
            ....
-----BEGIN CERTIFICATE-----
MIID3DCCAsSgAwIBAgIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQB
...
vdGNYoSfvu41GQAR5Vj5FnRJRzv5XQOZ3B6894GY1zY=
-----END CERTIFICATE-----
```

2. Поместите сертификат или цепочку сертификатов Zabbix сервера в файл, например, `/home/zabbix/zabbix_server.crt`. Первый сертификат — это сертификат Zabbix сервера, за ним — сертификат промежуточного 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/wUtcVIvWXdMIFJMOHw==
-----END CERTIFICATE-----
```

**Note:**
Как для клиентских, так и для серверных сертификатов используйте только вышеупомянутые атрибуты, чтобы избежать влияния на процесс проверки сертификата. Например, OpenSSL может не установить зашифрованное соединение, если используются расширения *Альтернативное имя субъекта X509v3* (*X509v3 Subject Alternative Name*) или *Тип сертификата Netscape* (*Netscape Cert Type*). Для получения дополнительной информации смотрите: Ограничения при использовании расширений сертификатов X.509 v3.

3. Поместите закрытый ключ Zabbix сервера в файл, например, `/home/zabbix/zabbix_server.key`:

```
-----BEGIN PRIVATE KEY-----
MIIEwAIBADANBgkqhkiG9w0BAQEFAASCBKowggSmAgEAAoIBAQC9tIXIJoVnNXDl
...
IJLkhbybBYEf47MLhffWa7XvZTY=
-----END PRIVATE KEY-----
```

4. Отредактируйте параметры конфигураци TLS в файле конфигурации Zabbix сервера:

```
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_server.crt
TLSKeyFile=/home/zabbix/zabbix_server.key
```

Zabbix прокси

1. Подготовьте файлы с сертификатами CA верхнего уровня, сертификатом (или цепочкой сертификатов) Zabbix прокси и закрытым ключом, как описано в разделе Zabbix сервер. Затем соответственно отредактируйте параметры `TLSCAFile`, `TLSCertFile` и `TLSKeyFile` в файле конфигурации Zabbix прокси.

2. Отредактируйте дополнительные параметры TLS в файле конфигурации Zabbix proxy:

- Для активного прокси: `TLSConnect=cert`
- Для пассивного прокси: `TLSAccept=cert`

**Note:**
Чтобы повысить безопасность прокси, вы также можете задать параметры `TLSServerCertIssuer` и `TLSServerCertSubject`. Для получения дополнительной информации смотрите Ограничение разрешённых Эмитента и Субъекта сертификата.

В конечном итоге параметры TLS в файле конфигурации прокси могут выглядеть следующим образом:

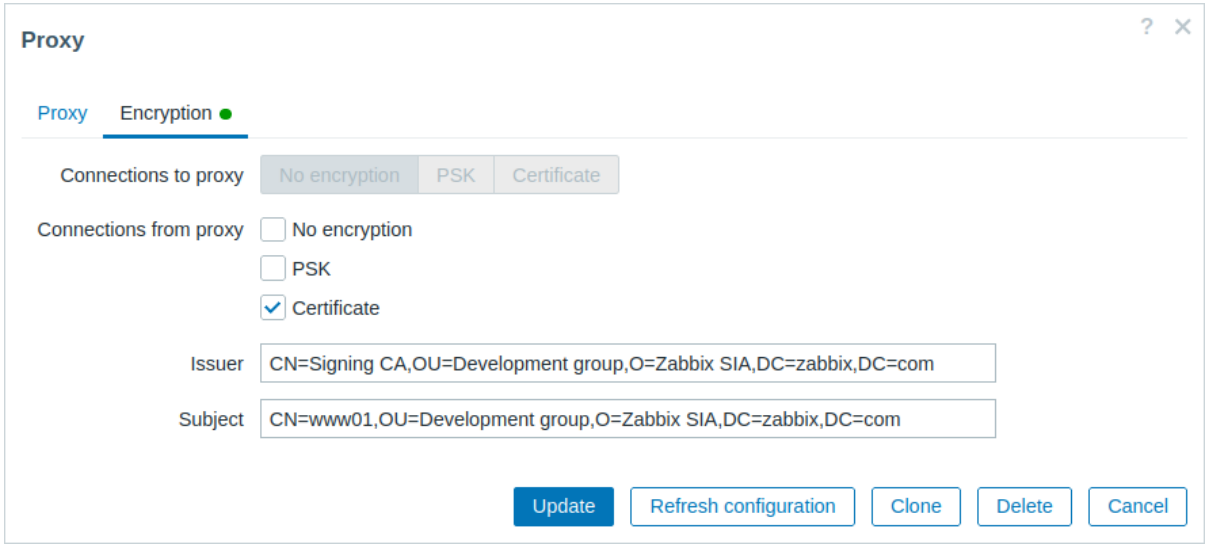
```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_proxy.crt
TLSKeyFile=/home/zabbix/zabbix_proxy.key
```

3. Настройте шифрование для этого прокси в веб-интерфейсе Zabbix:

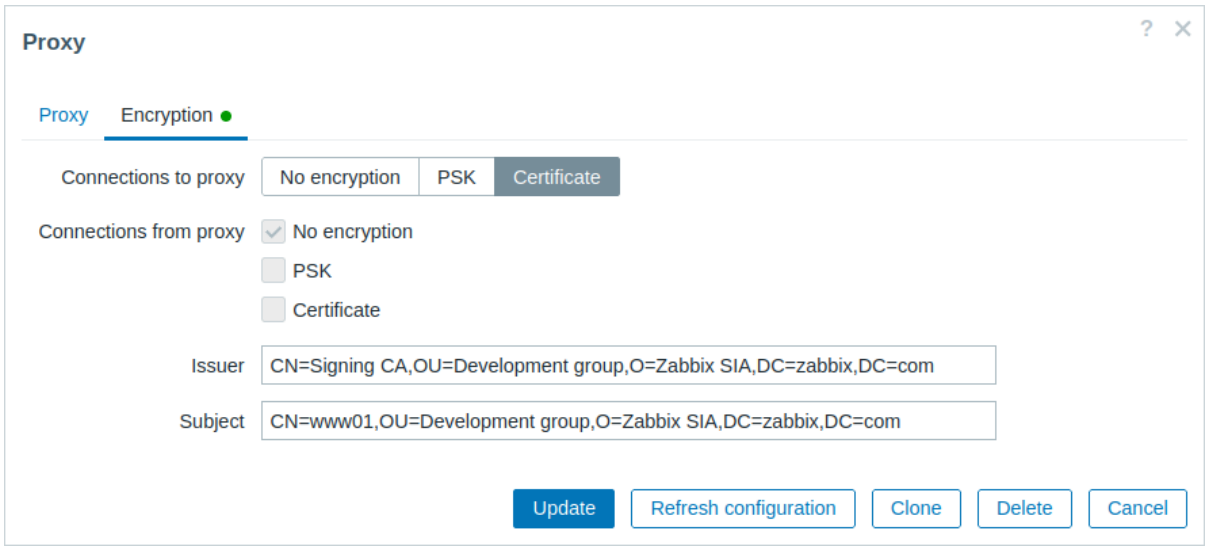
- Перейдите к: *Администрирование → Прокси (Administration → Proxies)*
- Выберите прокси и нажмите на вкладку *Шифрование (Encryption)*

В примере ниже поля *Эмитент (Issuer)* и *Субъект (Subject)* заполнены. За дополнительной информацей о том, для чего и как использовать эти поля, обратитесь к Ограничение разрешённых эмитента и субъекта сертификата.

Для активного прокси:




Для пассивного прокси:




Zabbix агент

1. Подготовьте файлы с сертификатами CA верхнего уровня, сертификатом (или цепочкой сертификатов) Zabbix агента и закрытым ключом, как описано в разделе Zabbix сервер. Затем отредактируйте соответственно параметры `TLSCAFile`, `TLSCertFile` и `TLSKeyFile` в файле конфигурации Zabbix агента.

2. Отредактируйте дополнительные параметры TLS в файле конфигурации Zabbix агента:

- Для активного агента: `TLSConnect=cert`
- Для пассивного агента: `TLSAccept=cert`

**Note:**
Чтобы улучшить безопасность агента, можно указать параметры `TLSServerCertIssuer` и `TLSServerCertSubject`. Для получения дополнительной информации смотрите Ограничение разрешённых Эмитента и Субъекта сертификата.

В конечном итоге параметры TLS в файле конфигурации агента могут выглядеть следующим образом. Обратите внимание, что пример предполагает, что хост наблюдается через прокси, поэтому в качестве Субъекта сертификата

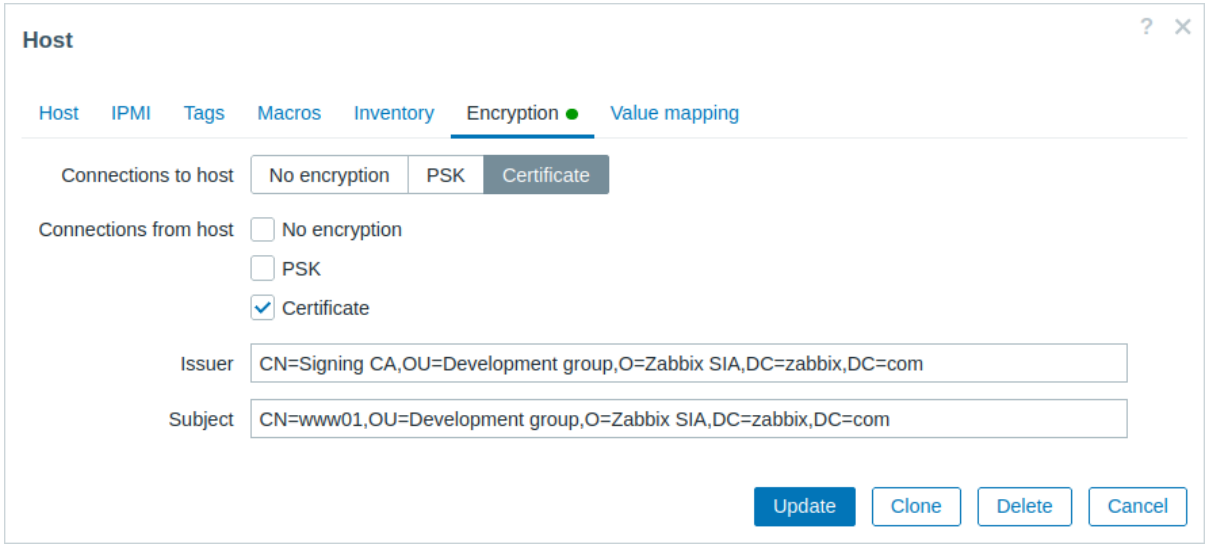
указан прокси:

```
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. Настройте шифрование для узла сети, наблюдаемого этим агентом, в веб-интерфейсе Zabbix:

- Перейдите к: *Сбор данных → Узлы сети (Data collection → Hosts)*
- Выберите узел сети и нажмите на вкладку *Шифрование (Encryption)*

В примере ниже поля *Эмитент (Issuer)* и *Субъект (Subject)* заполнены. Дополнительную информацию о том, для чего и как использовать эти поля, смотрите Ограничение разрешённых Эмитента и Субъекта сертификата.




Веб-сервис Zabbix

1. Подготовьте файлы с корневыми CA-сертификатами, сертификатом/цепочкой сертификатов веб-сервиса Zabbix и закрытым ключом, как описано в разделе сервер Zabbix. Затем соответствующим образом отредактируйте параметры `TLSCAFile`, `TLSCertFile` и `TLSKeyFile` в файле конфигурации веб-сервиса Zabbix.

2. Отредактируйте дополнительный TLS-параметр в файле конфигурации веб-сервиса Zabbix: `TLSAccept=cert`

TLS-параметры в итоговом файле конфигурации веб-сервиса могут выглядеть следующим образом:

```
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_web_service.crt
TLSKeyFile=/home/zabbix/zabbix_web_service.key
```

3. Настройте сервер Zabbix для подключения к веб-сервису Zabbix, настроенному на использование TLS, отредактировав параметр `WebServiceURL` в файле конфигурации сервера Zabbix:

```
WebServiceURL=https://example.com:443/report
```

Ограничение разрешённых Эмитента и Субъекта сертификата

Когда два компонента Zabbix (например, сервер и агент) устанавливают TLS соединение, они оба проверяют сертификаты друг друга. Если сертификат узла подписан доверенным CA (с предварительно подготовленным сертификатом верхнего уровня в `TLSCAFile`), является действительным, он не истёк и проходит некоторые другие проверки, то коммуникация между компонентами может продолжаться. В этом простейшем случае эмитент и субъект сертификата не проверяются.

Однако, здесь имеется риск: кто угодно при наличии действительного сертификата может выдать себя за другого (например, сертификат хоста может быть использован, чтобы выдать себя за сервер). Хотя это может быть приемлемо в небольших средах, где сертификаты подписываются выделенным внутренним CA, а риск выдачи себя за кого-либо низок, этого может быть недостаточно в более крупных или более чувствительных к безопасности средах.

Если ваш CA верхнего уровня выдаёт другие сертификаты, которые не должны приниматься Zabbix, или вы хотите снизить риск действий от чужого имени, вы можете ограничить разрешённые сертификаты, указав их эмитента и субъект.

Например, в файле конфигурации Zabbix прокси вы можете указать:

```
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
```

С этими настройками активный прокси не будет общаться с Zabbix сервером, чей сертификат имеет других эмитента и субъект. Аналогично, пассивный прокси не примет запросы от такого сервера.

Правила сопоставления строк `Issuer` и `Subject`

Правила сопоставления строк `Issuer` и `Subject` следующие:

- Строки `Issuer` и `Subject` проверяются независимо. Обе строки необязательны.
- Неуказанная строка означает, что принимается любая строка.
- Строки сравниваются *как есть* и должны совпадать точно.
- Поддерживаются символы UTF-8. Однако подстановочные знаки (*) и регулярные выражения не поддерживаются.
- Реализованы следующие требования RFC 4514 - символы, требующие экранирования (с помощью обратной косой черты '\', U+005C):
  - в любом месте строки: '"' (U+0022), '+' (U+002B), ',' (U+002C), ';' (U+003B), '<' (U+003C), '>' (U+003E), '\\' (U+005C);
  - в начале строки: пробел (' ', U+0020) или знак номера ('#', U+0023);
  - в конце строки: пробел (' ', U+0020).
- Символы null (U+0000) не поддерживаются. Если встречается символ null, сопоставление завершается неудачей.
- Стандарты RFC 4517 и RFC 4518 не поддерживаются.

Например, если строки организации (O) в `Issuer` и `Subject` содержат завершающие пробелы, а строка организационного подразделения (OU) в `Subject` содержит двойные кавычки, эти символы необходимо экранировать:

```
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
```

Порядок и форматирование полей

Zabbix следует рекомендациям RFC 4514, в которых для этих полей указан "обратный" порядок: начиная с полей самого низкого уровня (CN), далее поля среднего уровня (OU, O) и заканчивая полями самого высокого уровня (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
```

В отличие от этого, OpenSSL по умолчанию отображает строки `Issuer` и `Subject` в порядке от верхнего уровня к нижнему. В следующем примере поля `Issuer` и `Subject` начинаются с верхнего уровня (DC) и заканчиваются полем нижнего уровня (CN). Форматирование с пробелами и разделителями полей также зависит от используемых параметров, поэтому оно не будет соответствовать формату, требуемому 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
```

Чтобы правильно отформатировать строки *Issuer* и *Subject* для Zabbix, вызовите OpenSSL со следующими параметрами:

```
$ 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
```

После этого вывод будет в обратном порядке, с разделением запятыми, и его можно будет использовать в конфигурационных файлах Zabbix и веб-интерфейсе:

```
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
```

Ограничения при использовании расширений сертификатов X.509 v3

При реализации сертификатов X.509 v3 в Zabbix некоторые расширения могут поддерживаться не полностью или могут привести к несовместимому поведению.

**Расширение «Альтернативное имя субъекта» (Subject Alternative Name)**

Zabbix не поддерживает расширение «Subject Alternative Name», которое используется для указания альтернативных имён DNS — таких как IP адрес или e-mail адрес. В Zabbix проверяется только значение поля «Субъект» сертификата (смотри Ограничение разрешённых Эмитента и Субъекта сертификата). Если сертификат содержит поле `subjectAltName`, то результат проверки сертификата может зависеть от конкретной комбинации наборов криптографических инструментов, используемых для компиляции компонентов Zabbix. В результате, в зависимости от этих комбинаций, Zabbix может либо принимать, либо отклонять сертификаты.

**Расширение «Использование Расширенного Ключа» (Extended Key Usage)**

Zabbix поддерживает расширение *Extended Key Usage*. Однако, если оно используется, то, как правило, требуется, чтобы были указаны как *clientAuth* (для аутентификации клиента TLS WWW), так и *serverAuth* (для аутентификации сервера TLS WWW). Например:

- При пассивных проверках, где Zabbix агент выступает в роли TLS сервера, в сертификате агента должен присутствовать атрибут *serverAuth*.
- При активных проверках, где агент выступает в качестве TLS клиента, в сертификате агента должен присутствовать атрибут *clientAuth*.

Хотя GnuTLS может выдавать предупреждение о нарушениях использования ключа, он обычно позволяет продолжать соединение, несмотря на эти предупреждения.

**Расширение «Ограничения Имени» (Name Constraints)**

Поддержка расширения *Name Constraints* различается в разных наборах инструментов шифрования. Убедитесь, что выбранный вами набор инструментов поддерживает это расширение. В зависимости от конкретного набора инструментов криптографии, если эта секция помечена как критическая, то это расширение может помешать Zabbix в загрузке сертификатов CA.

Списки отозванных сертификатов (CRL)

Если сертификат скомпрометирован, CA может отозвать его, включив сертификат в список отозванных сертификатов (CRL). Списки CRL можно настраивать в файлах конфигурации сервера, прокси и агента, используя параметр `TLSCRLFile`. Например:

```
TLSCRLFile=/home/zabbix/zabbix_crl_file.crt
```

Файл `zabbix_crl_file.crt` в данном случае может содержать списки CRL от нескольких CA и мог бы выглядеть следующим образом:

```
-----BEGIN X509 CRL-----
MIIB/DCB5QIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixkARkWA2Nv
...
treZeUPjb7LSmZ3K2hpbZN7SoOZcAoHQ3GWd9npuctg=
-----END X509 CRL-----
-----BEGIN X509 CRL-----
MIIB+TCB4gIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQBGRYDY29t
...
CAEebS2CND3ShBedZ8YSil59O6JvaDP61lR5lNs=
-----END X509 CRL-----
```

CRL файл загружается только при запуске Zabbix. Для обновления CRL перезапустите Zabbix.

> **Attention:**
> Если компоненты Zabbix скомпилированы с *OpenSSL* и используются списки CRL, убедитесь, что каждый CA верхнего и промежуточного уровней в цепочках сертификатов имеет соответствующий список CRL (даже если он пустой), включённый в `TLSCRLFile`.

**2. Использование общих ключей (pre-shared keys)**

Обзор

Каждый общий ключ (pre-shared key, PSK) в Zabbix на самом деле является парой из:

- несекретной строки идентификации PSK,
- секретного значения строки PSK.

Строкой идентификации PSK является непустая строка UTF-8. Например, «PSK ID 001 Zabbix agentd». Это уникальное имя, по которому компоненты Zabbix ссылаются на этот конкретный общий ключ. Не помещайте чувствительную информацию в строку идентификации PSK — она передаётся по сети незашифрованной.

Значением PSK является сложно угадываемая строка из шестнадцатеричных цифр, например: «e560cb0d918d26d31b4f642181f5f570a

Ограничения размера

В Zabbix имеется ограничение на размеры идентификатора и значения PSK, в некоторых случаях криптобиблиотеки могут иметь меньшее ограничение:

| Компонент | Макс размер идентификатора PSK | Мин размер значения PSK | Макс размер значения PSK |
|---|---|---|---|
| *Zabbix* | 128 UTF-8 символов | 128-бит (16-байт PSK, введённые как 32 шестнадцатеричных цифры) | 2048-бит (256-байт PSK, введённые как 512 шестнадцатеричных цифр) |
| *GnuTLS* | 128 байт (может включать UTF-8 символы) | - | 2048-бит (256-байт PSK, введённые как 512 шестнадцатеричных цифр) |
| *OpenSSL 1.0.x, 1.1.0* | 127 байт (может включать UTF-8 символы) | - | 2048-бит (256-байт PSK, введённые как 512 шестнадцатеричных цифр) |
| *OpenSSL 1.1.1* | 127 байт (может включать UTF-8 символы) | - | 512-бит (64-байт PSK, введённые как 128 шестнадцатеричных цифр) |
| *OpenSSL 1.1.1a and later* | 127 байт (может включать UTF-8 символы) | - | 2048-бит (256-байт PSK, введённые как 512 шестнадцатеричных цифр) |

**Attention:**
Веб-интерфейс Zabbix позволяет указать до 128 символов строки идентификатора PSK и до 2048-бит PSK вне зависимости от использованной крипто библиотеки.<br> Если некоторые компоненты Zabbix поддерживают меньшую длину, только пользователь несет ответственность за настройку идентификатора и значения PSK в соответствии с разрешённой в этих компонентах длиной.<br> Превышение ограничений длины приведёт к ошибкам в подключениях между компонентами Zabbix.

Перед тем как Zabbix сервер подключится к агенту с использованием общего ключа, сервер поищет идентификатор PSK и значение PSK этого агента в базе данных (в действительности в кэше конфигурации). Принимая подключение, агент использует идентификатор PSK и значение PSK из своего файла конфигурации. Если обе пары имеют одинаковые идентификаторы PSK и значения PSK, соединение может быть успешным.

**Attention:**
Каждая строка идентификации PSK должна иметь в паре с ней только одно значение. Пользователь несет ответственность за обеспечение того, что не будет двух ключей с одинаковыми строками идентификации, но разными значениями. Несоблюдение этого требования может привести к непредсказуемым сбоям связи между компонентами Zabbix, использующими общие ключи с этими строками идентификации PSK.

Генерирование общего ключа

Например, 256-битный (32 байт) PSK можно сгенерировать при помощи следующих команд:

- с *OpenSSL*:

```
$ openssl rand -hex 32
af8ced32dfe8714e548694e2d29e1a14ba6fa13f216cb35c19d0feb1084b0429
```

- с *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
```

Обратите внимание, что команда «psktool» выше генерирует файл базы данных с идентификатором ключа и со связанным с ним значением. Zabbix ожидает в файле с ключом только значение, поэтому строку идентификации и двоеточие («:») нужно оттуда удалить.

Настройка PSK для связи сервер-агент (пример)

На узле сети агента запишите значение PSK в файл, например, `/home/zabbix/zabbix_agentd.psk`. Файл должен содержать PSK в первой текстовой строке, например:

```
1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952
```

Установите права доступа к файлу PSK — он должен быть доступен на чтение только пользователю Zabbix.

Отредактируйте параметры TLS в файле конфигурации агента `zabbix_agentd.conf`, например, задайте:

```
TLSConnect=psk
TLSAccept=psk
TLSPSKFile=/home/zabbix/zabbix_agentd.psk
TLSPSKIdentity=PSK 001
```

Агент будет подключаться к серверу (активные проверки) и принимать от сервера и `zabbix_get` только подключения с использованием PSK. Идентификатор PSK будет "PSK 001".

Перезапустите агента. Теперь вы можете проверить подключение с помощью `zabbix_get`, например:

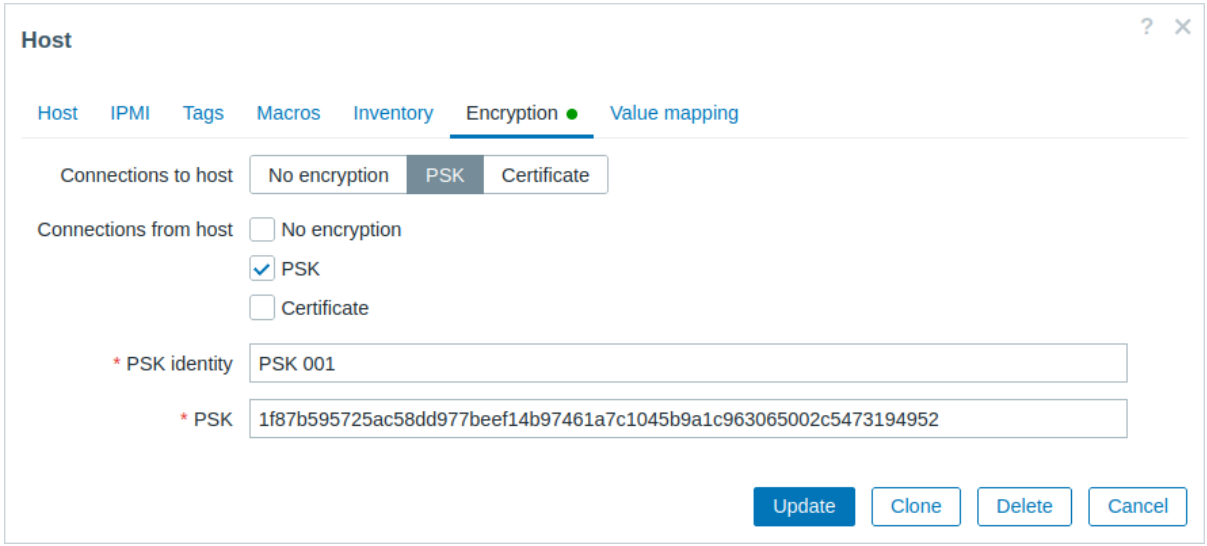
```
zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk --tls-psk-identity="PSK 001" --tl
```

(Чтобы минимизировать время простоя, см. как изменить тип подключения в Управление шифрованием соединения).

Настройте шифрование PSK для этого агента в веб-интерфейсе Zabbix:

- Перейдите в: *Сбор данных → Узлы сети*
- Выберите узел сети и нажмите вкладку **Шифрование**

Пример:




Все обязательные поля ввода отмечены красной звездочкой.

Когда кэш конфигурации будет синхронизирован с базой данных, новые соединения будут использовать PSK. Проверьте журналы сервера и агента на наличие сообщений об ошибках.

Настройка PSK для соединения сервер - активный прокси (пример)

На хосте прокси запишите значение общего ключа в файл, например, `/home/zabbix/zabbix_proxy.psk`. Этот файл должен содержать ключ в первой строке текста, например:

```
e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9
```

Задайте права доступа к PSK файлу — он должен быть доступен для чтения только пользователю Zabbix.

Измените параметры TLS в файле конфигурации `zabbix_proxy.conf`; например, задайте:

```
TLSConnect=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

Прокси будет подключаться к серверу, используя общий ключ. Идентификатором общего ключа будет «PSK 002».

(Чтобы свести к минимуму время простоя, смотрите как изменить тип подключения в Управлении зашифрованными соединениями).

Настройте общий ключ для этого прокси в веб-интерфейсе Zabbix. Перейдите на *Администрирование → Прокси (Administration→Proxies)*, выберите прокси, перейдите на вкладку «Шифрование (Encryption)». Переключите «Соединения с прокси (Connections from proxy)» на PSK. Вставьте в поле «Идентификатор PSK (PSK identity)» значение «PSK 002» и «e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9» в поле «PSK». Нажмите «Обновить (Update)».

Перезапустите прокси. Он начнет использовать зашифрованные соединения к серверу на основе общего ключа. Проверьте файлы журналов сервера и прокси на предмет сообщений об ошибках.

В случае пассивных прокси процедура очень похожа. Разница лишь в том, что в файле конфигурации прокси необходимо задать TLSAccept=psk и переключить «Подключения к прокси (Connections to proxy)» в Zabbix веб-интерфейсе на PSK.

**3. Решение проблем**

Общие рекомендации

- В случае проблем начните с понимания того, какой компонент действует как TLS клиент и какой действует как TLS сервер.<br> Zabbix сервер, прокси и агенты, в зависимости от взаимодействия между ними, все они могут действовать как в роли TLS сервера, так и в роли TLS клиента.<br> Например, Zabbix сервер, который подключается к агенту для пассивной проверки, действует как TLS клиент. Агент действует в роли TLS сервера.<br> Zabbix агент, запрашивая список активных проверок с прокси, действует как TLS клиент. Прокси действует как TLS сервер.<br> Утилиты zabbix_get и zabbix_sender всегда действуют как TLS клиенты.
- Zabbix использует взаимную аутентификацию.<br> Каждая сторона проверяет своего партнера и может отказать в соединении.<br> Например, Zabbix сервер, подключаясь к агенту, может немедленно закрыть соединение, если сертификат агента ошибочен. И наоборот — Zabbix агент, принимая подключение от сервера, может закрыть соединение, если агент не доверяет серверу.

- Исследуйте файлы журналов на обеих сторонах — и TLS клиента, и TLS сервера.<br> Сторона, которая отвергает соединение, может записать точную причину отказа в соединении. Другая сторона зачастую сообщает о весьма общей ошибке (например, ”Connection closed by peer”, ”connection was non-properly terminated”).
- Иногда неправильно настроенное шифрование приводит к запутанным сообщениям об ошибках, не указывая реальную причину проблем.<br> В подразделах ниже мы попытаемся предоставить (далеко не исчерпывающую) коллекцию сообщений и возможные причины, которые могут помочь в устранении проблем.<br> Пожалуйста, обратите внимание, что разные наборы инструментов криптографии (OpenSSL, GnuTLS) зачастую дают различные сообщения об ошибках в одних и тех же проблемных ситуациях.<br> Иногда сообщения об ошибках зависят даже от конкретной комбинации наборов инструментов криптографии с обеих сторон.

**1. Проблемы с типом подключения или правами**

Сервер настроен на подключение с использованием PSK к агенту, но агент принимает только незашифрованные соединения

В журнале сервера или прокси (с *GnuTLS* 3.3.16)

```
Get value from agent failed: zbx_tls_connect(): gnutls_handshake() failed: \
    -110 The TLS connection was non-properly terminated.
```

В журнале сервера или прокси (с *OpenSSL* 1.0.2c)

```
Get value from agent failed: TCP connection successful, cannot establish TLS to [[127.0.0.1]:10050]: \
    Connection closed by peer. Check allowed connection types and access rights
```

Одна сторона подключается с использованием сертификата, но другая сторона принимает только PSK и наоборот

В любом журнале (с *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.
```

В любом журнале (с *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"
```

Попытка использовать Zabbix sender, скомпилированный с поддержкой TLS, для отсылки данных на Zabbix сервер/прокси, скомпилированный без TLS

В журнале на стороне инициатора соединения:

Linux:

```
...In zbx_tls_init_child()
...OpenSSL library (version OpenSSL 1.1.1  11 Sep 2018) initialized
...
...In zbx_tls_connect(): psk_identity:"PSK test sender"
...End of zbx_tls_connect():FAIL error:'connection closed by peer'
...send value error: TCP successful, cannot establish TLS to [[localhost]:10051]: connection closed by pee
```

Windows:

```
...OpenSSL library (version OpenSSL 1.1.1a  20 Nov 2018) initialized
...
...In zbx_tls_connect(): psk_identity:"PSK test sender"
...zbx_psk_client_cb() requested PSK identity "PSK test sender"
...End of zbx_tls_connect():FAIL error:'SSL_connect() I/O error: [0x00000000] The operation completed succ
...send value error: TCP successful, cannot establish TLS to [[192.168.1.2]:10051]: SSL_connect() I/O erro
```

В журнале на стороне, принимающей соединения:

```
...failed to accept an incoming connection: from 127.0.0.1: support for TLS was not compiled in
```

Одна сторона подключается с использованием PSK, но другая сторона использует LibreSSL или была скомпилирована без поддержки шифрования

LibreSSL не поддерживает PSK.

В журнале на стороне, инициирующей соединение:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() I/O error: [0] Success
```

В журнале на стороне, принимающей соединение:

```
...failed to accept an incoming connection: from 192.168.1.2: support for PSK was not compiled in
```

В веб-интерфейсе Zabbix:

```
Get value from agent failed: TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect()
```

Одна сторона подключается с использованием PSK, но другая сторона использует OpenSSL с выключенной поддержкой PSK

В журнале на стороне, инициирующей соединение:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() set result code to SSL_ERR
```

В журнале на стороне, принимающей соединение:

```
...failed to accept an incoming connection: from 192.168.1.2: TLS handshake set result code to 1: file ssl
```

**2. Проблемы с сертификатами**

OpenSSL используется с CRL и по некоторым CA в цепочке сертификатов их CRL не включены в `TLSCRLFile`

В журнале TLS сервера в случае *OpenSSL* узла:

```
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"
```

В журнале TLS сервера в случае *GnuTLS* узла:

```
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 истёк или истекает в процессе работы сервера

*OpenSSL*, в журнале сервера:

- до истечения срока действия:

```
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"
```

- после истечения срока действия:

```
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"
```

Дело в том, что при наличии действительного CRL аннулированный сертификат записывается как «certificate revoked». При истекшем CRL сообщение об ошибке меняется на «certificate expired», которое может ввести в заблуждение.

*GnuTLS*, в журнале сервера:

- до и после истечения срока действия одинаково:

```
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.
```

Самоподписанный сертификат, неизвестный CA

*OpenSSL*, в журнале:

```
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"'
```

Это наблюдалось, когда сертификат сервера по ошибке имел одну и ту же строку Issuer и Subject, хотя он был подписан CA. Эмитент (Issuer) и Субъект (Subject) одинаковы в сертификате CA верхнего уровня, но не могут быть одинаковыми в сертификате сервера. (То же самое относится к сертификатам прокси и агента.)

Чтобы проверить, содержит ли сертификат те же записи Эмитента и Субъекта, запустите:

```
openssl x509 -in <  _      .crt> -noout -text
```

Допустимо, чтобы одинаковые значения Эмитента и Субъекта имел корневой сертификат (верхнего уровня).

**3. Проблемы с PSK**

PSK содержит нечётное количество hex-цифр

Прокси или агент не запускаются, сообщение в журнале прокси или агента:

```
invalid PSK in file "/home/zabbix/zabbix_proxy.psk"
```

В GnuTLS передана строка идентификации PSK длиннее 128 байт

В журнале на стороне TLS клиента:

```
gnutls_handshake() failed: -110 The TLS connection was non-properly terminated.
```

В журнале на стороне TLS сервера:

```
gnutls_handshake() failed: -90 The SRP username supplied is illegal.
```

С OpenSSL 1.1.1 используется слишком длинное значение PSK

В журнале на стороне, инициирующей соединение:

```
...OpenSSL library (version OpenSSL 1.1.1  11 Sep 2018) initialized
...
...In zbx_tls_connect(): psk_identity:"PSK 1"
...zbx_psk_client_cb() requested PSK identity "PSK 1"
...End of zbx_tls_connect():FAIL error:'SSL_connect() set result code to SSL_ERROR_SSL: file ssl\statem\ex
```

В журнале на строне, принимающей соединение:

```
...Message from 123.123.123.123 is missing header. Message ignored.
```

Эта проблема возникает, когда OpenSSL обновляется с версии 1.0.x или 1.1.0 на 1.1.1 и если значение PSK длиннее чем 512-бит (64-байт PSK, введённый как 128 16-ричных цифр).

Смотрите также: Ограничения размера значений.

## 18. Веб-интерфейс

**Обзор** Для лёгкого доступа к Zabbix из любого места и с любой платформы, предоставляется веб-интерфейс.

> **Note:**
> Если Вы используете несколько экземпляров веб-интерфейса, убедитесь, что локали и необходимые библиотеки (LDAP, SAML и т.д.) установлены и настроены одинаково для каждого из веб-интерфейсов.

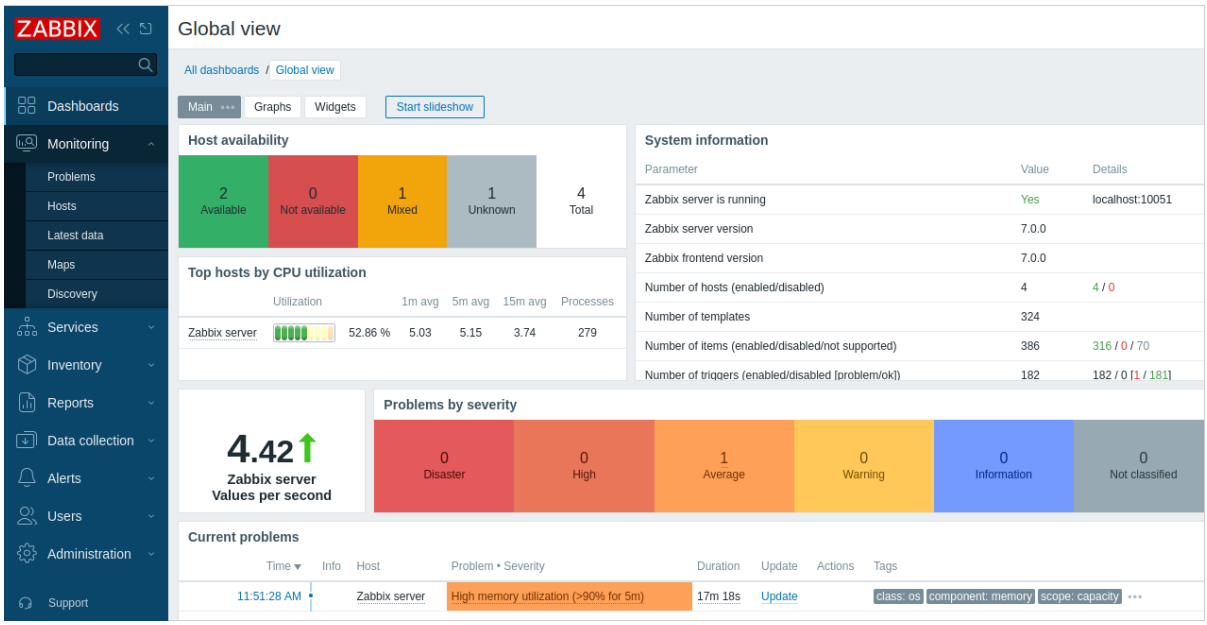
**Помощь по веб-интерфейсу** Ссылка на помощь предоставляется в формах веб-интерфейса Zabbix с прямыми ссылками на соответствующие части документации.

### 1. Меню

Обзор

Вертикальное меню на боковой панели обеспечивает доступ к различным разделам веб-интерфейса Zabbix.

В теме по умолчанию меню тёмно-синего цвета.

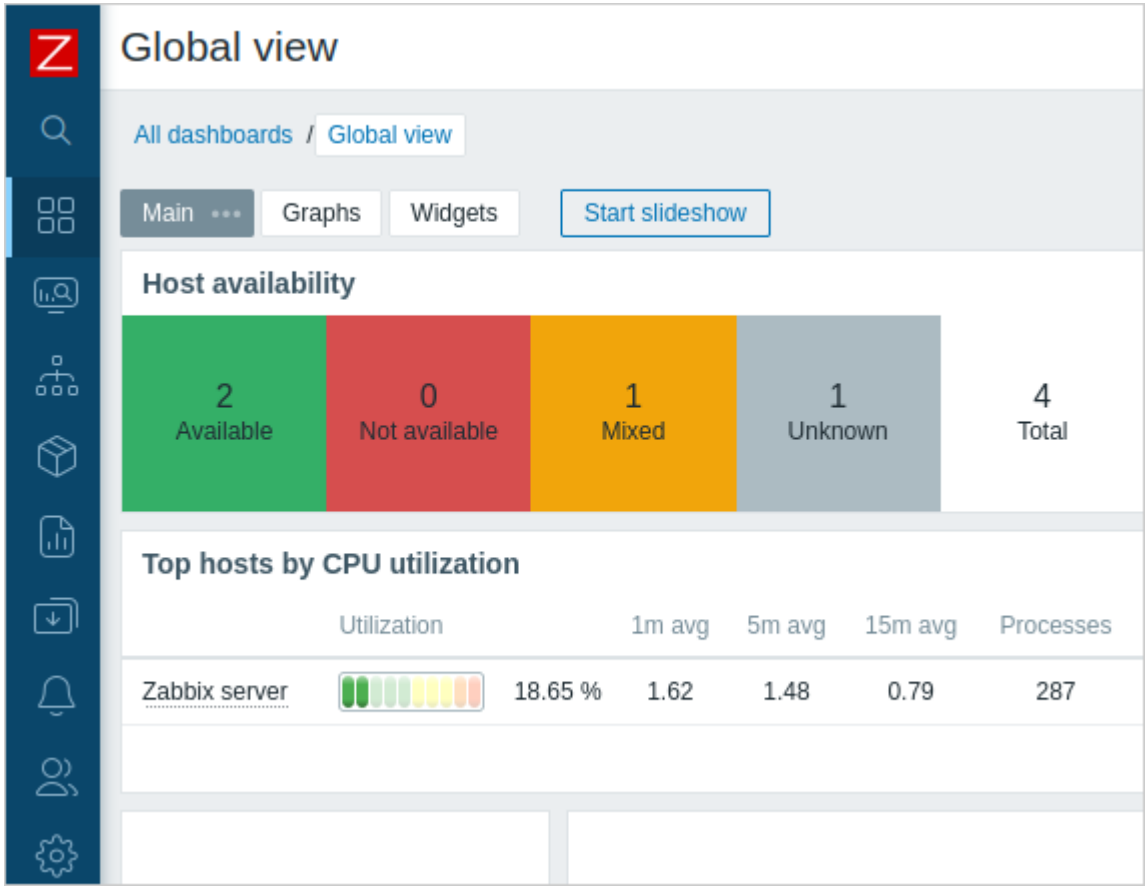



Работа с меню

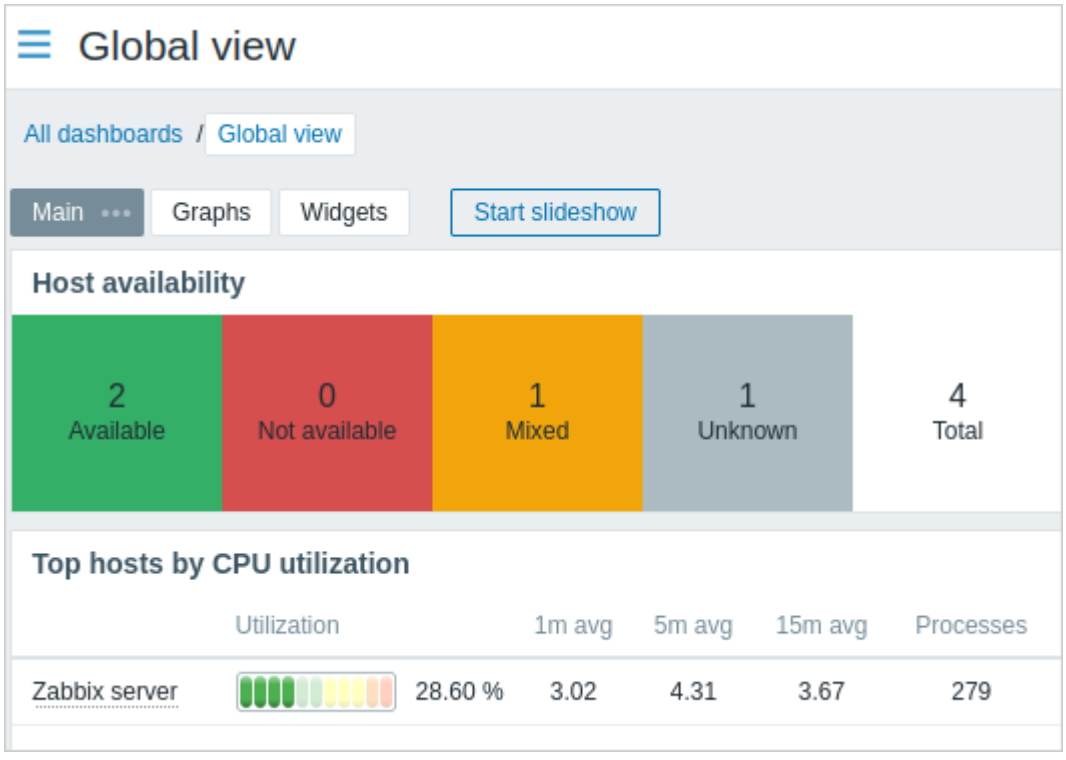
Поле глобального поиска находится под логотипом Zabbix.

Меню можно свернуть или полностью скрыть:

- Чтобы свернуть меню, нажмите рядом с логотипом Zabbix. В свернутом меню видны только значки.




- Чтобы скрыть меню, нажмите рядом с логотипом Zabbix. В скрытом меню все элементы скрыты.




Свернутое меню

Когда меню свернуто до одних значков, полное меню снова появляется, как только курсор мыши оказывается над ним. Обратите внимание, что оно появляется поверх содержимого страницы; чтобы сдвинуть содержимое страницы вправо, нужно нажать кнопку разворачивания. Если курсор мыши снова окажется вне полного меню, меню снова свернется через две секунды.

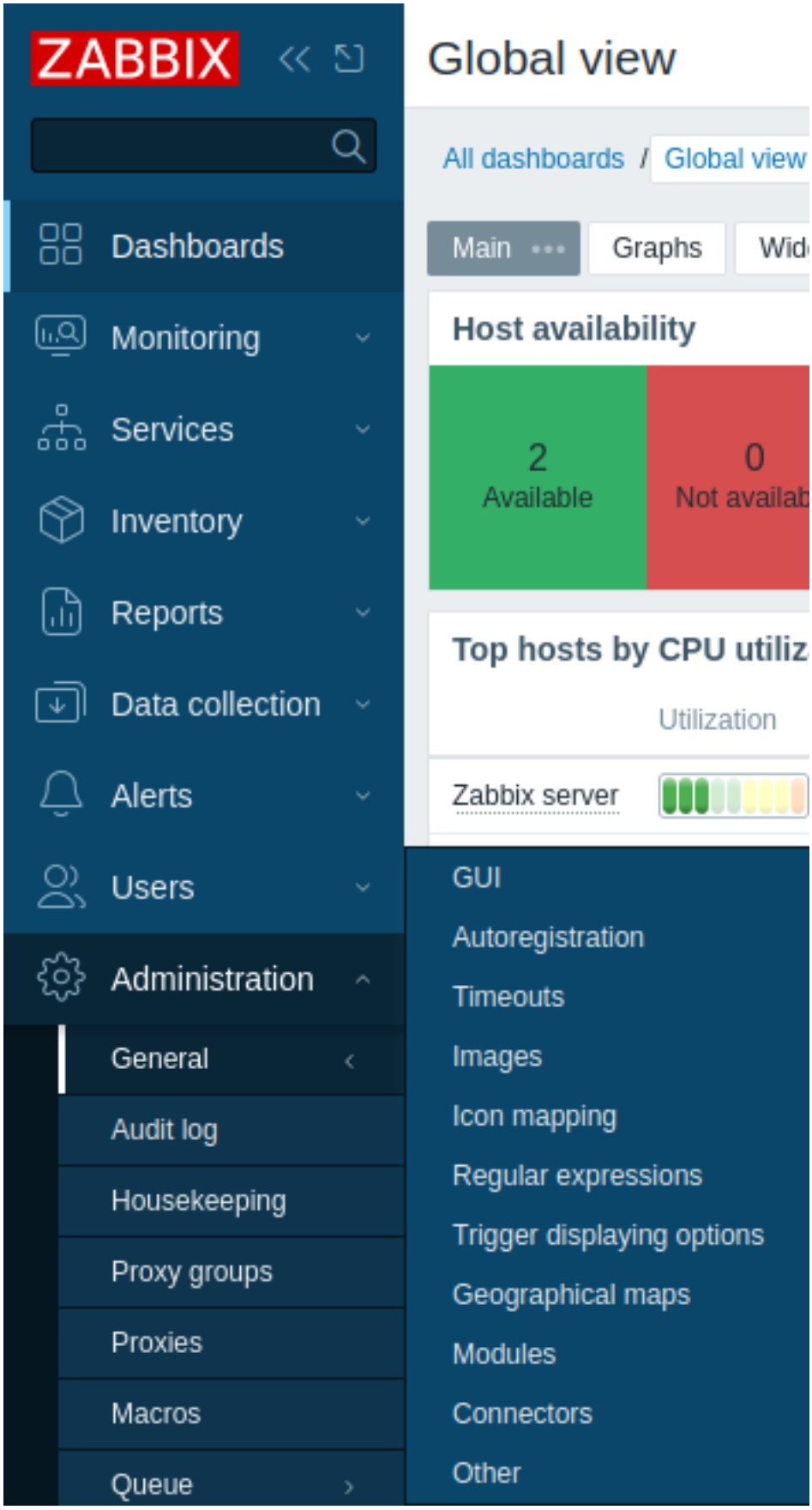
Вы также можете полностью развернуть свернутое меню, нажав клавишу Tab. Повторное нажатие клавиши Tab позволит перейти к фокусу на следующем элементе меню.

Скрытое меню

Даже если меню полностью скрыто, полный список меню можно открыть всего одним щелчком мыши, нажав на значок «бургер» . Обратите внимание, что оно появляется поверх содержимого страницы; чтобы сдвинуть содержимое страницы вправо, нужно снова показать меню, нажав кнопку отображения боковой панели.

Уровни меню

В меню может быть до трёх уровней.




Контекстные меню

Помимо главного меню, Zabbix предоставляет контекстные меню для узел сети, элемент данных и событие для быстрого доступа к часто используемым объектам, таким как последние значения, простой график, форма настройки, связанные скрипты или внешние ссылки.

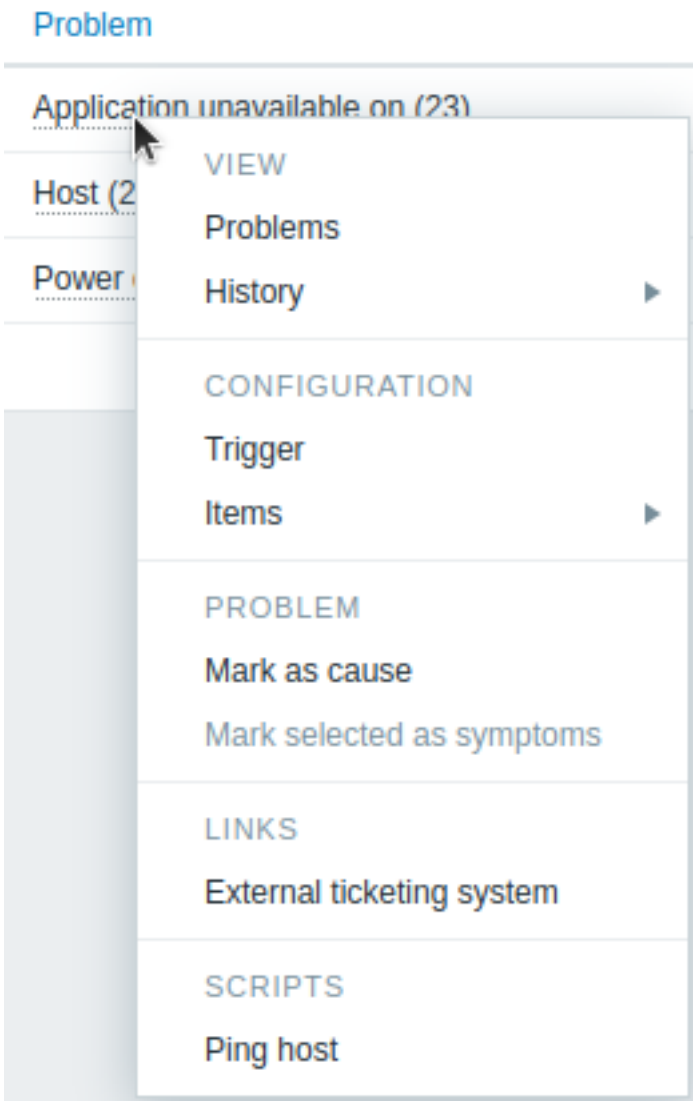
Контекстные меню доступны при щелчке по имени узла сети, элемента данных или проблемы/триггера в поддерживаемых местах.

### 1. Меню событий

Обзор

Меню события содержит ярлыки для действий или разделов веб-интерфейса, которые часто требуются для события.

Меню события можно открыть, щелкнув по названию события.




Содержимое

Контекстное меню событий имеет шесть разделов: *Просмотр*, *Действия*, *Настройка*, *Проблема*, *Ссылки* и *Скрипты*. Для объектов, которые не настроены, ссылки недоступны и отображаются серым цветом. Разделы *Скрипты* и *Ссылки* отображаются, если их объекты настроены.

Раздел *Просмотр (View)* содержит ссылки:

- **Проблемы (Problems)** — окрывает список нерешённых проблем нижележащего триггера;
- **История (History)** — ведёт на *Последние данные (Latest data)* (график либо история элемента данных) по нижележащим элементам данных. Если триггер использует несколько элементов данных, будут доступны ссылки по каждому из них.

Раздел *Действия (Actions)* доступен только в виджете *Обзор триггеров (Trigger overview)*. Он содержит ссылки:

- **Обновить проблему (Update problem)** — открывает экран обновления проблемы.

Раздел *Настройка (Configuration)* содержит ссылки, чтобы настроить:

- **Триггер (Trigger)**, который вызвал проблему;
- **Элементы данных (Items)**, используемые в триггерном выражении.

**Note:**
Обратите внимание, что раздел настроек доступен только пользователям с ролью Администратор и Супер-администратор.

Раздел *Проблемы (Problem)* содержит опции, чтобы:

- **Отметить причиной (Mark as cause)** — пометить эту проблему как причину;
- **Отметить выбранное симптомами (Mark selected as symptoms)** — пометить выбранные проблемы как симптомы данной проблемы.

Раздел *Ссылки (Links)* содержит ссылки на:

- доступ к настроенному URL триггера;
- доступ к пользовательским ссылкам, настроенным в разделе глобальных скриптов (с областью «Действие вручную над событиями» и типом «URL»);
- доступ к настроенным внешним тикетам для проблем (смотрите опцию *Добавить запись в меню события (Include event menu entry)* при настройке вебхуков).

Раздел *Скрипты (Scripts)* содержит ссылки на внешние глобальные скрипты (с областью *Действие вручную над событиями (Manual event action)*). Эта возможность может быть удобна для запуска скриптов, используемых для управления тикетами проблем во внешних системах.

Поддерживаемые места

Контекстное меню события доступно при нажатии на имя проблемы или события в различных разделах веб-интерфейса, например:

- Панели виджеты, такие как виджет *Problems*, виджет *Trigger overview* и т. д.
- Мониторинг → Проблемы
- Мониторинг → Проблемы → Детали события
- Отчеты → Топ 100 триггеров (глобальные скрипты и доступ к внешнему тикету в этом месте не поддерживаются)

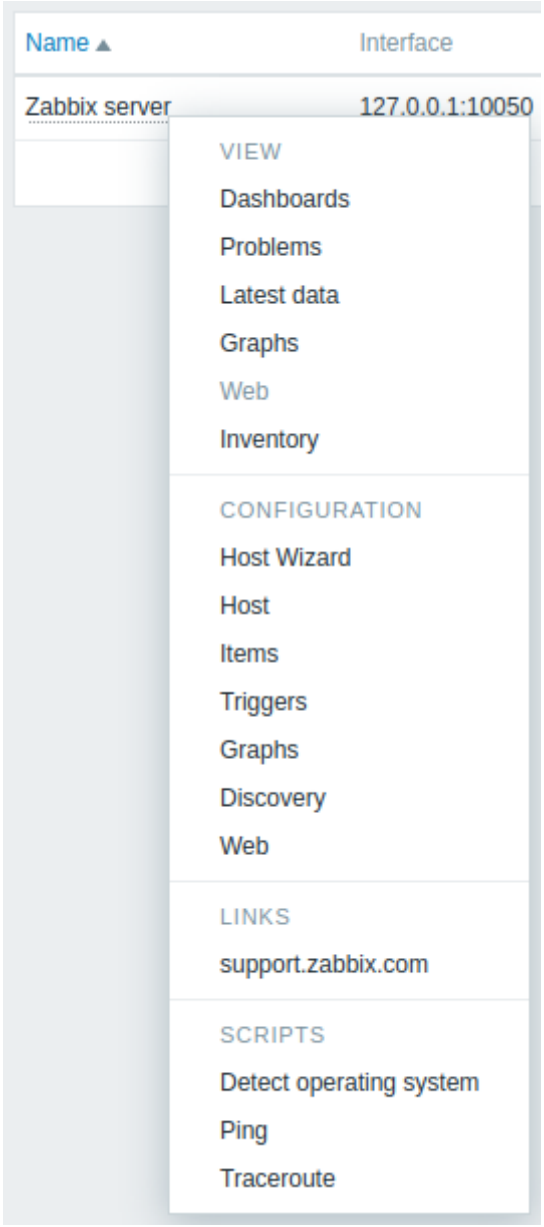
**2. Меню узлов сети**

Обзор

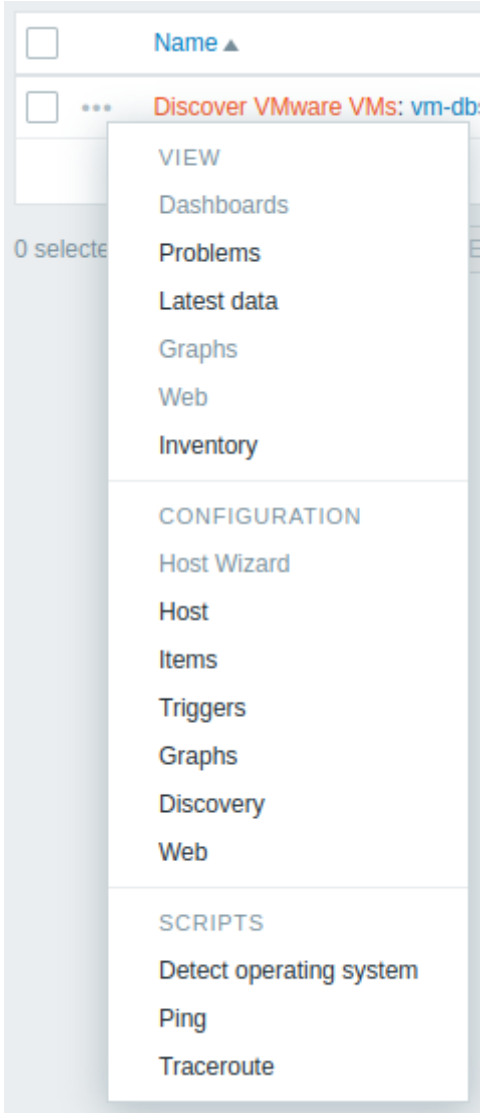
Меню узла сети содержит ярлыки для действий или разделов веб-интерфейса, которые часто требуются для узла сети.

Меню узла сети можно открыть, щелкнув по имени узла сети или значку с тремя точками, в зависимости от раздела веб-интерфейса, например:

*Monitoring* → *Latest data*




*Data collection* → *Hosts*




Содержимое

Контекстное меню узла сети имеет четыре раздела: *View*, *Configuration*, *Links* и *Scripts*. Для сущностей, которые не настроены, ссылки отключены и отображаются серым цветом. Разделы *Scripts* и *Links* отображаются, если соответствующие сущности настроены.

Раздел *View* содержит ссылки на:

- **Dashboards** - открывает виджеты и графики.
- **Problems** - открывает раздел *Problems* со списком нерешенных проблем, связанных с соответствующим триггером.
- **Latest data** - открывает раздел *Latest data* со списком всех последних данных текущего узла сети.
- **Graphs** - открывает простые графики текущего узла сети.
- **Web** - открывает ссылку на настроенные веб-сценарии.
- **Inventory** - открывает ссылку на инвентаризацию текущего узла сети.

Раздел *Configuration* содержит ссылки на:

- **Host Wizard** - открывает Host Wizard для текущего узла сети (отключено для обнаруженных узлов сети).
- **Host** - форма настройки текущего узла сети.
- **Items** - список элементов данных текущего узла сети.
- **Triggers** - список триггеров текущего узла сети.
- **Graphs** - простые графики текущего узла сети.
- **Discovery** - список правил низкоуровневого обнаружения текущего узла сети.

- **Web** - список веб-сценариев текущего узла сети.

> **Note:**
> Обратите внимание, что раздел настройки доступен только пользователям с ролями Admin и Super admin.

Раздел *Links* содержит ссылки для:

- доступа к настроенному trigger URL.
- доступа к пользовательским ссылкам, настроенным в Global scripts (с областью действия *Manual host action* и типом 'URL').

Раздел *Scripts* позволяет выполнять global scripts, настроенные для текущего узла сети. Чтобы эти скрипты были доступны в меню узла сети, для них должна быть задана область действия *Manual host action*.

Поддерживаемые расположения

Меню узла сети доступно при нажатии на имя узла сети в различных разделах веб-интерфейса, например:

- Панели мониторинга виджеты, такие как Problems, Top items, Trigger overview и т. д.
- Monitoring → Problems
- Monitoring → Problems → Event details
- Monitoring → Hosts
- Monitoring → Hosts → Web Monitoring
- Monitoring → Latest data
- Monitoring → Maps
- Inventory → Hosts
- Reports → Top 100 triggers

Меню узла сети также доступно при нажатии на значок с тремя точками в разделе Data collection → Hosts.

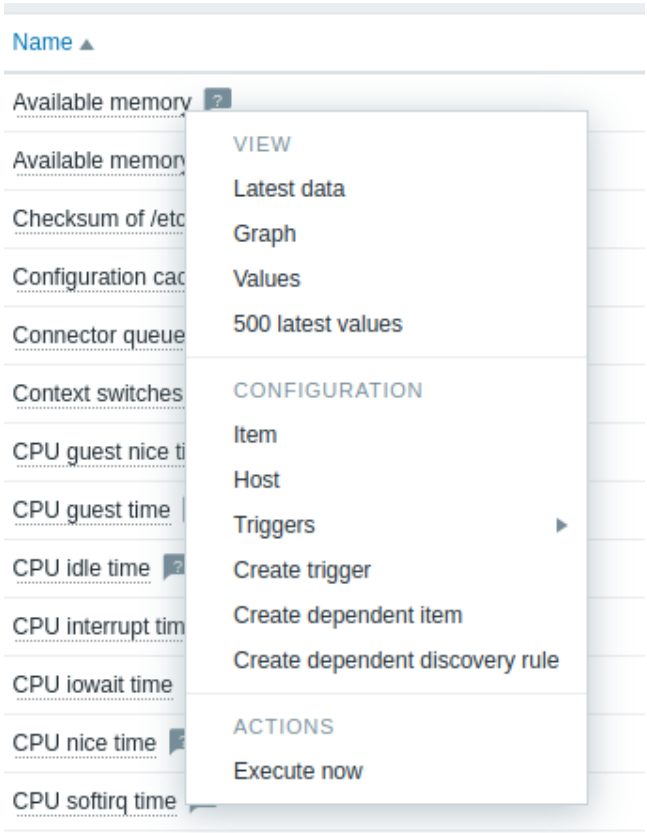
### 3. Меню элементов данных

Обзор

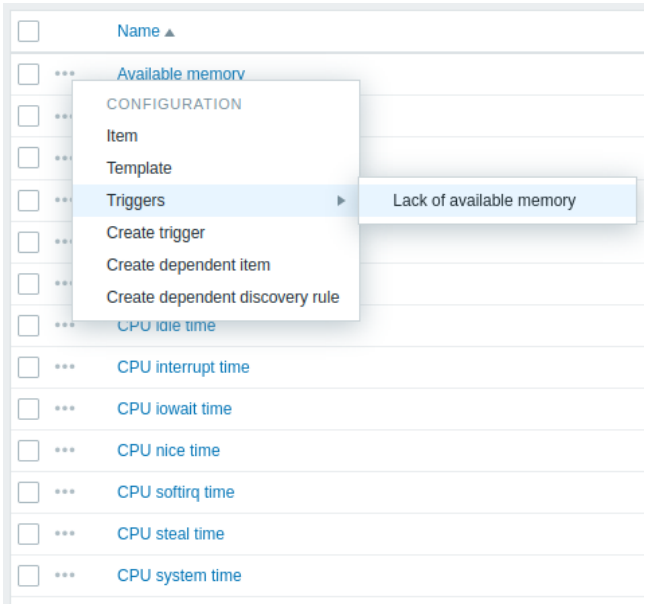
Меню элемента данных содержит ярлыки для действий или разделов веб-интерфейса, которые часто требуются для элемента данных.

Меню элемента данных можно открыть, щелкнув по имени элемента данных или по значку с тремя точками, в зависимости от раздела веб-интерфейса, например:

*Monitoring* → *Latest data*




*Data Collection* → *Templates* → *Items*




Содержание

Меню элемента данных имеет три раздела: *View*, *Configuration* и *Actions*.

Раздел *View* содержит следующие параметры:

- **Latest data** - открывает раздел *Latest data*, отфильтрованный по текущему узлу сети и элементу данных;
- **Graph** - открывает простой график текущего элемента данных;
- **Values** - открывает список всех значений, полученных для текущего элемента данных за последние 60 минут;
- **500 latest values** - открывает список 500 последних значений для текущего элемента данных.

Раздел *Configuration* (доступен только пользователям типов *Admin* и *Super admin*) содержит следующие параметры:

- **Item** - открывает форму настройки элемента данных текущего элемента данных;
- **Template** - открывает форму настройки шаблона, к которому относится элемент данных (доступно только при переходе к меню элемента данных из *Data collection* → *Templates* → *Items*);
- **Host** - открывает форму настройки узла сети, к которому относится элемент данных;
- **Triggers** - при наведении курсора открывает список триггеров элемента данных, если они есть; щелчок по триггеру открывает его форму настройки триггера;
- **Create trigger** - открывает форму настройки триггера для создания триггера для этого элемента данных;
- **Create dependent item** - открывает форму настройки элемента данных для создания зависимого элемента данных, где текущий элемент данных является мастер-элементом;
- **Create dependent discovery rule** - открывает форму настройки правила обнаружения для создания зависимого правила обнаружения, где текущий элемент данных является мастер-элементом.

Раздел *Actions* содержит следующий параметр:

- **Execute now** - немедленно выполняет проверку для получения нового значения элемента данных.

Поддерживаемые расположения

Меню элемента данных доступно при щелчке по имени элемента данных в различных разделах веб-интерфейса, например:

- Мониторинг → Последние данные
- Сбор данных → Узлы сети → Элементы данных
- Сбор данных → Узлы сети → Правила обнаружения → Прототипы элементов данных

Меню элемента данных доступно при щелчке по значению элемента данных в виджете панели Топ элементов.

### 2. Разделы веб-интерфейса

**Структура меню** Меню веб-интерфейса Zabbix имеет следующую структуру:

- Панели (Dashboards)
- Мониторинг (Monitoring)
  - Проблемы (Problems)
  - Узлы сети (Hosts)
  - Последние данные (Latest data)
  - Карты сетей (Maps)
  - Обнаружение (Discovery)
- Услуги (Services)
  - Услуги (Services)
  - SLA
  - Отчёт SLA (SLA report)
- Инвентаризация (Inventory)
  - Обзор (Overview)
  - Узлы сети (Hosts)
- Отчёты (Reports)
  - Информация о системе (System information)
  - Регулярные отчёты (Scheduled reports)
  - Отчёт о доступности (Availability report)
  - Топ 100 триггеров (Top 100 triggers)
  - Журнал аудита (Audit log)
  - Журнал действий (Action log)
  - Оповещения (Notifications)
- Сбор данных (Data collection)
  - Группы шаблонов (Template groups)
  - Группы узлов сети (Host groups)
  - Шаблоны (Templates)
  - Узлы сети (Hosts)
  - Обслуживание (Maintenance)
  - Корреляция событий (Event correlation)
  - Обнаружение (Discovery)
- Оповещения (Alerts)
  - Действия (Actions)
    - Действия триггеров (Trigger actions)
    - Действия услуг (Service actions)
    - Действия обнаружения (Discovery actions)
    - Действия авторегистрации (Autoregistration actions)
    - Внутренние действия (Internal actions)
  - Способы оповещений (Media types)
  - Скрипты (Scripts)
- Пользователи (Users)
  - Группы пользователей (User groups)
  - Роли пользователей (User roles)
  - Пользователи (Users)
  - API токены (API tokens)
  - Аутентификация (Authentication)
- Администрирование (Administration)
  - Общие (General)
    - Веб-интерфейс (GUI)
    - Авторегистрация (Autoregistration)
    - Время ожидания (Timeouts)
    - Изображения (Images)
    - Соответствия иконок (Icon mapping)
    - Регулярные выражения (Regular expressions)
    - Опции отображения триггеров (Trigger displaying options)
    - Географические карты (Geographical maps)

    - Модули (Modules)
    - Коннекторы (Connectors)
    - Другое (Other)
- Журнал аудита (Audit log)
- Очистка истории (Housekeeping)
- Группы прокси (Proxy groups)
- Прокси (Proxies)
- Макросы (Macros)
- Очередь (Queue)
    - Обзор очереди (Queue overview)
    - Обзор очереди по прокси (Queue overview by proxy)
    - Детали очереди (Queue details)

### 1. Панели

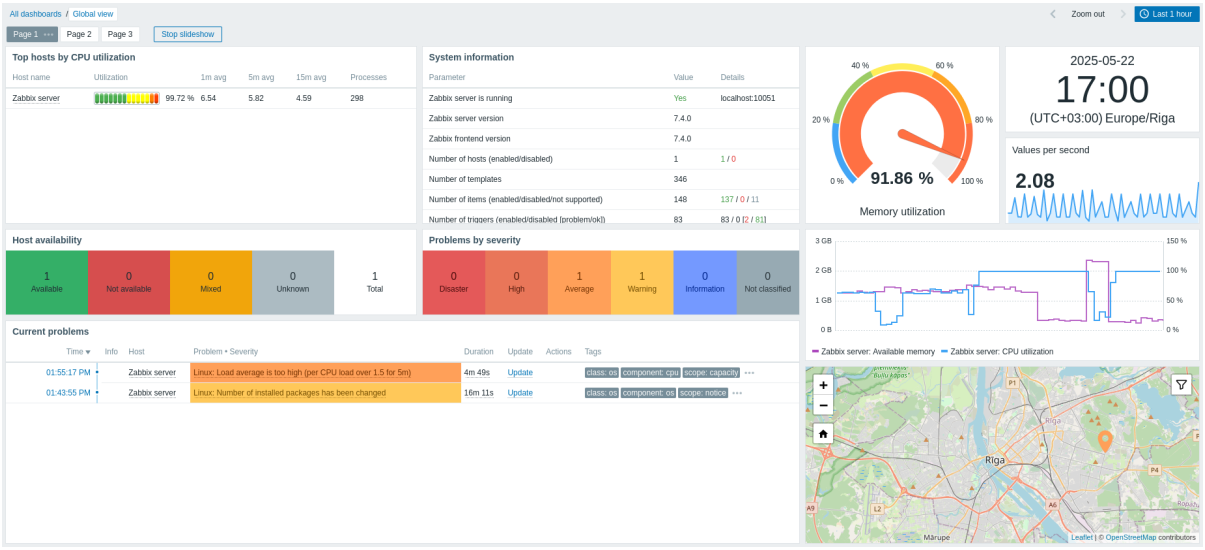
Обзор

Раздел *Панели* предназначен для показа сводки всей важной информации на **панели**.

Хотя одновременно может отображаться только одна панель, можно настроить несколько панелей. Каждая панель может содержать одну или несколько страниц, которые могут чередоваться в режиме слайд-шоу.

Страница панели состоит из виджетов, и каждый виджет предназначен для отображения информации определённого вида и источника, таких как обзор, карта, график, часы и т. д.

Доступ к узлам сети в виджетах зависит от прав доступа к этим узлам сети.




Страницы и виджеты добавляются на панель и редактируются в режиме редактирования панели. Страницы могут просматриваться и чередоваться в режиме просмотра панели.

Период времени, отображаемый в виджетах графиков, управляется при помощи селектора *Период времени*, расположенного над виджетами. Подпись к селектору *периода времени*, расположенная справа, отображает текущий выбранный период. Нажатие на подписи к вкладке позволяет раскрыть или свернуть селектор *периода времени*.

Размер панели

Минимальная ширина панели составляет 1200 пикселей. Панель не будет ужиматься менее этой ширины; если окно браузера меньше, то вместо этого будет отображаться горизонтальная полоса прокрутки.

Максимальная ширина панели равна ширине окна браузера. Виджеты панели растягиваются по горизонтали, чтобы заполнить окно. В то же время, виджет панели не может растянуться по горизонтали за пределы окна.

Панель состоит из 72 столбцов по горизонтали всегда одинаковой ширины, которые растягиваются/сжимаются динамически (но не менее чем до 1200 пикселей в сумме).

По вертикали панель может содержать максимум 64 строки; каждая строка имеет фиксированную высоту 70 пикселей.

Таким образом, виджет может быть до 72 столбцов в ширину и 64 строк в высоту.

Просмотр дашбордов

Чтобы просмотреть все настроенные дашборды, нажмите *Все дашборды* сразу под заголовком раздела.

Дашборды отображаются с тегом sharing:

- *My* - указывает на дашборд, принадлежащий текущему пользователю
- *Shared* - указывает на общедоступный дашборд или частный дашборд, которым поделились с любым пользователем или группой пользователей

Обратите внимание, что теги общего доступа отображаются только для дашбордов, принадлежащих текущему пользователю. Дашборды, принадлежащие другим пользователям, перечисляются без тегов общего доступа.

Фильтр, расположенный справа над списком, позволяет фильтровать дашборды по имени и по тем, которые созданы текущим пользователем.

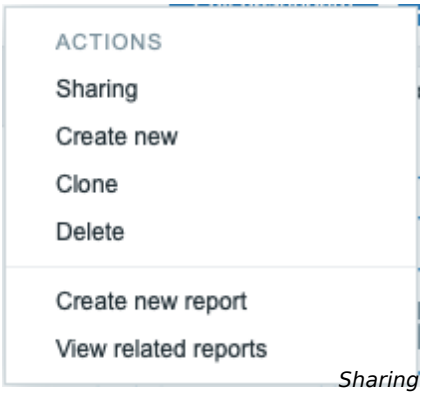
Чтобы удалить один или несколько дашбордов, отметьте флажками соответствующие дашборды и нажмите *Удалить* под списком.

Просмотр дашборда

Чтобы просмотреть отдельный дашборд, щелкните его имя в списке дашбордов.

При **просмотре** дашборда доступны следующие параметры:

| | |
|---|---|
| Edit dashboard | Переключиться в режим **редактирования** дашборда.<br>Режим редактирования также открывается при создании нового дашборда и при нажатии кнопки редактирования у виджета. |
| | Открыть меню действий (см. описания действий ниже). |




*Sharing* - изменить настройки общего доступа для дашборда.
*Create new* - создать новый дашборд.
*Clone* - создать новый дашборд, скопировав свойства существующего. Сначала вам будет предложено ввести параметры дашборда. Затем новый дашборд откроется в режиме редактирования со всеми виджетами исходного дашборда.
*Delete* - удалить дашборд.
*Create new report* - открыть всплывающее окно с формой настройки отчета. Недоступно, если у пользователя нет прав на управление запланированными отчетами.
*View related reports* - открыть всплывающее окно со списком существующих отчетов, основанных на текущем дашборде. Недоступно, если связанных отчетов нет или у пользователя нет прав на просмотр запланированных отчетов.

Показывать только содержимое страницы (режим kiosk).
К режиму kiosk также можно перейти с помощью следующих параметров URL: `/zabbix.php?action=dashboard.view&kiosk=1`.
Чтобы выйти в обычный режим: `/zabbix.php?action=dashboard.view&kiosk=0`.

Общий доступ

Панели мониторинга могут быть общедоступными или приватными.

Общедоступные панели мониторинга видны всем пользователям. Приватные панели мониторинга видны только их владельцу и пользователям/группам пользователей, добавленным в настройках общего доступа.

Чтобы изменить статус общего доступа панели мониторинга, нажмите *Общий доступ* в меню действий при просмотре отдельной панели мониторинга:

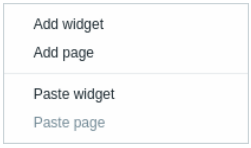
| Parameter | Description |
|---|---|
| *Type* | Выберите тип панели мониторинга:<br>**Private** - панель мониторинга видна только выбранным группам пользователей и пользователям.<br>**Public** - панель мониторинга видна всем. |
| *List of user group shares* | Выберите группы пользователей, которым доступна панель мониторинга.<br>Можно разрешить доступ только для чтения или чтения и записи. |
| *List of user shares* | Выберите пользователей, которым доступна панель мониторинга.<br>Можно разрешить доступ только для чтения или чтения и записи. |

См. Разрешения на панели мониторинга, чтобы узнать, как параметры общего доступа ограничивают действия, доступные пользователям.

Редактирование панели мониторинга

При **редактировании** панели мониторинга доступны следующие параметры:

Редактировать общие параметры панели мониторинга.
Добавить новый виджет.
Нажатие кнопки со стрелкой откроет меню действий (см. описания действий ниже).

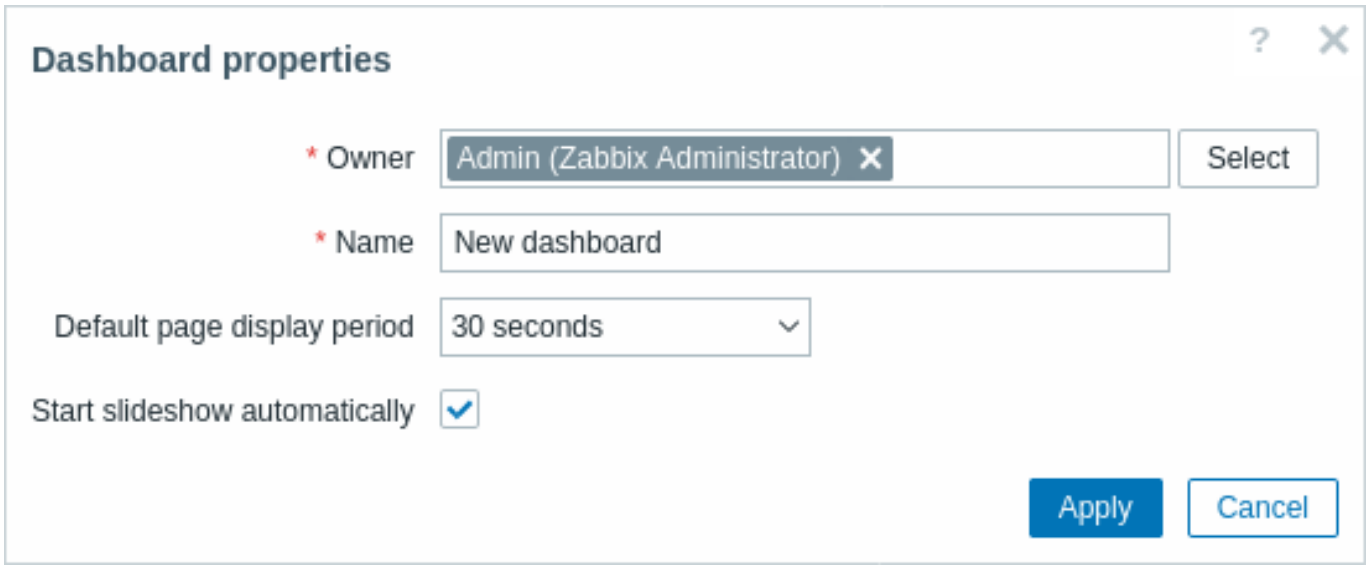
| | |
|---|---|
|  | *Add widget* - добавить новый виджет.<br>*Add page* - добавить новую страницу.<br>*Paste widget* - вставить скопированный виджет. Этот параметр недоступен, если виджет не был скопирован. Одновременно можно скопировать только один объект (виджет или страницу).<br>*Paste page* - вставить скопированную страницу. Этот параметр недоступен, если страница не была скопирована. |
| | Сохранить изменения панели мониторинга. |
|  | Отменить изменения панели мониторинга. |

Создание панели мониторинга

Создать новую панель мониторинга можно двумя способами:

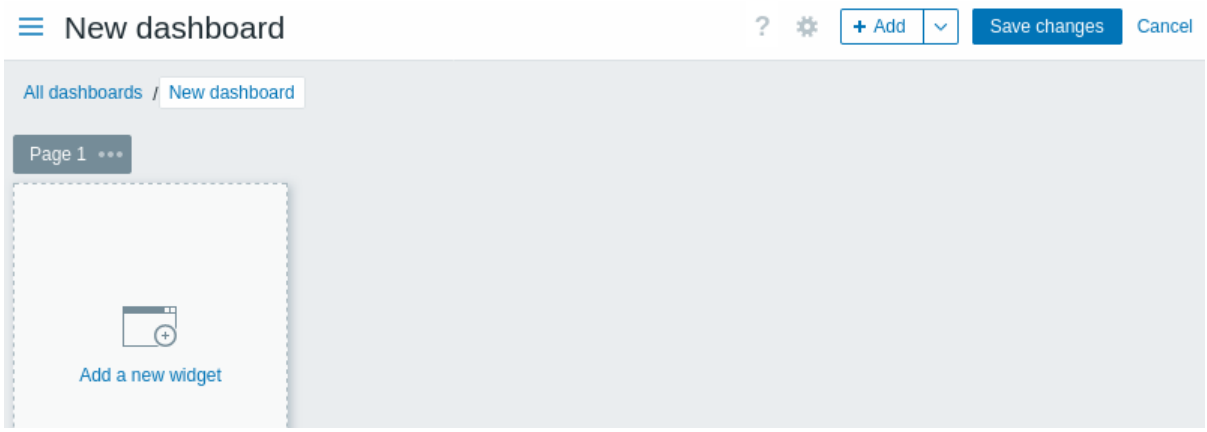
- Нажмите *Create dashboard* при просмотре всех панелей мониторинга
- Выберите *Create new* в меню действий при просмотре отдельной панели мониторинга

Сначала вам будет предложено ввести общие параметры панели мониторинга:




| Parameter | Description |
|---|---|
| *Owner* | Выберите системного пользователя, который будет владельцем панели мониторинга. |
| *Name* | Введите имя панели мониторинга. |
| *Default page display period* | Выберите период, в течение которого страница панели мониторинга отображается перед переходом к следующей странице в слайд-шоу. |
| *Start slideshow automatically* | Установите этот флажок, чтобы автоматически запускать слайд-шоу, если существует более одной страницы панели мониторинга. |

После нажатия *Apply* откроется пустая панель мониторинга:



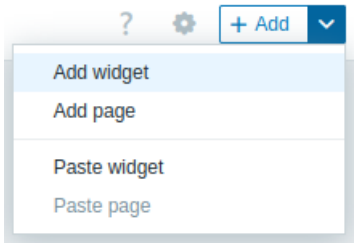

Чтобы заполнить панель мониторинга, вы можете добавить виджеты и страницы.

Нажмите кнопку *Save changes*, чтобы сохранить панель мониторинга. Если вы нажмете *Cancel*, панель мониторинга не будет создана.

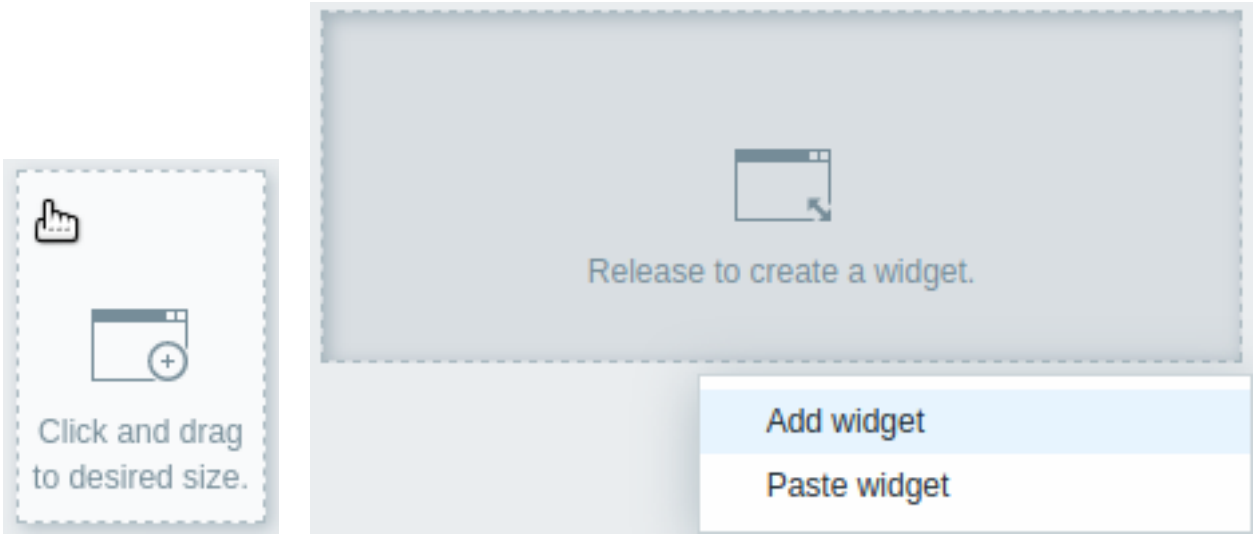
Добавление виджетов

Вы можете добавить на панель мониторинга различные виджеты (журнал действий, часы, статус обнаружения и т. д.). Виджеты можно добавлять двумя способами:

- Нажмите кнопку *Add* или стрелку рядом с ней, затем выберите *Add widget* в меню действий. После настройки виджет будет добавлен в размере по умолчанию и размещен после уже существующих виджетов.

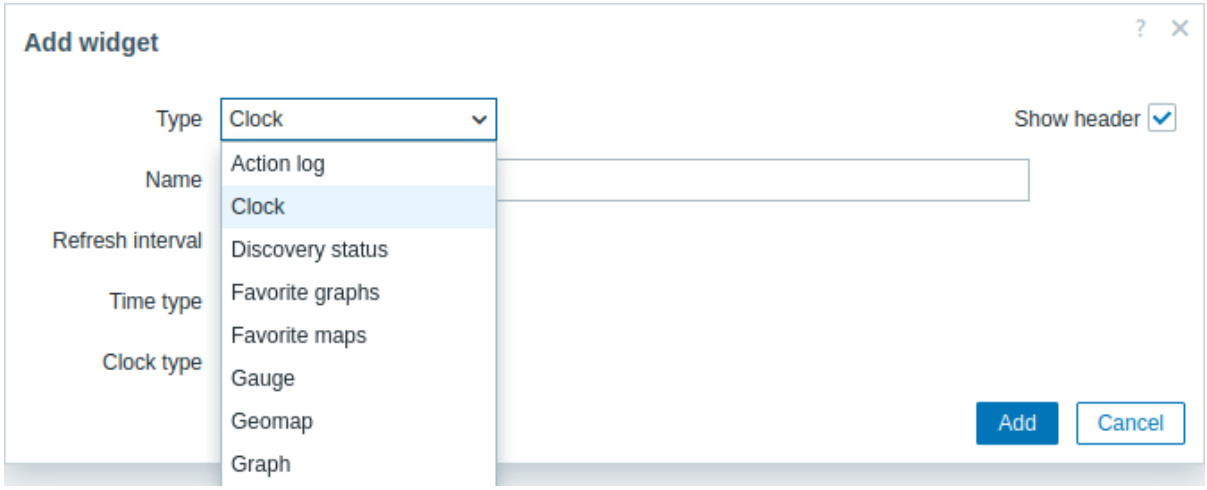



- Переместите курсор мыши на пустое место на панели мониторинга. Появится заполнитель, затем щелкните его, чтобы открыть форму настройки виджета. После настройки виджет будет добавлен в размере по умолчанию или изменен так, чтобы занять доступное пространство. Кроме того, вы можете перетащить заполнитель, чтобы задать размер виджета, и отпустить его, чтобы открыть форму настройки виджета (если виджет скопирован в буфер обмена, сначала вам будет предложено выбрать между *Add widget* и *Paste widget*).




В форме настройки виджета:

1. Выберите тип виджета.
2. Настройте параметры виджета.
3. Нажмите *Add*, чтобы добавить виджет на панель мониторинга.



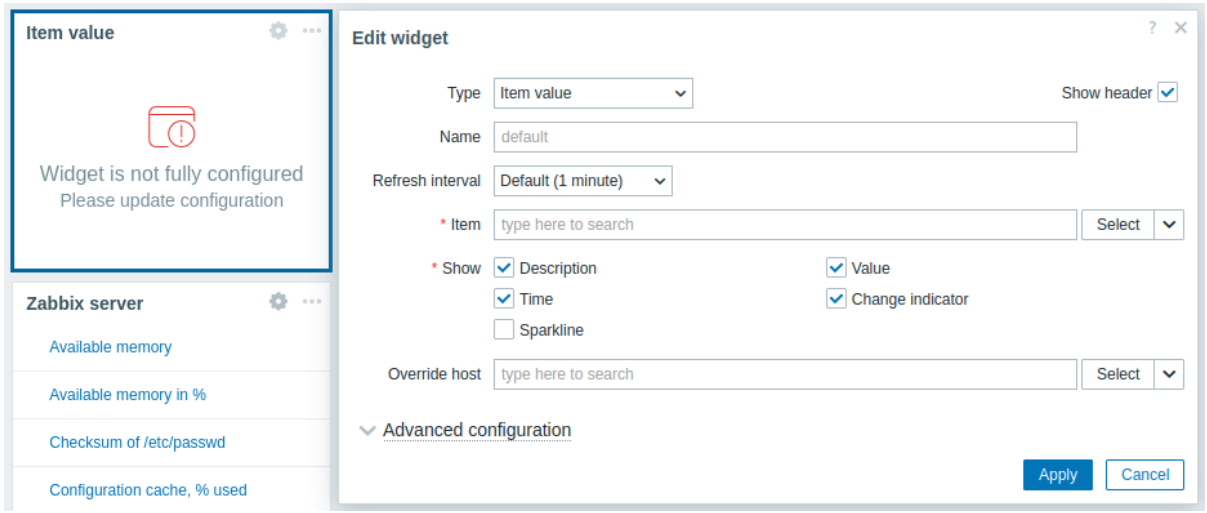

Редактирование виджетов

Виджеты можно изменять по размеру и перемещать в режиме редактирования панели мониторинга, перетаскивая их строку заголовка. Вы также можете перемещать форму настройки виджета, перетаскивая ее строку заголовка.

У каждого виджета в правом верхнем углу есть элементы управления:

- - редактировать виджет
- - открыть меню виджета

Во время редактирования виджет подсвечивается, а изменения отображаются в реальном времени. Если обязательные параметры виджета не настроены, виджет переходит в состояние *Widget is not fully configured*.




После редактирования нажмите *Apply* в виджете, затем *Save changes* в правом верхнем углу панели мониторинга, чтобы применить изменения.

Копирование и вставка виджетов

Для быстрого создания новых виджетов с теми же настройками можно использовать технологию copy/paste (копирование и вставка) виджетов панели. Виджеты можно копировать как в пределах той же панели, так и между панелями, открытыми в различных вкладках баузера.

Чтобы скопировать виджет, используйте меню виджетов. Чтобы вставить скопированный виджет (доступно в режиме редактирования панели):

- нажмите на стрелку рядом с кнопкой *Добавить (Add)* и выберите опцию *Вставить виджет (Paste widget)*.
- альтернативный вариант — щёлкните кнопкой мыши на пустую область на панели и выберите *Вставить виджет (Paste widget)*.

Скопированный виджет может быть использован для вставки поверх существующего виджета, используя опцию *Вставить (Paste)* в меню виджета.

Создание слайд-шоу

Слайд-шоу будет запускаться автоматически, если на панели данных содержится две или более страниц (см. Добавление страниц) и если выполняется одно из следующих условий:

- В свойствах панели данных отмечена опция *Автоматически запускать слайд-шоу*
- URL панели данных содержит параметр `slideshow=1`

Страницы переключаются с интервалами, заданными в свойствах панели данных и отдельных страниц. Нажмите:

- *Остановить слайд-шоу* - чтобы остановить слайд-шоу
- *Запустить слайд-шоу* - чтобы запустить слайд-шоу

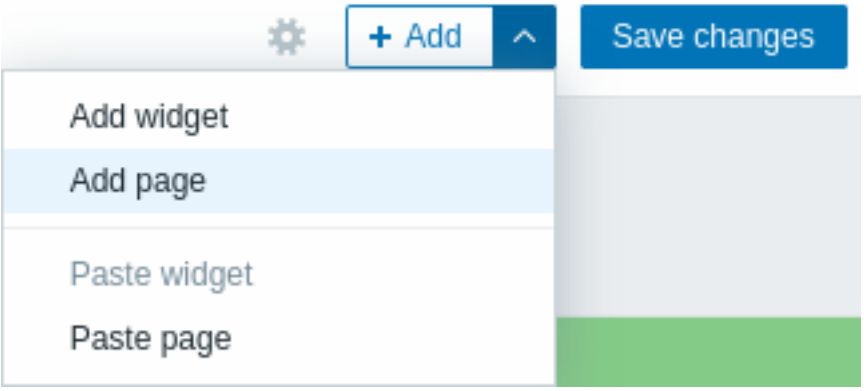
Элементы управления, связанные со слайд-шоу, также доступны в киоск-режиме (в котором отображается только содержимое страницы):

- - остановить слайд-шоу
- - запустить слайд-шоу
- - вернуться на одну страницу назад
- - перейти на следующую страницу

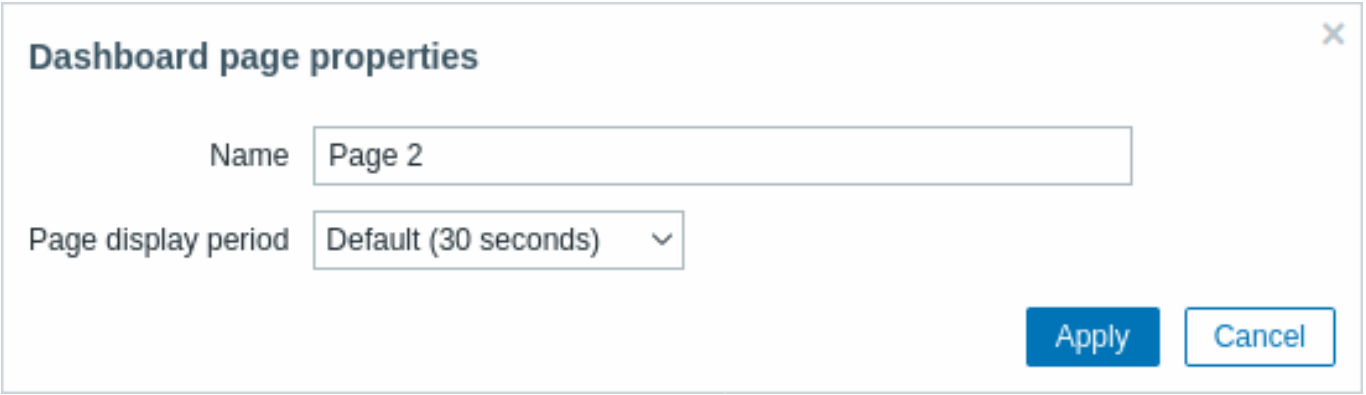
Добавление страниц

Чтобы добавить новую страницу на панель:

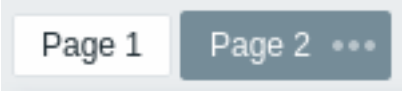
- Убедитесь, что панель находится в режиме редактирования
- Нажмите стрелку рядом с кнопкой *Add* и выберите пункт *Add page*




- Заполните общие параметры страницы и нажмите *Apply*. Если оставить имя пустым, страница будет добавлена с именем `Page N`, где 'N' — порядковый номер страницы. Период отображения страницы позволяет настроить, как долго страница показывается в слайд-шоу.




Будет добавлена новая страница, что будет обозначено новой вкладкой (*Page 2*).



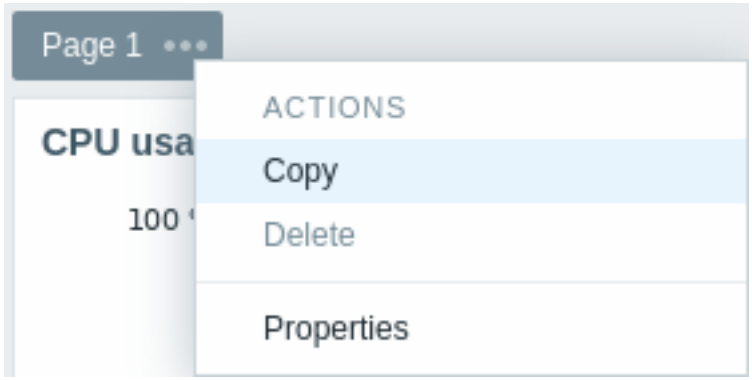
Страницы можно упорядочивать, перетаскивая вкладки страниц. При изменении порядка сохраняются исходные имена страниц. Всегда можно перейти на любую страницу, щелкнув ее вкладку.

Когда новая страница добавлена, она пустая. Вы можете добавить на нее виджеты, как описано выше.

Копирование и вставка страниц

Страницы панели мониторинга можно копировать и вставлять, что позволяет создавать новую страницу с свойствами существующей. Их можно вставлять из той же панели мониторинга или из другой панели мониторинга.

Чтобы вставить существующую страницу в панель мониторинга, сначала скопируйте ее, используя меню страницы:

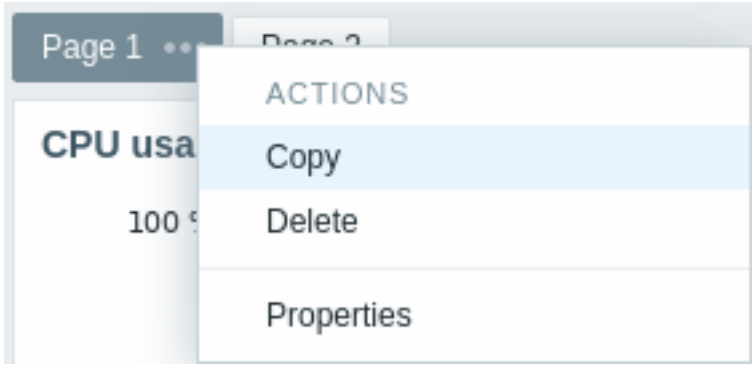



Чтобы вставить скопированную страницу:

- Убедитесь, что панель мониторинга находится в режиме редактирования
- Нажмите стрелку рядом с кнопкой *Add* и выберите пункт *Paste page*

Меню страницы

Меню страницы можно открыть, щелкнув по трем точкам рядом с названием страницы:

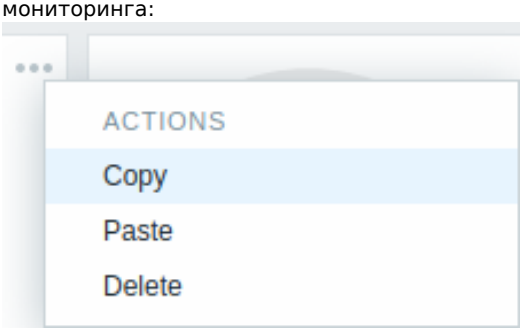
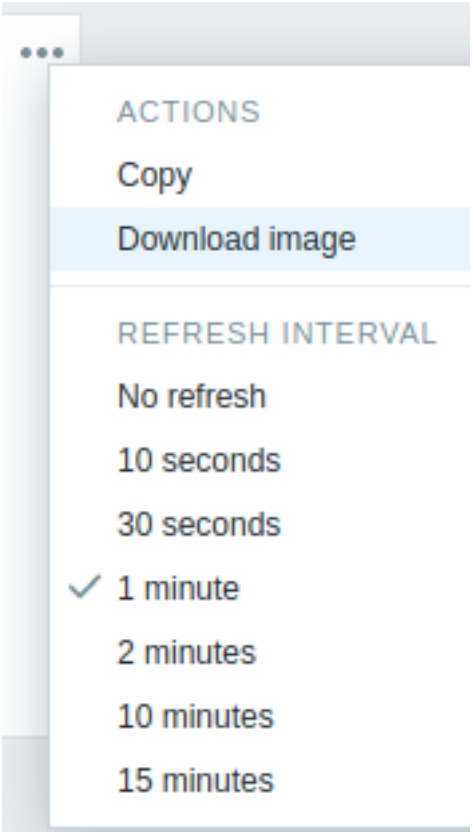



Оно содержит следующие параметры:

- *Copy* - копировать страницу
- *Delete* - удалить страницу (страницы можно удалять только в режиме редактирования панели)
- *Properties* - настроить параметры страницы (название и период отображения страницы в слайд-шоу)

Меню виджета

Меню виджета содержит разные параметры в зависимости от того, находится ли панель мониторинга в режиме редактирования или просмотра:

| Меню виджета | Параметры |
|---|---|
| В режиме редактирования панели мониторинга:<br>  | *Copy* - скопировать виджет.<br>*Paste* - вставить скопированный виджет поверх этого виджета. Этот параметр недоступен, если ни один виджет не был скопирован.<br>*Delete* - удалить виджет. |
| В режиме просмотра панели мониторинга:<br>  | *Copy* - скопировать виджет.<br>*Download image* - скачать виджет в виде изображения PNG. Этот параметр доступен для виджетов, которые поддерживают эту функцию (см. страницы отдельных виджетов). В зависимости от настроек браузера изображение либо автоматически сохраняется в папку Downloads, либо открывается диалог сохранения, позволяющий выбрать расположение файла.<br>*Refresh interval* - выбрать частоту обновления содержимого виджета. |

Права доступа к панелям

Права доступа к панелям для обычных пользователей и для пользователей типа «Администратор» (при условии, что для назначенной им роли пользователя включён доступ к панелям и опция *Создание и изменение панелей (Create and edit dashboards)*) ограничены так, как описано ниже.

- Они могут видеть и клонировать публичные панели, даже если никакие пользователи или группы пользователей не добавлены в настройки общего доступа панели.
- Они могут видеть и клонировать приватные панели, если у них имеются по крайней мере права на их *Чтение*, заданные посредством настроек общего доступа.
- Они могут редактировать и удалять панели, только если у них имеются права на их *Чтение/запись*, заданные посредством настроек общего доступа.
- Они не могут изменить владельца панели.

**1. Виджеты панели**

Обзор

В этом разделе приведены сведения о параметрах, общих для виджетов панели.

Общие параметры

Следующие параметры являются общими для каждого виджета:

| | |
|---|---|
| *Имя* | Введите имя виджета. |
| *Интервал обновления* | Настройте интервал обновления по умолчанию.<br><br>Интервалы обновления виджетов по умолчанию варьируются от *Без обновления* до *15 минут* в зависимости от типа виджета. Например:<br>- *Без обновления* для виджета *URL*;<br>- *1 минута* для виджета *Журнал действий*;<br>- *15 минут* для виджета *Часы*.<br><br>Интервалы обновления можно задать как значение по умолчанию для всех пользователей. Переключите панель в режим редактирования, нажмите кнопку редактировать виджет и выберите нужный интервал обновления из раскрывающегося списка.<br><br>Каждый пользователь также может задать собственный интервал обновления виджета. В режиме просмотра панели нажмите кнопку с тремя точками на виджете и выберите нужный интервал обновления из раскрывающегося списка. Обратите внимание, что уникальный интервал обновления пользователя имеет приоритет над настройкой виджета и сохраняется даже при изменении настройки виджета. |
| *Показывать заголовок* | Установите флажок, чтобы постоянно отображать заголовок виджета.<br>Если флажок не установлен, заголовок скрыт для экономии места и становится видимым только при наведении курсора на виджет (как в режиме просмотра, так и в режиме редактирования). Заголовок также отображается полупрозрачно при перетаскивании виджета в новое место. |

Специфические параметры

Чтобы увидеть специфические параметры для каждого виджета, перейдите на страницы отдельных виджетов:

- Журнал действий
- Часы
- Состояние обнаружения
- Избранные графики
- Избранные карты
- Шкала
- Геокарта
- График
- График (классический)
- Прототип графика
- Соты
- Доступность узла сети
- Карточка узла сети
- Навигатор узлов сети
- Карточка элемента данных
- История элемента данных
- Навигатор элементов данных
- Значение элемента данных
- Карта
- Дерево навигации по картам
- Круговая диаграмма
- Узлы сети с проблемами
- Проблемы
- Проблемы по уровню критичности
- Отчет SLA
- Сведения о системе
- Топ узлов сети
- Топ элементов данных
- Топ триггеров
- Обзор триггеров
- URL
- Мониторинг веб-сайтов

Динамические параметры

Многие виджеты имеют параметры, которые позволяют им использовать конфигурационные данные совместно с другими виджетами или панелями.

Группы узлов сети, Узлы сети, Элемент данных, Список элементов данных

Параметры *Группы узлов сети*, *Узлы сети*, *Элемент данных* и *Список элементов данных* позволяют выбрать либо соответствующие сущности, либо источник данных, содержащий группы узлов сети, узлы сети или элементы данных, для которых виджет может отображать данные.

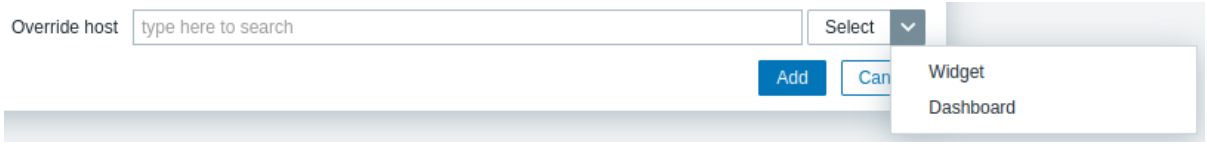
Для параметров *Группы узлов сети*, *Элемент данных* и *Список элементов данных* источником данных может быть совместимый виджет с той же панели.

Для параметра *Узлы сети* источником данных может быть совместимый виджет с той же панели или сама панель.

**Note:**
Виджет *Map* также может передавать данные о группах узлов сети и узлах сети совместимым виджетам. Дополнительные сведения см. в разделе Поведение виджета.

Переопределение узла сети

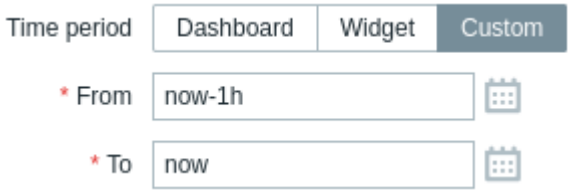
Параметр *Override host* позволяет выбрать источник данных, содержащий узел сети, для которого виджет может отображать данные. Источником данных может быть совместимый виджет с той же панели мониторинга или сама панель мониторинга.




- Чтобы указать совместимый виджет, введите его имя и выберите его. В качестве альтернативы нажмите кнопку *Select* (или кнопку раскрывающегося списка, затем "Widget"), чтобы открыть всплывающее окно со списком доступных виджетов.
- Чтобы указать панель мониторинга, нажмите кнопку раскрывающегося списка, затем "Dashboard". После сохранения панели мониторинга поле *Host* (для выбора узлов сети) появится в верхней части панели мониторинга.

Период времени

Параметр *Период времени (Time period)* позволяет выбирать источник данных, содержащий период, для которого виджет может отобразить данные. Источником данных может быть совместимый виджет с той же панели, сама панель либо период времени, настроенный на самом виджете.



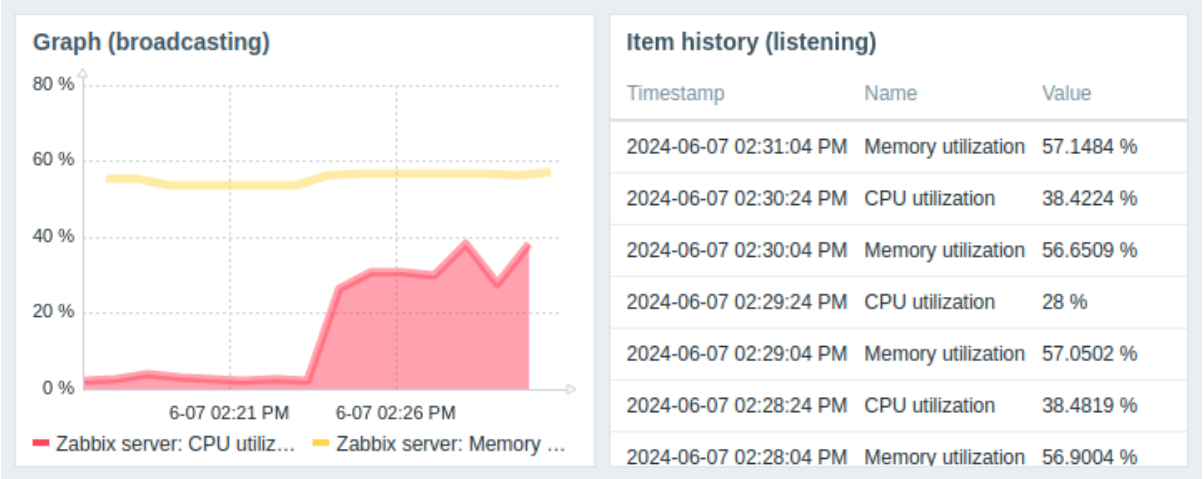

- Чтобы указать совместимый виджет, переключите *Период времени (Time period)* в положение «Виджет (Widget)», введите имя виджета и выберите его. Альтернативный вариант — нажмите кнопку *Выбрать (Select)*, чтобы открыть всплывающее окно доступных виджетов.
- Чтобы указать панель, переключите *Период времени (Time period)* в положение «Панель (Dashboard)». После сохранения панели в её верхней части появится селектор *Период времени (Time period)*.
- Чтобы настроить период времени в самом виджете, переключите *Период времени (Time period)* в положение «Пользовательское (Custom)» и введите или выберите начальное и конечное время периода.

**Note:**
Независимо от настройки параметра *Период времени (Time period)* виджета, (другие) совместимые виджеты могут использовать его в качестве источника данных для периода времени.

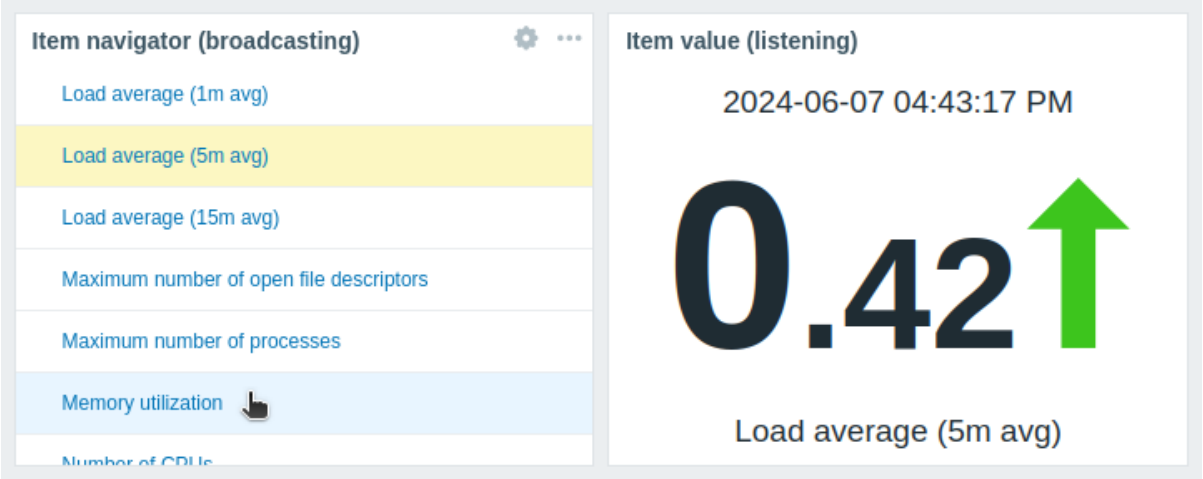
Поведение виджетов

Виджеты различаются по тому, как они **передают** данные другим виджетам.

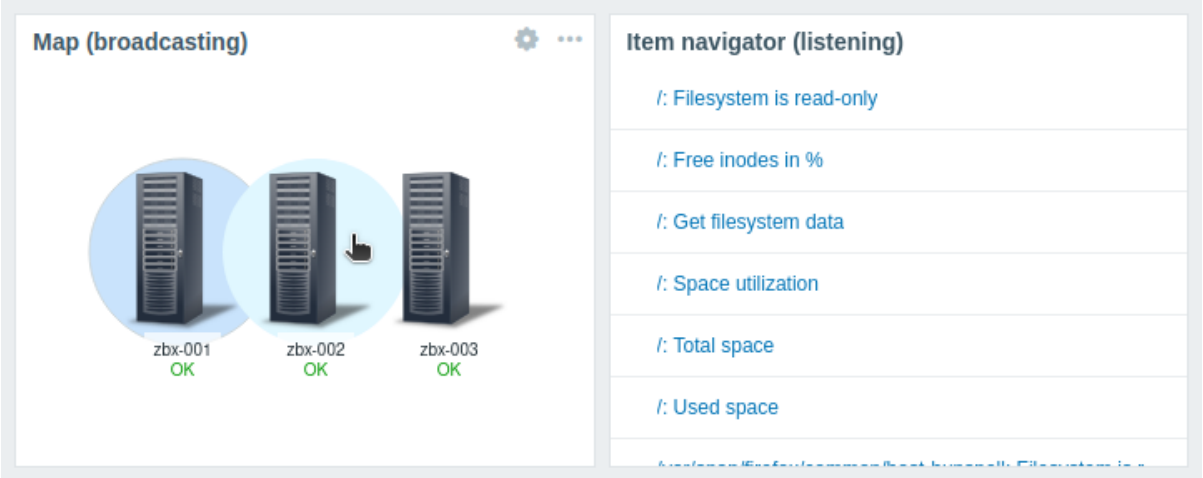
Все виджеты, способные передавать данные, начинают делать это автоматически сразу после создания. Например, виджет *Graph* немедленно передает данные о временном периоде виджетам, которые их принимают.




Виджеты, поддерживающие выбор сущности (например, *Item navigator*), автоматически передают данные для первой доступной сущности. Например, виджет *Item navigator* передает данные для первого элемента данных в своем списке элементов данных. Когда выбирается другой элемент данных, он передает данные для этого элемента. При наведении курсора элемент подсвечивается светло-голубым цветом; при выборе — желтым.



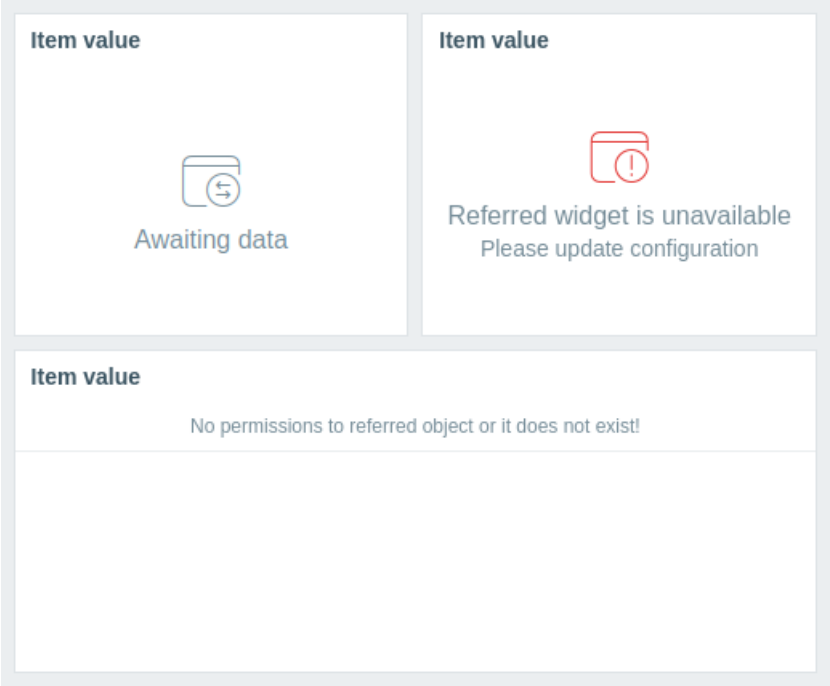

Виджет *Map* также автоматически передает данные для первой доступной сущности (например, узел сети), которая находится ближе всего к верхнему левому углу области просмотра виджета (для виджета *Geomap* — ближе всего к центру области просмотра виджета). Когда выбирается другая сущность, он передает данные для этой сущности. При наведении курсора сущность подсвечивается светло-голубым цветом; при выборе — темно-синим цветом.




У виджетов также есть особое поведение при **получении** данных от других виджетов:

- Если виджет-источник данных не передает данные, принимающий виджет переходит в состояние *Awaiting data*.

- Если виджет-источник данных был удален, заменен несовместимым виджетом или перемещен на другую страницу панели, принимающий виджет переходит в состояние *Referred widget is unavailable*.
- Если указанный узел сети в источнике данных (виджете или панели) не содержит сущность, настроенную в принимающем виджете (элемент данных, график, карту и т. д.), или если у пользователя нет прав доступа к узлу сети, принимающий виджет отображает следующее сообщение: *"No permissions to referred object or it does not exist!"*




Сведения о возможностях передачи и получения данных для каждого виджета см. в разделе Совместимость виджетов.

Совместимость виджетов

Некоторые виджеты могут передавать данные конфигурации другим виджетам, некоторые могут принимать данные, а некоторые умеют и то и другое. Например:

- Виджет *Action log* может только получать данные о периоде времени от виджетов *Graph*, *Graph (classic)* и *Graph prototype*.
- Виджет *Geomap* может передавать данные о узлах сети виджетам, которые их принимают (*Honeycomb*, *Top items* и т. д.), а также может принимать данные о группах узлов сети и узлах сети от виджетов, которые их передают (*Honeycomb*, *Problem hosts* и т. д.).
- Виджет *Clock* не может ни передавать, ни принимать какие-либо данные.

В следующей таблице указаны возможности передачи и приема данных для каждого виджета.

| Widget | Broadcasts | Listens |
|---|---|---|
| *Action log* | - | Период времени |
| *Clock* | - | - |
| *Discovery status* | - | - |
| *Favorite graphs* | - | - |
| *Favorite maps* | - | - |
| *Gauge* | - | Узлы сети, Элементы данных |
| *Geomap* | Узлы сети | Группы узлов сети, Узлы сети |
| *Graph* | Период времени | Узлы сети, Элементы данных, Период времени |
| *Graph (classic)* | Период времени | Узлы сети, Элементы данных, Графики, Период времени |
| *Graph prototype* | Период времени | Узлы сети, Период времени |
| *Honeycomb* | Узлы сети, Элементы данных | Группы узлов сети, Узлы сети |
| *Host availability* | - | Группы узлов сети |
| *Host card* | - | Узлы сети |
| *Host navigator* | Узлы сети | Группы узлов сети |
| *Item card* | - | Узлы сети, Элементы данных, Период времени |
| *Item history* | Элементы данных | Узлы сети, Период времени |

| Widget | Broadcasts | Listens |
|---|---|---|
| *Item navigator* | Элементы данных | Группы узлов сети, Узлы сети |
| *Item value* | - | Узлы сети, Элементы данных, Период времени |
| *Map* | Группы узлов сети, Узлы сети | Карты |
| *Map navigation tree* | Карты | - |
| *Pie chart* | - | Элементы данных, Период времени |
| *Problem hosts* | Группы узлов сети | Группы узлов сети, Узлы сети |
| *Problems* | События | Группы узлов сети, Узлы сети |
| *Problems by severity* | Группы узлов сети | Группы узлов сети, Узлы сети |
| *SLA report* | - | - |
| *System information* | - | - |
| *Top hosts* | Узлы сети | Группы узлов сети, Узлы сети |
| *Top items* | - | Группы узлов сети, Узлы сети |
| *Top triggers* | - | Период времени |
| *Trigger overview* | - | Группы узлов сети, Узлы сети |
| *URL* | - | Узлы сети |
| *Web monitoring* | Группы узлов сети | Группы узлов сети, Узлы сети |

1. Журнал действий

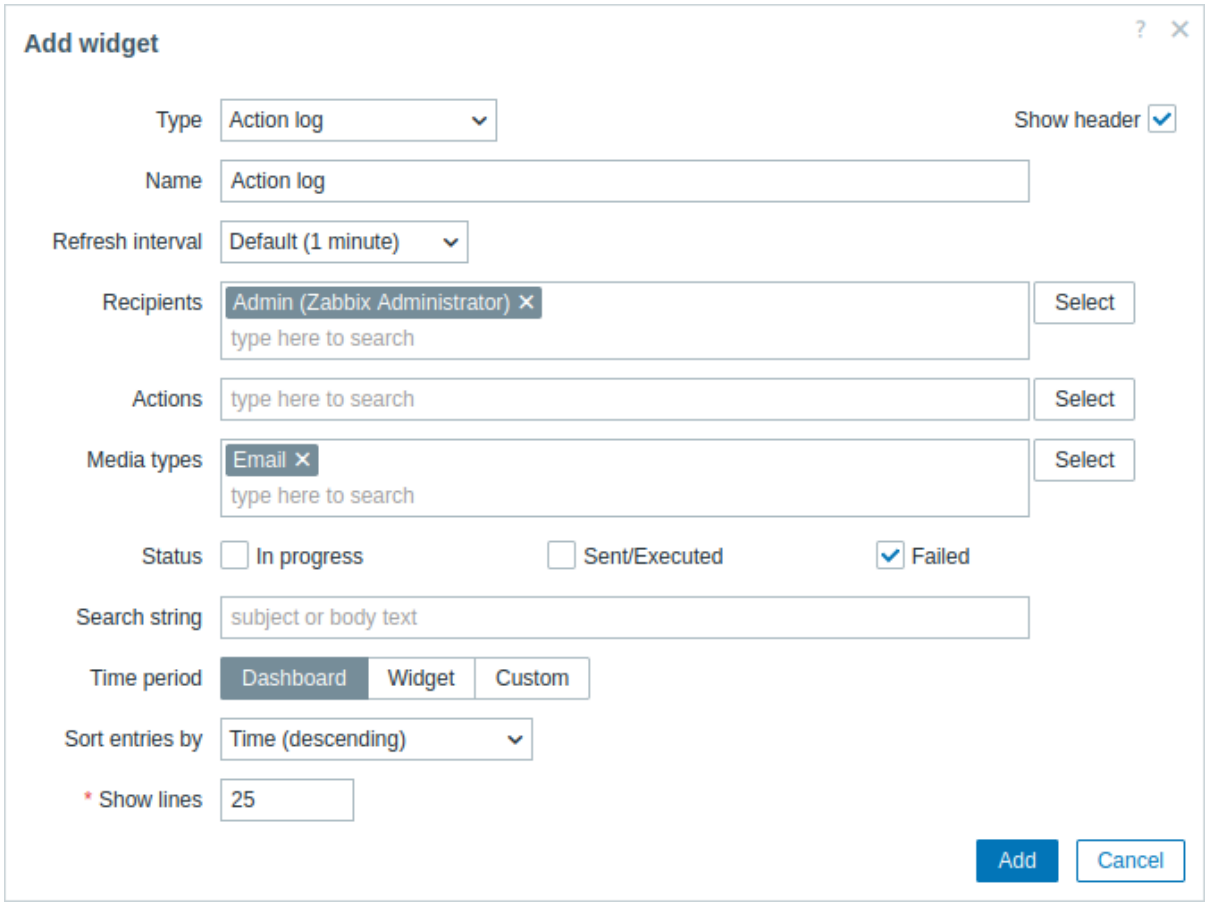
Обзор

В виджете журнала действий можно отображать сведения о действиях операций (уведомления, удаленные команды). Он дублирует информацию из *Отчеты* → *Журнал действий*.

Можно отображать до 1000 записей.

Конфигурация

Чтобы настроить, выберите *Action log* в качестве типа:




Помимо параметров, общих для всех виджетов, можно задать следующие специфические параметры:

| | |
|---|---|
| *Recipients* | Фильтрация записей по получателям. Это поле поддерживает автодополнение, поэтому при вводе имени получателя будет предложен выпадающий список подходящих получателей. Если получатели не выбраны, будут отображаться сведения о действиях операций для всех получателей. |
| *Actions* | Фильтрация записей по действиям. Это поле поддерживает автодополнение, поэтому при вводе имени действия будет предложен выпадающий список подходящих действий. Если действия не выбраны, будут отображаться сведения о действиях операций для всех действий. |
| *Media types* | Фильтрация записей по типам медиа. Это поле поддерживает автодополнение, поэтому при вводе имени типа медиа будет предложен выпадающий список подходящих типов медиа. Если типы медиа не выбраны, будут отображаться сведения о действиях операций для всех типов медиа. |
| *Status* | Установите флажок, чтобы фильтровать записи по соответствующему статусу:<br>**In progress** - отображаются действия операций, которые выполняются;<br>**Sent/Executed** - отображаются действия операций, для которых было отправлено уведомление или которые были выполнены;<br>**Failed** - отображаются действия операций, завершившиеся с ошибкой. |
| *Search string* | Фильтрация записей по содержимому сообщения/удаленной команды. Если ввести здесь строку, будут отображаться только те действия операций, в сообщении/удаленной команде которых содержится введенная строка. Макросы не подставляются. |
| *Time period* | Фильтрация записей по временному периоду. Выберите источник данных для временного периода:<br>**Dashboard** - используйте селектор *Time period* в качестве источника данных;<br>**Widget** - используйте совместимый виджет, указанный в параметре *Widget*, в качестве источника данных;<br>**Custom** - используйте временной период, указанный в параметрах *From* и *To*, в качестве источника данных; если задано, в правом верхнем углу виджета будет отображаться значок часов, показывающий установленное время при наведении курсора. |
| *Widget* | Введите или выберите совместимый виджет в качестве источника данных для временного периода.<br>Этот параметр доступен, если для *Time period* установлено значение ”Widget”. |
| *From* | Введите или выберите начало временного периода.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* установлено значение ”Custom”. |
| *To* | Введите или выберите конец временного периода.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* установлено значение ”Custom”. |
| *Sort entries by* | Сортировать записи по:<br>**Time** (по убыванию или возрастанию);<br>**Type** (по убыванию или возрастанию);<br>**Status** (по убыванию или возрастанию);<br>**Recipient** (по убыванию или возрастанию). |
| *Show lines* | Укажите, сколько строк журнала действий будет отображаться в виджете. |

2. Часы

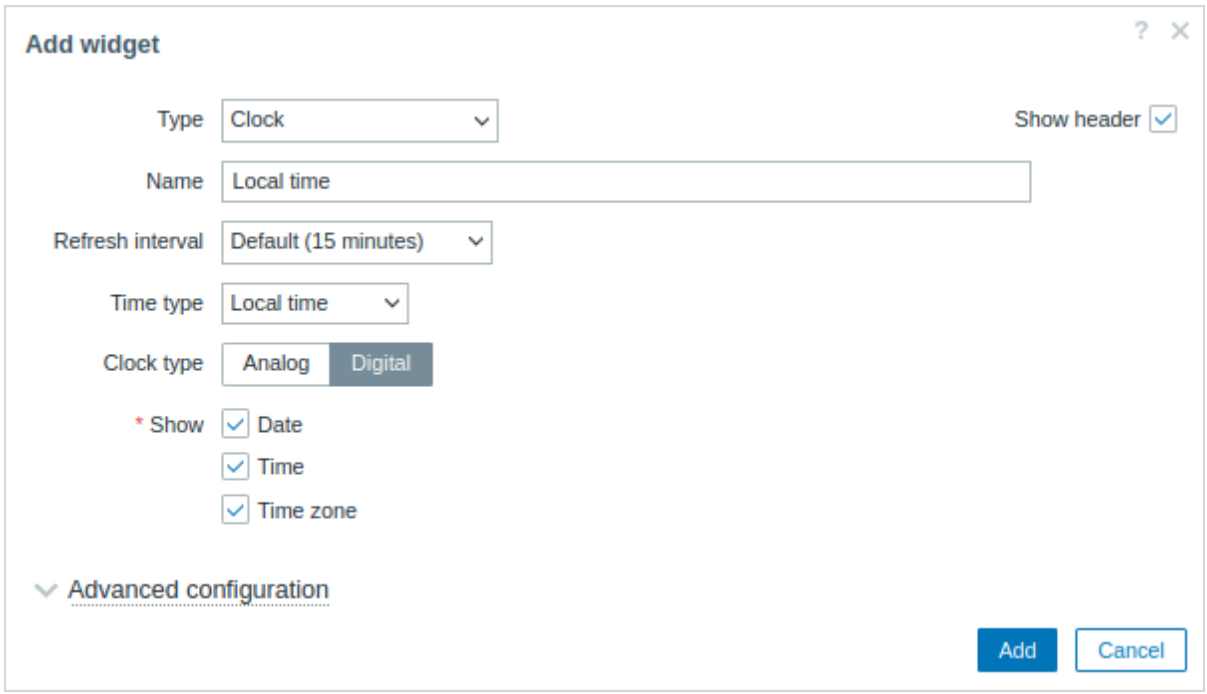
Обзор

В виджете часов вы можете отображать местное время, время сервера или указанного узла сети.

Могут отображаться как аналоговые, так и цифровые часы:

Конфигурация

Чтобы настроить, выберите *Clock* в качестве типа:



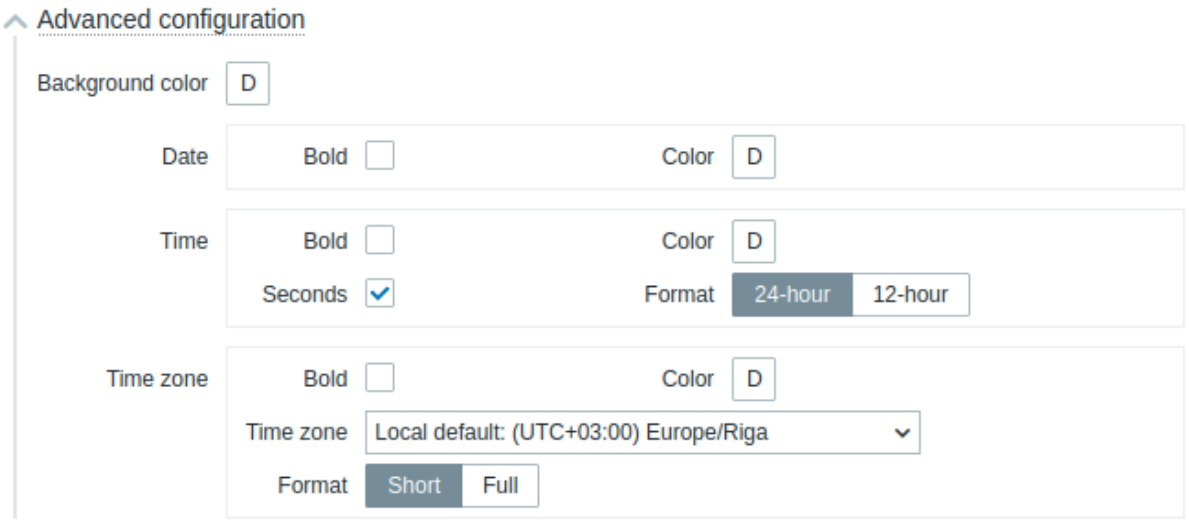

Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Time type* | Выберите локальное время, время сервера или указанного узла сети.<br>Время сервера будет идентично часовому поясу, заданному глобально или для пользователя Zabbix. |
| *Item* | Выберите элемент данных для отображения времени. Чтобы отображать время узла сети, используйте элемент данных `system.localtime[local]` item. Этот элемент данных должен существовать на узле сети.<br>Это поле доступно только при выборе *Host time*. |
| *Clock type* | Выберите тип часов:<br>**Analog** - аналоговые часы<br>**Digital** - цифровые часы |
| *Show* | Выберите информационные единицы (дата, время, часовой пояс) для отображения на цифровых часах.<br>Размер единицы динамический: он заполняет все доступное пространство виджета в зависимости от отображаемых единиц, размера виджета и масштаба отображения.<br>Это поле доступно только если в поле *Clock type* выбрано "Digital"; необходимо выбрать как минимум один тип информационной единицы. |
| *Advanced configuration* | Нажмите заголовок *Advanced configuration*, чтобы отобразить параметры advanced configuration для цифровых часов.<br>Этот раздел доступен только если в поле *Clock type* выбрано "Digital". |

Расширенная настройка

Опции расширенной настройки доступны в сворачиваемом разделе *Расширенная настройка (Advanced configuration)*, и только для тех элементов, которые выбраны в поле *Показать (Show)* (смотрите выше).

Кроме того, расширенная настройка позволяет поменять цвет фона для целого виджета.



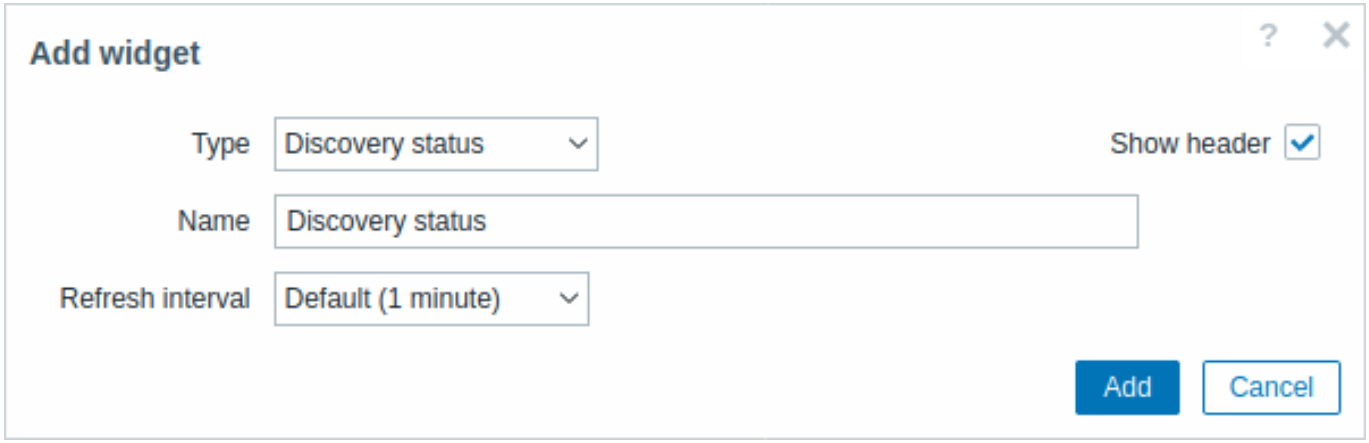

| | |
|---|---|
| *Цвет фона (Background color)* | Выберите из палитры цветов цвет фона.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| ***Дата (Date)*** | |
| *Жирный (Bold)* | Установите флажок, чтобы дата отображалась жирным шрифтом. |
| *Цвет (Color)* | Выберите из палитры цветов цвет даты.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| ***Время (Time)*** | |
| *Жирный (Bold)* | Установите флажок, чтобы время отображалось жирным шрифтом. |
| *Цвет (Color)* | Выберите из палитры цветов цвет времени.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| *Секунды (Seconds)* | Установите флажок для отображения секунд. В противном случае будут отображаться только часы и минуты. |
| *Формат (Format)* | Выберите, отображать время в 24-часовом или 12-часовом формате. |
| ***Часовой пояс (Time zone)*** | |

| | |
|---|---|
| *Жирный (Bold)* | Установите флажок, чтобы часовой пояс отображался жирным шрифтом. |
| *Цвет (Color)* | Выберите из палитры цветов цвет часового пояса.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| *Часовой пояс (Time zone)* | Выберите часовой пояс. |
| *Формат (Format)* | Выберите, отображать часовой пояс в кратком формате (например, `New York`) или в полном формате (например, `(UTC-04:00) America/New York`). |

3 Статус обнаружения

Обзор

Этот виджет отображает сводку состояния по активным правилам сетевого обнаружения.



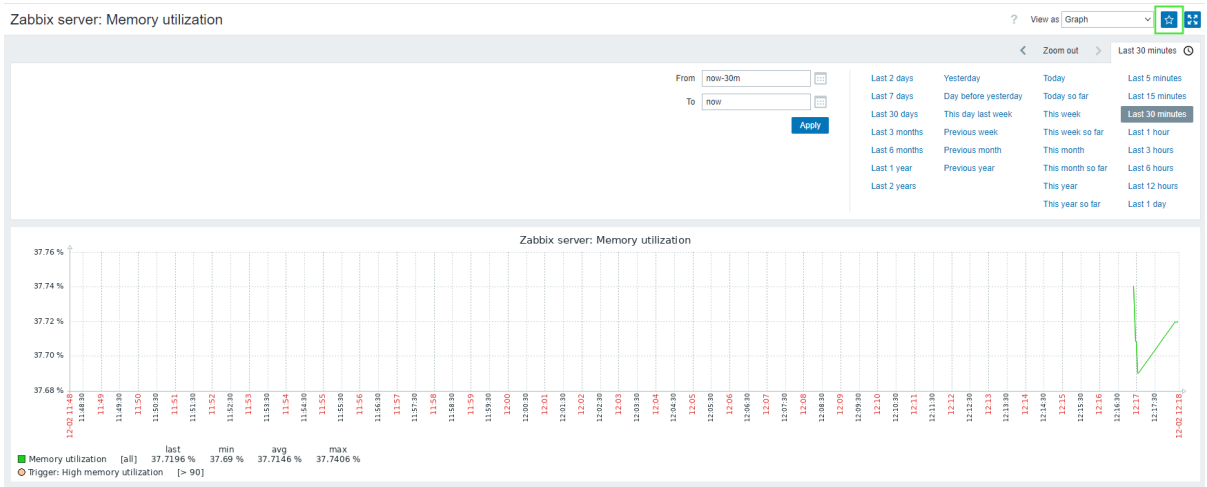

Все параметры конфигурации являются общими для всех виджетов.

4 Избранные графики

Обзор

Этот виджет содержит ярлыки к наиболее нужным графикам, отсортированным по алфавиту.

Список ярлыков заполняется, когда вы просматриваете график в Monitoring -> Latest data -> Graphs, а затем нажимаете его кнопку *Добавить в избранное*.




Все параметры конфигурации являются общими для всех виджетов.

5 Избранных карт

Обзор

Этот виджет содержит ярлыки к наиболее нужным картам, отсортированным по алфавиту.

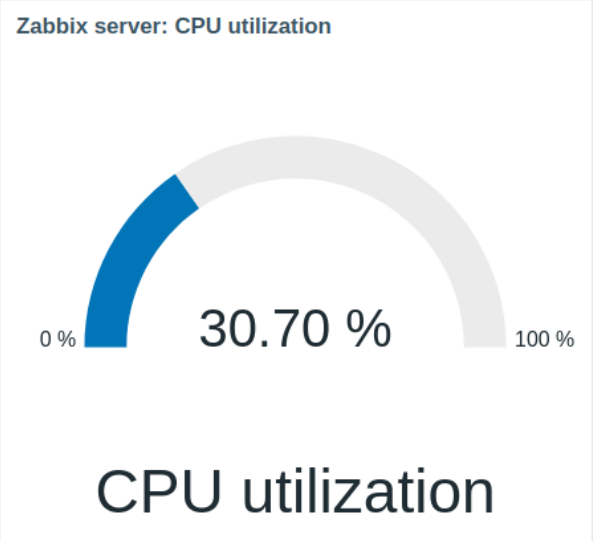
Список ярлыков заполняется, когда вы просматриваете карту, а затем нажимаете ее кнопку *Добавить в избранное*.

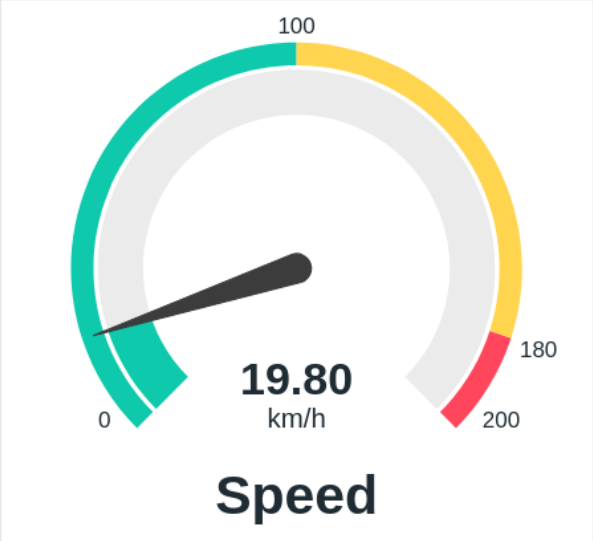
Все параметры конфигурации являются общими для всех виджетов.

6 Индикатор

Обзор

Виджет gauge отображает значение одного элемента данных в виде шкалы.

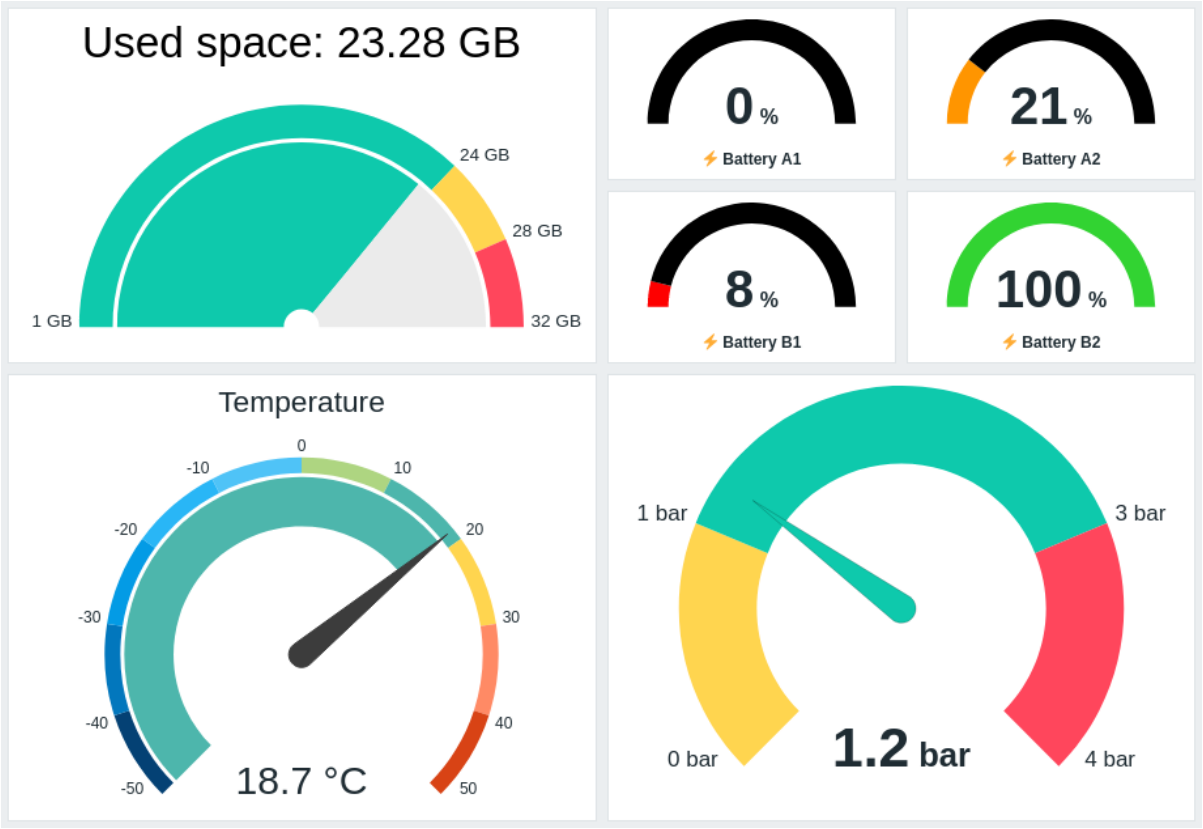



При настройке виджет может состоять из следующих элементов:

- Описание элемента данных (например, ”CPU utilization”, ”Speed”)
- Значение элемента данных (например, ”30.70”, ”19.80”)
- Единицы значения элемента данных (например, ”%”, ”km/h”)
- Шкала gauge (например, ”0%/100%”, ”0/100/180/200”)
- Дуга gauge (дуга значения gauge и дуга порогов gauge)
- Стрелка gauge

Виджет можно визуально настроить с помощью параметров расширенной настройки, чтобы создать широкий спектр визуальных стилей:



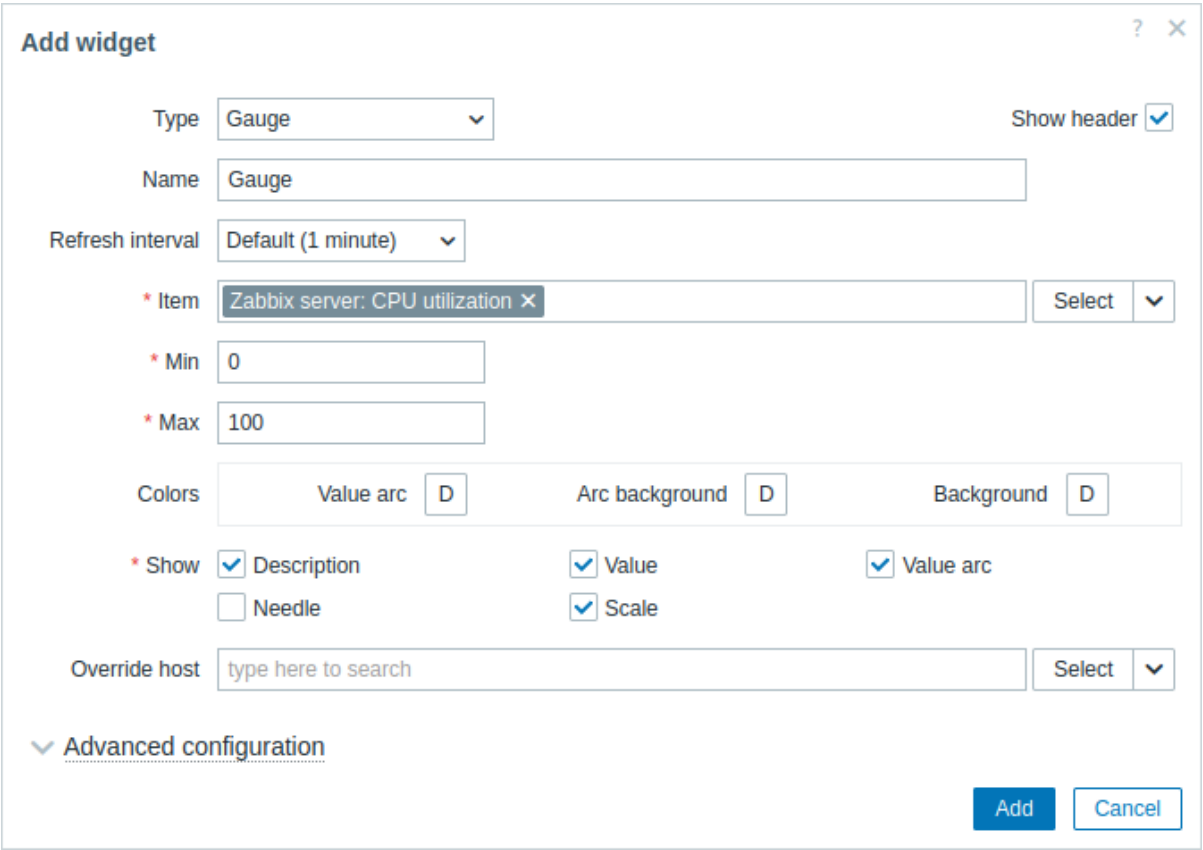

Виджет gauge может отображать только числовые значения. Отображение двоичных значений не поддерживается.

При нажатии на виджет gauge открывается график для элемента данных.

Информацию, отображаемую в виджете *Gauge*, можно загрузить в виде изображения PNG, выбрав параметр *Download image* в меню виджета.

Настройка

Для настройки выберите *Радиальный датчик (Gauge)* в качестве типа:



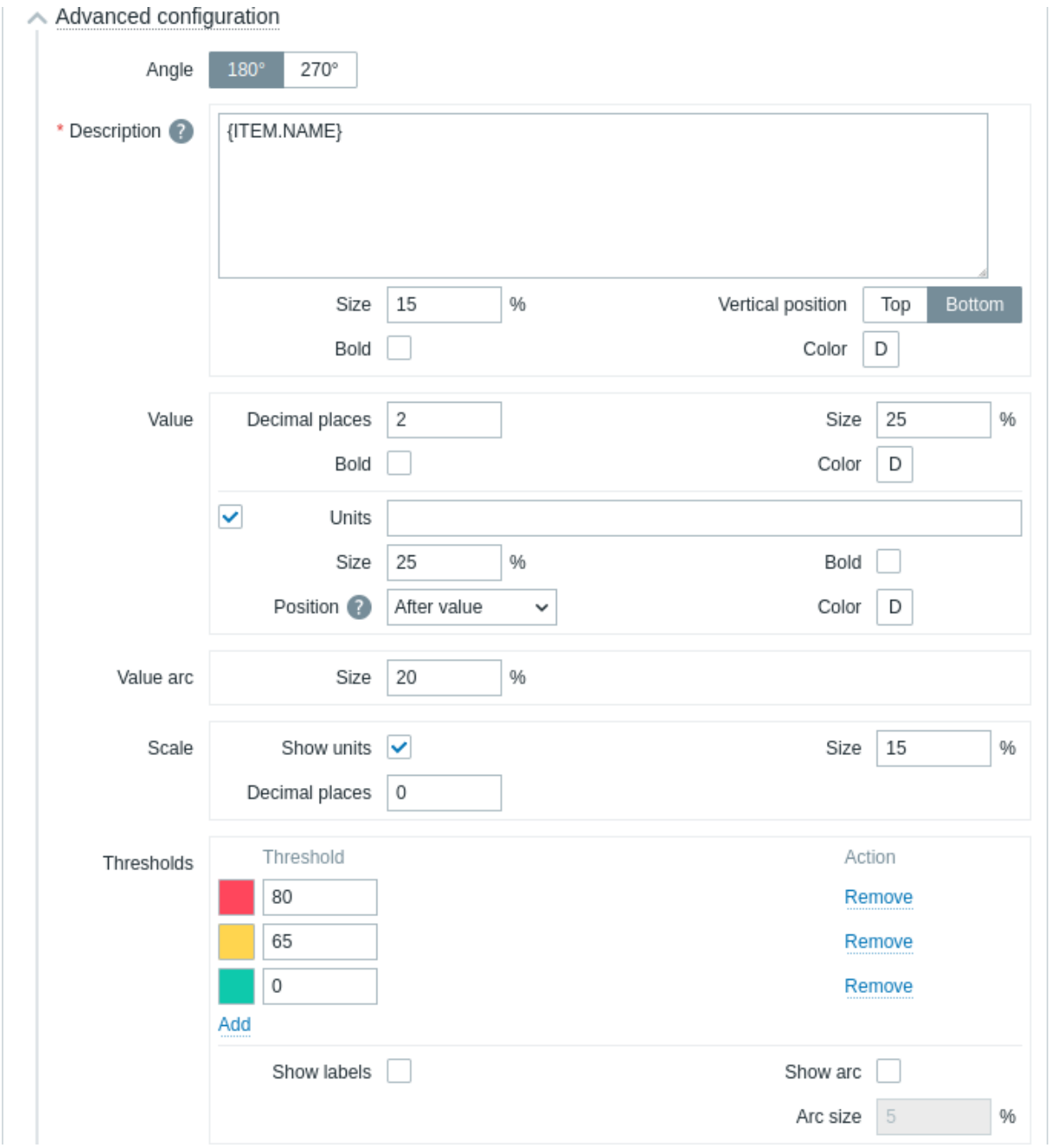

В дополнение к параметрам, которые являются общими для всех виджетов, вы можете выставить следующие

специфичные опции:

| | |
|---|---|
| *Элемент данных (Item)* | Выберите элемент данных.<br>В качестве альтернативы выберите для элементов данных в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени элемента данных будет автоматически предложен список совпадающих элементов данных в выпадающем меню.<br>Обратите внимание, что вы можете выбрать только элементы данных, которые возвращают числовые значения. |
| *Мин (Min)* | Введите минимальное значение шкалы.<br>Поддерживаются суффиксы (например, «1d», «2w», «4K», «8G»). Поддерживаются преобразования значений. |
| *Макс (Max)* | Введите максимальное значение шкалы.<br>Поддерживаются суффиксы (например, «1d», «2w», «4K», «8G»). Поддерживаются преобразования значений. |
| *Цвета (Colors)* | Выберите цвет из палитры:<br>**Значение дуги (Value arc)** — выберите цвет дуги значений радиального датчика;<br>**Фон дуги (Arc background)** — выберите цвет фона дуги значений и дуги пороговых значений радиального датчика;<br>**Фон (Background)** — выберите цвет фона радиального датчика.<br><br>«D» обозначает цвет по умолчанию, который зависит от темы веб-интерфейса. Если заданы *Пороги (Thresholds)*, цвет по умолчанию для **Значения дуги (Value arc)** зависит от цвета порога. Чтобы вернуться к цвету по умолчанию, нажмите кнопку *Использовать умолчание (Use default)* в палитре. |
| *Показать (Show)* | Отметьте флажок для отображения соответствующего элемента радиального датчика — описание (description), значение (value), дага значений (value arc), стрелка (needle), шкала (scale) (минимальное и максимальное значение радиального датчика в начале и в конце дуги значений). Снимите флажок, чтобы скрыть. Должен быть выбран хотя бы один элемент.<br><br>Обратите внимание, что стрелка и шкала радиального датчика могут отображаться, если отображается дуга значений или дуга пороговых значений (смотрите опции расширенной настройки). Также обратите внимание, что если отображается стрелка радиального датчика, значение располагается под стрелкой; если же стрелка скрыта, значение выравнивается по нижнему краю дуги. |
| *Переопределение узла сети (Override host)* | Выберите для узла сети в качестве источника данных совместимый виджет или панель.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Расширенная настройка (Advanced configuration)* | Нажмите на заголовок *Расширенная настройка (Advanced configuration)* для отображения опций расширенной настройки. Здесь же вы сможете настроить элементы радиального датчика, выбранные в поле *Показать (Show)*. |

Расширенная настройка

Параметры расширенной настройки доступны в сворачиваемом разделе *Расширенная настройка (Advanced configuration)*:




| | |
|---|---|
| *Угол (Angle)* | Выберите угол шкалы (180° или 270°). |
| ***Описание (Description)*** | |
| *Описание (Description)* | Введите описание элемента данных. Это описание может переопределять название элемента данных по умолчанию.<br>Поддерживаются многострочные описания. Допустима комбинация текста и поддерживаемых макросов.<br>Поддерживаются макросы {HOST.*}, {ITEM.*}, {INVENTORY.*} и пользовательские макросы. |
| *Размер (Size)* | Введите высоту шрифта для описания элемента данных (в процентах, относительно общей высоты виджета). |
| *Расположение по вертикали (Vertical position)* | Выберите расположение по вертикали описания элемента данных (по верхнему либо нижнему краю, относительно дуги шкалы). |
| *Жирный (Bold)* | Установите флажок, чтобы описание отображалось жирным шрифтом. |
| *Цвет (Color)* | Выберите из палитры цветов цвет описания элемента данных.<br>«D» означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| ***Значение (Value)*** | |
| *Число десятичных знаков (Decimal places)* | Введите количество десятичных знаков, отображаемых со значением.<br>Эта опция влияет только на элементы данных, которые возвращают значения с типом числовой (с плавающей точкой). |

| | |
|---|---|
| *Размер (Size)* | Введите высоту шрифта для значения (в процентах, относительно высоты дуги). |
| *Жирный (Bold)* | Установите флажок, чтобы значение отображалось жирным шрифтом. |
| *Цвет (Color)* | Выберите из палитры цветов цвет значения.<br>«D» означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| ***Единицы измерения (Units)*** | |
| *Единицы измерения (Units)* | Установите флажок, чтобы вместе со значением элемента данных отображались единицы измерения.<br>Если вы вводите название единицы измерения, это переопределит единицы измерения, заданные в настройках элемента данных. |
| *Размер (Size)* | Введите высоту шрифта для единиц измерения элемента данных (в процентах, относительно высоты дуги). |
| *Жирный (Bold)* | Установите флажок, чтобы единицы измерения отображались жирным шрифтом. |
| *Расположение (Position)* | Выберите расположение единиц измерения элемента данных (выше, ниже, перед или после, относительно значения элемента данных).<br>Эта опция игнорируется для слудещих единиц измерения, относящихся ко времени: unixtime, uptime, s. |
| *Цвет (Color)* | Выберите из палитры цветов цвет единиц измерения.<br>«D» означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| ***Значение дуги (Value arc)*** | |
| *Размер дуги (Arc size)* | Введите высоту значения дуги радиального датчика (в процентах, относительно радиуса дуги). |
| ***Стрелка (Needle)*** | |
| *Цвет (Color)* | Выберите из палитры цветов цвет стрелки радиального датчика.<br>«D» означает цвет по умолчанию (зависит от темы веб-интерфейса). Если заданы *Пороги*, цвет по умолчанию для стрелки зависит от цвета порога . Чтобы вернуться к значению по умолчанию, нажмите в палитре кнопку *Использовать умолчание (Use default)*. |
| ***Шкала (Scale)*** | |
| *Показывать единицы измерения (Show units)* | Установите флажок, чтобы вместе с минимальным и максимальным значением шкалы отображались единицы измерения. |
| *Размер (Size)* | Введите высоту шрифта для минимального и максимального значения шкалы (в процентах, относительно высоты дуги шкалы). |
| *Число десятичных знаков (Decimal places)* | Введите количество десятичных знаков, отображаемых с минимальным и максимальным значением шкалы.<br>Эта опция влияет только на элементы данных, которые возвращают значения с типом числовой (с плавающей точкой). |
| ***Пороги (Thresholds)*** | |
| *Пороги (Thresholds)* | Нажмите *Добавить (Add)* для добавления порога, выберите цвет порога из палитры и укажите числовое значение.<br>При сохранении список порогов будет отсортирован в возрастающем порядке.<br>Обратите внимание, что цвета, настроенные для порогов, будут правильно отображаться только для числовых элементов данных.<br>Поддерживаются суффиксы (например, «1d», «2w», «4K», «8G»). Поддерживаются преобразования значений. |
| *Показывать подписи (Show labels)* | Установите флажок, чтобы пороговые значения отображались как подписи на шкале. |
| *Показать дугу (Show arc)* | Установите флажок, чтобы отображалась дуга порогов. |
| *Размер дуги (Arc size)* | Введите высоту дуги порогов (в процентах, относительно радиуса дуги значений). |

## 7 Геокарта

### Обзор

Виджет геокарты отображает узлы сети в виде маркеров на географической карте, используя JavaScript библиотеку с

открытым исходным кодом Leaflet для интерактивных карт.

**Note:**
Zabbix предлагает несколько предопределённых поставщиков услуг плиток карты (map tile) и возможность добавить собственного поставщика услуг плиток или даже размещать сами плитки (настраивается в разделе меню *Администрирование → Общие → Географические карты*).

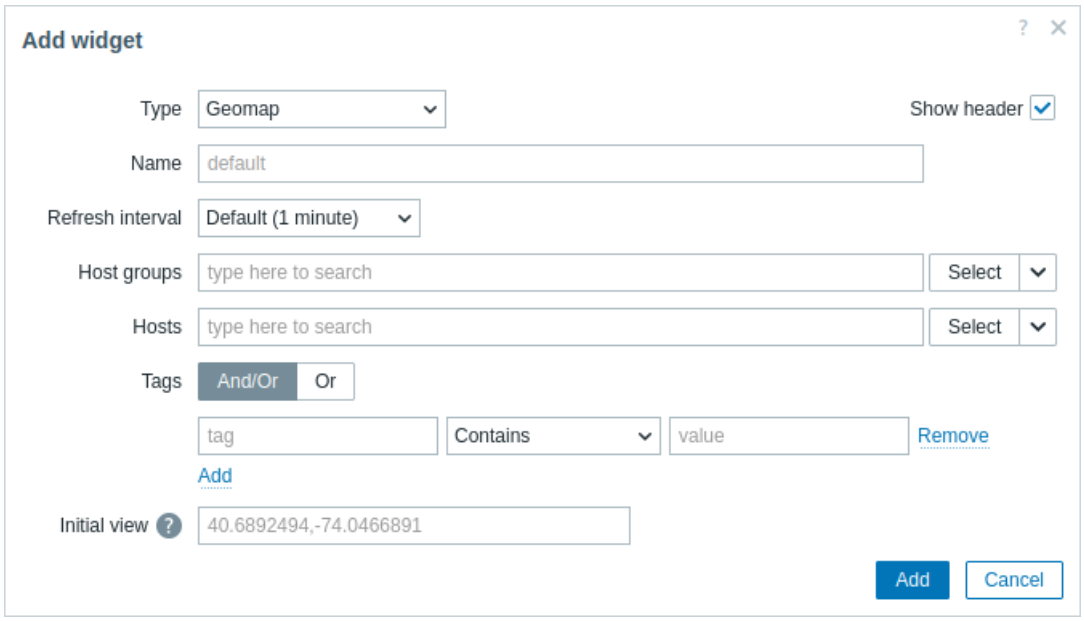
По умолчанию виджет отображает все активированные узлы сети, которые имеют корректные координаты, указанные в в их инвентарных данных. В параметрах виджета можно настроить фильтрацию узлов сети.

Корректные координаты узла сети:

- Широта: от -90 до 90 (может быть целым числом или числом с плавающей точкой)
- Долгота: от -180 до 180 (может быть целым числом или числом с плавающей точкой)

Настройка

Чтобы добавить виджет, выберите *Геокарта (Geomap)* в качестве типа.



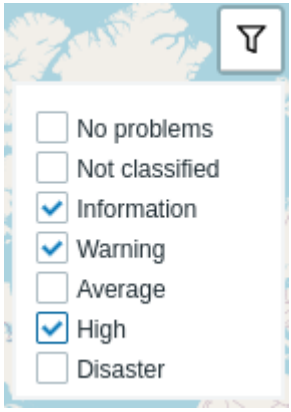

В дополнение к параметрам, которые являются общими для всех виджетов, вы можете выставить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети для отображения на карте.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню.<br>Если ничего не выбрано ни в поле *Группы узлов сети (Host groups)*, ни в поле *Узлы сети (Hosts)*, то будут отображены все узлы сети с корректными координатами.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Узлы сети (Hosts)* | Выберите узлы сети для отображения на карте.<br>В качестве альтернативы выберите для узлов сети в качестве источника данных совместимый виджет или панель.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени узла сети будет автоматически предложен список совпадающих узлов сети в выпадающем меню.<br>Если ничего не выбрано ни в поле *Группы узлов сети (Host groups)*, ни в поле *Узлы сети (Hosts)*, то будут отображены все узлы сети с корректными координатами.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |

| | |
|---|---|
| *Теги*<br>*(Tags)* | Укажите теги для ограничения количества узлов сети, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно указать несколько условий. Поиск соответствия имён тегов всегда чувствителен к регистру.<br><br>По каждому условию доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием *Или (Or)*;<br>**Или (Or)** — достаточно соблюдения одного условия. |
| *Начальный вид*<br>*(Initial view)* | Этот параметр недоступен при настройке виджета на панели шаблона.<br>Координаты центра и необязательный уровень масштабирования (через запятую) для отображения при первоначальной загрузке виджета, в формате < >,< >,< ><br>Если указан начальный масштаб, то виджет Геокарты загружается с данным уровнем масштабирования. В противном случае начальный масштаб рассчитывается как половина от максимального масштаба для конкретного поставщика плиток.<br>Начальный вид игнорируется, если задан вид по умолчанию (см. ниже).<br>Примеры:<br>40.6892494,-74.0466891<br>40.6892494,-122.0466891 |

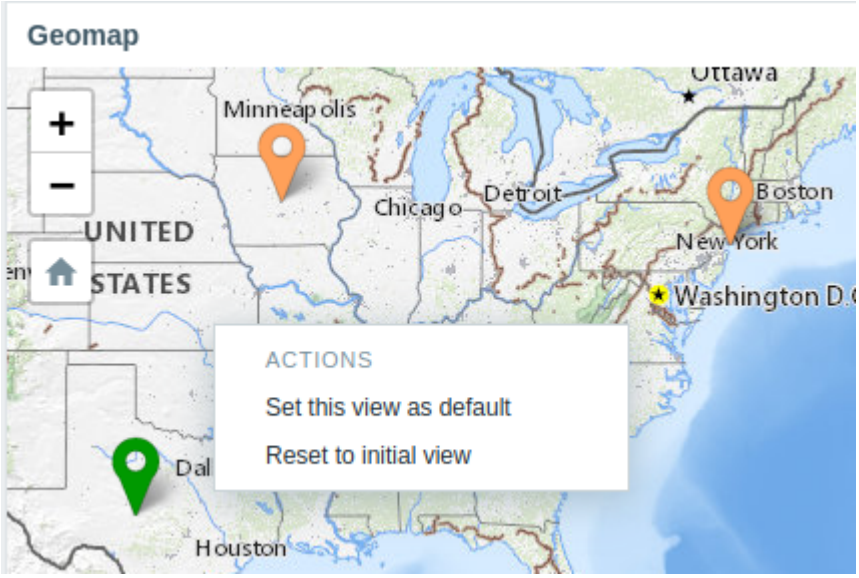
Маркеры узлов сети, отображаемые на карте, имеют цвет самой серьезной проблемы узла сети, а если у узла сети нет проблем, то зеленый цвет. При нажатии на маркер узла сети можно просмотреть видимое имя узла сети и количество нерешенных проблем, сгруппированных по степени серьезности. При нажатии на видимое имя откроется меню узла сети.

Узлы сети, отображаемые на карте, можно фильтровать по степени серьезности проблемы. Нажмите значок фильтра в правом верхнем углу виджета и отметьте нужные степени серьезности.




Можно увеличивать и уменьшать масштаб карты с помощью кнопок плюс и минус в левом верхнем углу виджета, а также с помощью колесика прокрутки мыши или сенсорной панели. Чтобы сделать текущий вид видом по умолчанию, щелкните правой кнопкой мыши в любом месте на карте и выберите *Set this view as default*. Этот параметр переопределит параметр виджета *Initial view* для текущего пользователя. Чтобы отменить это действие, снова щелкните правой кнопкой мыши в любом месте на карте и выберите *Reset to initial view*.

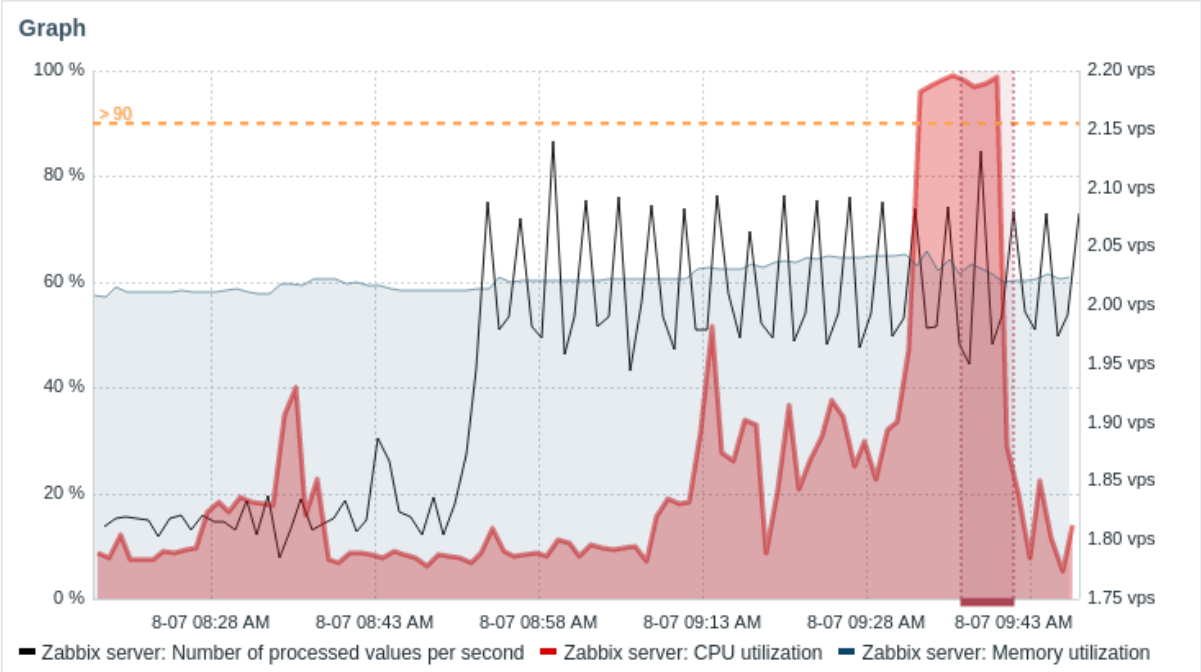
Если задан *Initial view* или *Default view*, вы можете в любой момент вернуться к этому виду, нажав значок домика слева.




8 График

Обзор

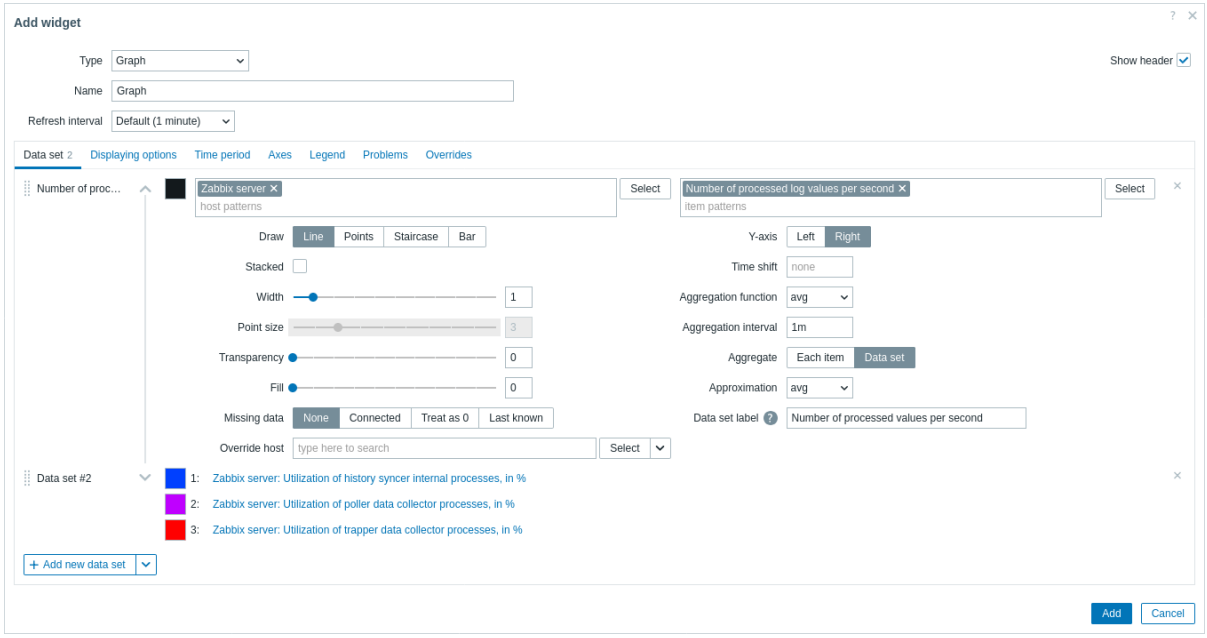
Виджет графика предоставляет современный и универсальный способ визуализации данных, собранных Zabbix, с использованием техники рисования векторных изображений. Этот виджет графика поддерживается начиная с Zabbix 4.0. Обратите внимание, что виджет графика, поддерживаемый до Zabbix 4.0, по-прежнему можно использовать как Graph (classic). См. также раздел *Добавление виджетов* на странице *Панели мониторинга* для получения дополнительных сведений.




Информацию, отображаемую в виджете *Graph*, можно загрузить в виде изображения PNG, выбрав параметр *Download image* в меню виджета.

Настройка

Для настройки выберите *График (Graph)* в качестве типа:




Набор данных

Вкладка **Набор данных** позволяет выбирать данные для графика путем добавления наборов данных. Можно добавить два типа наборов данных:

- *Шаблоны элементов данных* - отображаются данные из совпадающих элементов данных. Можно выбрать один базовый цвет или выбрать строку палитры, чтобы назначить отдельные цвета каждому совпавшему элементу данных.
- *Список элементов данных* - отображаются данные из выбранных элементов данных. Для каждого элемента можно выбрать цвет отдельно с помощью палитры.

По умолчанию добавляется набор данных типа *Шаблоны элементов данных*.

| | |
|---|---|
| *Набор данных* | Для набора данных **Шаблоны элементов данных**:<br>Выберите или введите шаблоны узла сети и элемента данных; на графике будут отображаться данные элементов данных, которые соответствуют этим шаблонам; может отображаться до 50 элементов данных.<br>Для выбора можно использовать шаблоны с подстановочными знаками (например, * вернет результаты, соответствующие нулю или более символам).<br>Чтобы указать шаблон с подстановочным знаком, введите строку вручную и нажмите *Enter*.<br>Символ подстановки всегда интерпретируется, поэтому нельзя добавить, например, элемент данных с именем *item** отдельно, если есть другие совпадающие элементы данных (например, *item2*, *item3*).<br>Указание шаблонов узла сети и элемента данных обязательно для наборов данных типа "Шаблоны элементов данных".<br>См. также: Подробности настройки набора данных.<br><br>Для набора данных **Список элементов данных**:<br>Выберите элементы данных для графика, нажав кнопку *Добавить элемент данных*.<br>Также можно выбрать совместимые виджеты в качестве источника данных для элементов, нажав кнопку *Добавить виджет*.<br>Указание элементов данных или виджетов обязательно для наборов данных типа "Список элементов данных".<br>См. также: Подробности настройки набора данных.<br><br>Обратите внимание, что допускаются только числовые типы элементов данных.<br><br>При настройке виджета на шаблонной панели параметр для указания шаблонов узлов сети недоступен, а параметр для указания списка элементов данных позволяет выбрать только элементы данных, настроенные на шаблоне. |

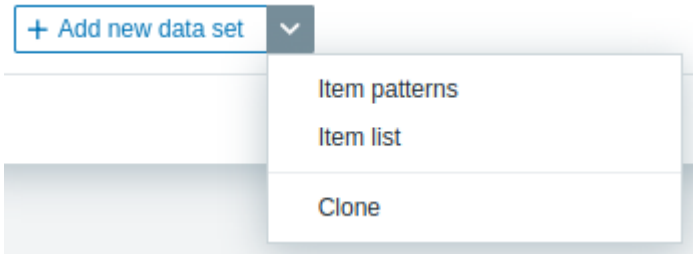
| | |
|---|---|
| *Отрисовка* | Выберите тип отрисовки метрики.<br>Возможные типы отрисовки: *Линия* (по умолчанию), *Точки*, *Лестница* и *Столбцы*.<br>Обратите внимание: если в линейном/ступенчатом графике есть только одна точка данных, она отображается как точка независимо от типа отрисовки. Размер точки вычисляется на основе ширины линии, но он не может быть меньше 3 пикселей, даже если ширина линии меньше. |
| *С накоплением* | Установите флажок, чтобы отображать данные как накопленные (с заполненными областями).<br>Этот параметр недоступен, если выбран тип отрисовки *Точки*. |
| *Ширина* | Задайте ширину линии.<br>Этот параметр доступен, если выбран тип отрисовки *Линия* или *Лестница*. |
| *Размер точки* | Задайте размер точки.<br>Этот параметр доступен, если выбран тип отрисовки *Точки*. |
| *Прозрачность* | Задайте уровень прозрачности. |
| *Заполнение* | Задайте уровень заполнения.<br>Этот параметр доступен, если выбран тип отрисовки *Линия* или *Лестница*. |
| *Отсутствующие данные* | Выберите вариант отображения отсутствующих данных:<br>**Нет** - промежуток остается пустым;<br>**Соединять** - соединяются два граничных значения;<br>**Считать как 0** - отсутствующие данные отображаются как значения 0;<br>**Последнее известное** - отсутствующие данные отображаются со значением, равным последнему известному значению; не применяется для типов отрисовки *Точки* и *Столбцы*. |
| *Переопределить узел сети* | Выберите совместимый виджет или панель как источник данных для узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Ось Y* | Выберите сторону графика, на которой будет отображаться ось Y. |
| *Сдвиг времени* | При необходимости укажите сдвиг времени.<br>В этом поле можно использовать суффиксы времени. Допускаются отрицательные значения. |
| *Функция агрегации* | Укажите, какую функцию агрегации использовать для каждого элемента данных или для всего набора данных в пределах выбранного *Интервала агрегации*:<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение;<br>**sum** - отображать сумму значений;<br>**count** - отображать количество значений;<br>**first** - отображать первое значение;<br>**last** - отображать последнее значение.<br>Если выбрано **не используется**, отображаются все значения (без агрегации).<br><br>Для *min*, *max*, *avg* и *sum* можно отображать только числовые данные. Для *count* нечисловые данные будут преобразованы в числовые.<br>См. также: Агрегация на графиках. |
| *Интервал агрегации* | Укажите интервал для агрегации значений.<br>В этом поле можно использовать суффиксы времени. Числовое значение без суффикса будет считаться секундами.<br><br>Обратите внимание: если виджет настроен на отображение исторических данных на основе трендов (*Выбор исторических данных* установлен в *Тренды* или *Авто*), рекомендуется использовать интервал агрегации, кратный 1 часу (например, 3600, 60m, 1h, 3h и т. д.). Тренды хранят почасовые агрегированные значения, поэтому использование интервала агрегации, не кратного 1 часу (например, 100s, 7min, 15min, 90min и т. д.), может привести к результатам, которые трудно интерпретировать. |
| *Агрегировать* | Укажите, что агрегировать:<br>**Каждый элемент данных** - каждый элемент данных в наборе данных будет агрегирован и отображен отдельно;<br>**Набор данных** - все элементы данных набора будут агрегированы и отображены как одно значение. |

| | |
|---|---|
| *Аппроксимация* | Укажите, какое значение отображать, если на один вертикальный пиксель графика приходится более одного значения:<br>**all** - отображать наименьшее, наибольшее и среднее значения;<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение.<br><br>Этот параметр полезен при отображении графика за большой период времени с частым интервалом обновления (например, за год значений, собираемых каждые 10 минут). |
| *Метка набора данных* | Укажите метку набора данных, которая отображается в настройке графика *Набор данных* и в *Легенде* графика (для агрегированных наборов данных).<br>Все наборы данных нумеруются, включая те, для которых указана *Метка набора данных*. Если метка не указана, набор данных будет автоматически помечен по своему номеру (например, ”Набор данных #2”, ”Набор данных #3” и т. д.). Нумерация наборов данных пересчитывается после изменения порядка/перетаскивания наборов данных.<br>Слишком длинные метки наборов данных будут сокращены, чтобы поместиться в месте отображения (например, ”Number of proc...”). |

Подробности настройки набора данных

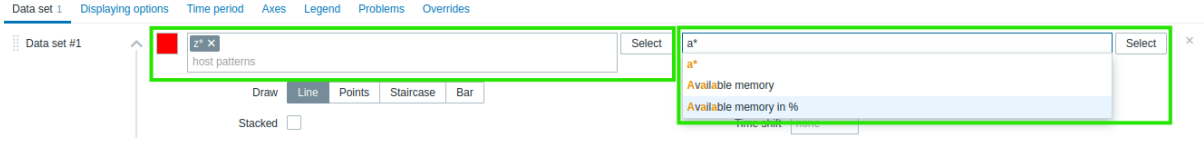
Существующие наборы данных отображаются в списке. Вы можете:

- Щелкнуть значок перемещения и перетащить набор данных в новое место в списке.
- Щелкнуть значок разворачивания, чтобы развернуть подробности набора данных. После разворачивания этот значок превращается в значок сворачивания.
- Щелкнуть значок цвета, чтобы открыть палитру. Вы можете ввести шестнадцатеричный код, выбрать сплошной образец цвета или перейти на вкладку *Palette* и выбрать строку предопределенных цветов. Выбранный цвет применяется напрямую для наборов данных *Item list* или как основа для автоматически создаваемых оттенков в *Item patterns*. Используйте Tab для перехода между элементами диалога, клавиши со стрелками для навигации по образцам или строкам палитры, Enter для выбора и Esc для отмены.
- Щелкнуть кнопку *Add new data set*, чтобы добавить пустой набор данных, позволяющий выбрать шаблоны узлов сети и элементов данных. Если щелкнуть значок со стрелкой вниз рядом с кнопкой *Add new data set*, появится выпадающее меню, позволяющее добавить новый набор данных *Item patterns* или *Item list*, либо *Clone* текущий открытый набор данных. Если все наборы данных свернуты, параметр *Clone* недоступен.

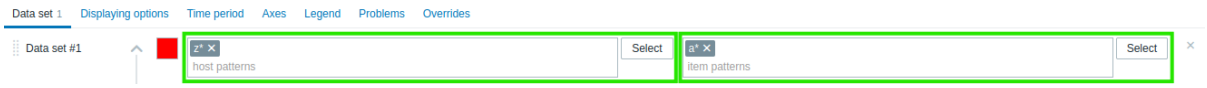



Набор данных **Item patterns** содержит поля *Host patterns* и *Item patterns*, которые распознают как полные имена, так и шаблоны, содержащие символ подстановки (*). Эта функция позволяет выбрать все имена узлов сети и имена элементов данных, содержащие выбранный шаблон. При вводе имени элемента данных или шаблона элемента данных в поле *Item patterns* в раскрывающемся списке отображаются только элементы данных, относящиеся к выбранному имени(ам) узла сети.

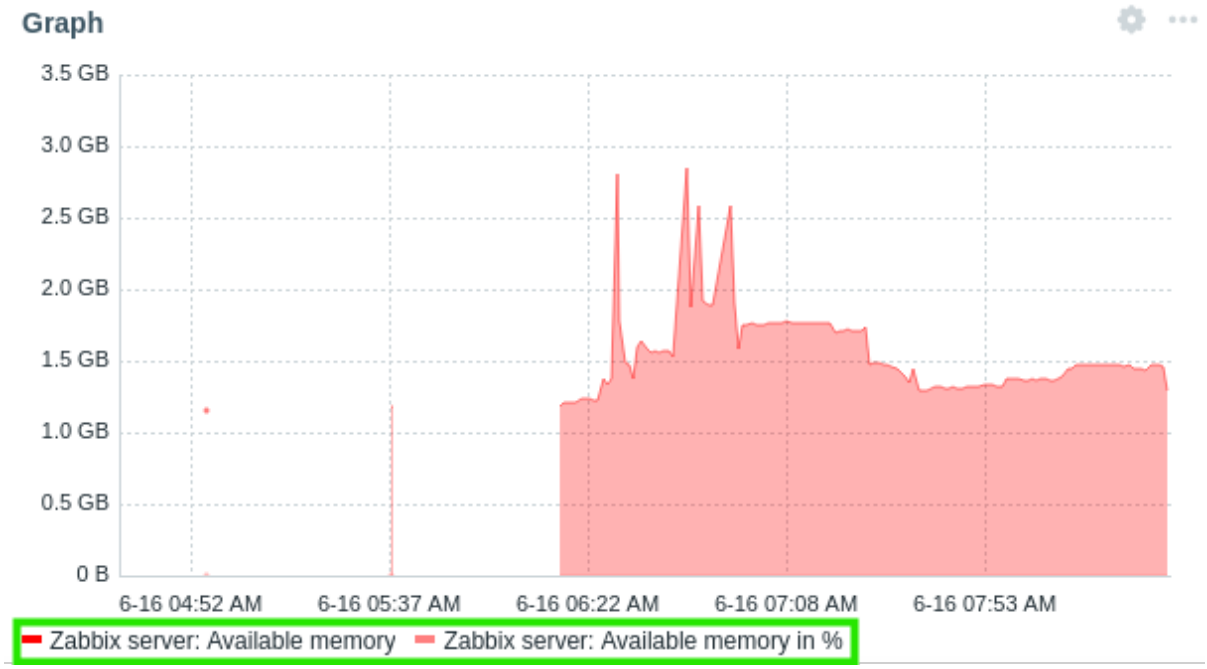
Например, если в поле *Host patterns* введен шаблон **z***, в раскрывающемся списке отображаются все имена узлов сети, содержащие этот шаблон: **z*, Zabbix server, Zabbix proxy**. После нажатия *Enter* этот шаблон принимается и отображается как **z***. Аналогично, если в поле *Item patterns* введен шаблон **a***, в раскрывающемся списке отображаются все имена элементов данных, содержащие этот шаблон: **a*, Available memory, Available memory in %**.



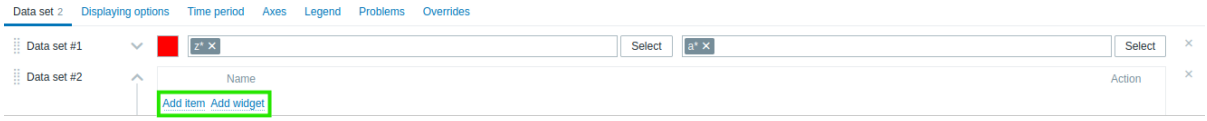

После нажатия *Enter* шаблон принимается и отображается как **a***.



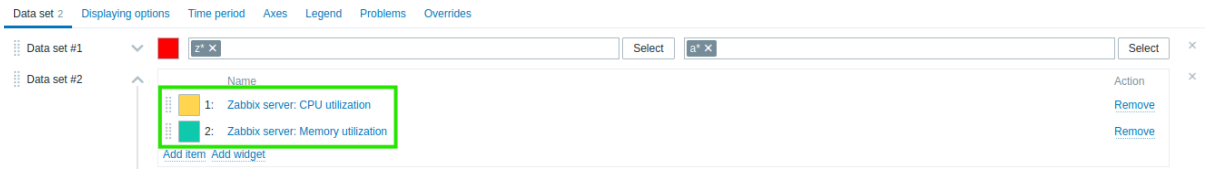

Затем график отображает все элементы данных, относящиеся к выбранному имени(ам) узла сети.



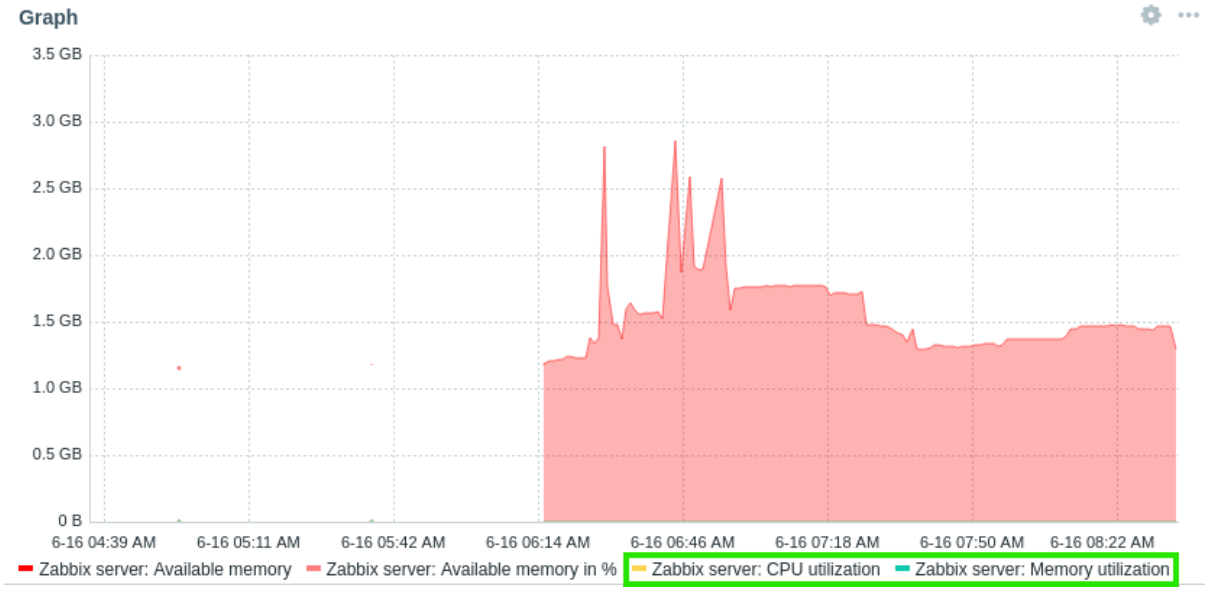

Набор данных **Item list** содержит кнопку *Add item*, которая позволяет добавлять элементы данных для отображения на графике. Вы также можете добавить совместимые виджеты в качестве источника данных для элементов данных, нажав кнопку *Add widget*.




Например, при нажатии кнопки *Add item* открывается всплывающее окно, содержащее параметр *Host*. После выбора узла сети в списке отображаются все его элементы данных, доступные для выбора.




После выбора одного или нескольких элементов данных они будут отображаться в списке элементов данных набора данных и на графике.

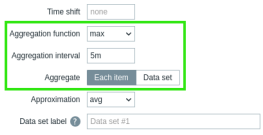



Агрегация на графиках

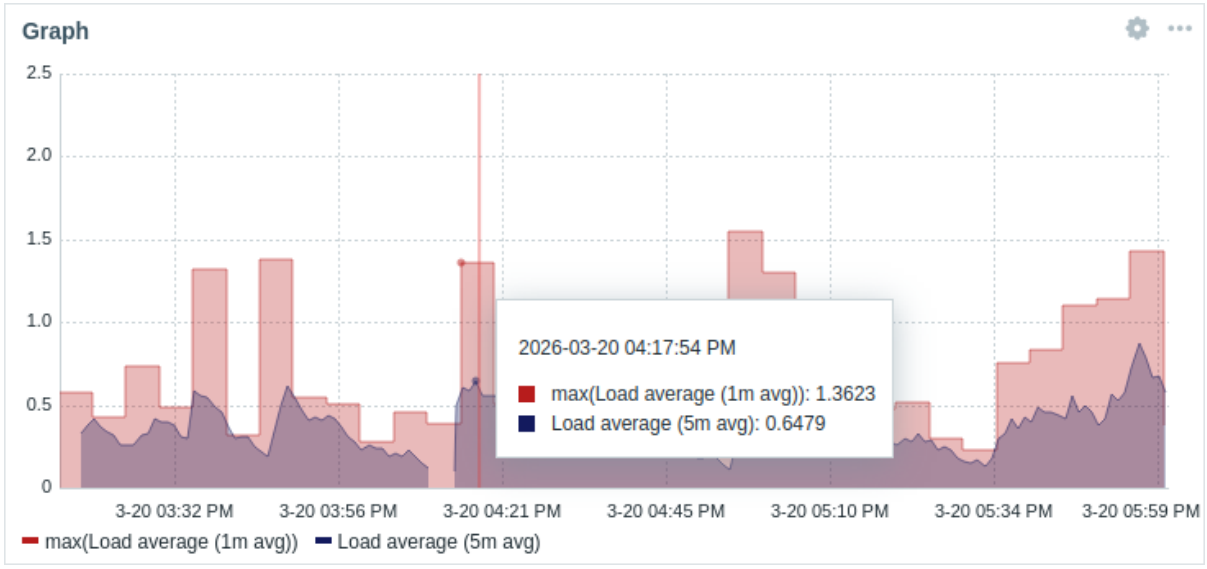
Агрегация позволяет отображать агрегированное значение (минимум, максимум, среднее и т. д.) за указанный временной интервал (5 минут, час, день и т. д.) вместо всех отдельных значений. Внутри набора данных агрегация может применяться либо к каждому элементу данных отдельно, либо ко всему набору данных.

Некоторые варианты использования функции агрегации могут быть следующими: - Среднее количество запросов к серверу Nginx в день. - Наименьший объем дискового пространства среди кластеров за неделю.

В следующем примере к элементу данных применяется функция агрегации ”max” с интервалом агрегации в пять минут:



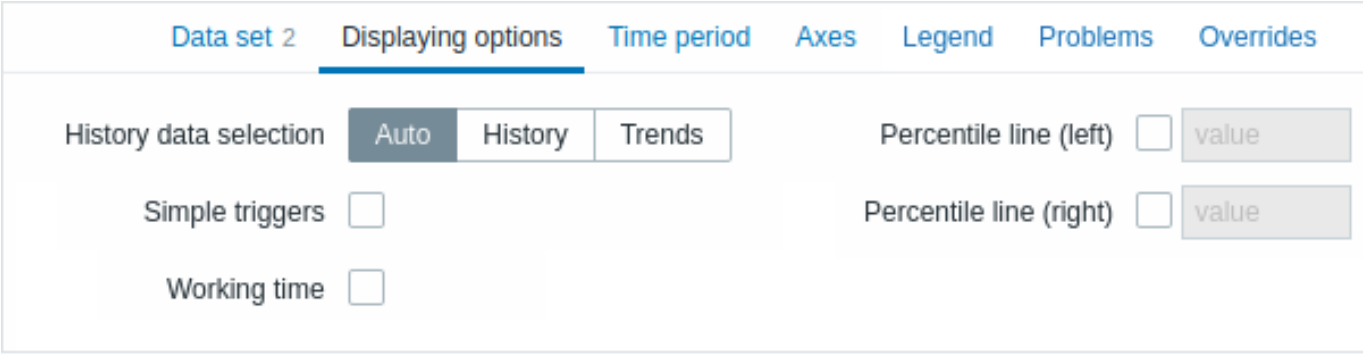

В примере показана максимальная загрузка CPU за пять минут (агрегированная из средних значений за одну минуту) по сравнению со средней загрузкой CPU за пять минут.




Если для легенды выбрано *Show aggregation function*, элементы данных отображаются в скобках, а в легенде и всплывающей подсказке перед ними указывается функция агрегации. Если для графика задан *Data set label*, метка отображается в скобках, а перед ней указывается функция агрегации.

Отображение параметров

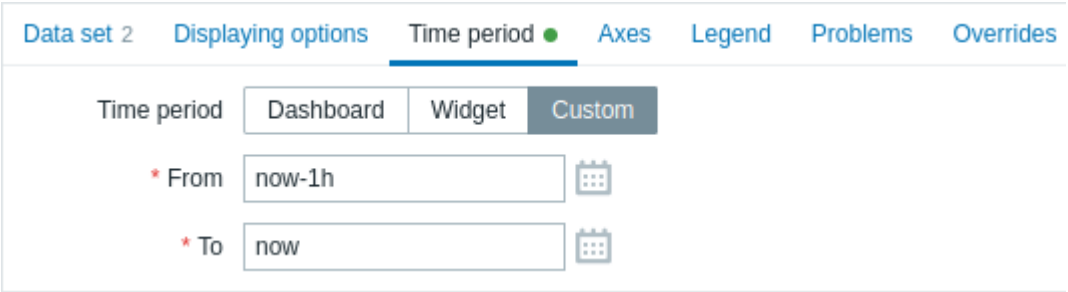
Вкладка **Отображение параметров** позволяет задать выбор исторических данных:




| | |
|---|---|
| *Выбор исторических данных* | Задайте источник данных графика:<br>**Auto** - данные выбираются в соответствии с классическим алгоритмом построения графика (по умолчанию);<br>**History** - данные из history;<br>**Trends** - данные из trends. |
| *Простые триггеры* | Установите флажок, чтобы показывать пороги триггеров для простых триггеров. Пороги будут отображаться пунктирными линиями с использованием цвета уровня серьезности триггера.<br>Простой триггер - это триггер с одной функцией (только `last`, `max`, `min`, `avg`) для одного элемента данных в выражении.<br>Можно отобразить не более трех триггеров. Обратите внимание, что триггер должен находиться в пределах отображаемого диапазона, чтобы быть видимым. |
| *Рабочее время* | Установите флажок, чтобы показывать рабочее время на графике. Рабочее время (рабочие дни) отображается на графиках белым фоном, а нерабочее время - серым (в теме веб-интерфейса по умолчанию *Original blue*). |
| *Линия перцентиля (левая)* | Установите флажок и введите значение перцентиля, чтобы показать указанный перцентиль в виде линии на левой оси Y графика.<br>Например, если задан перцентиль 95%, линия перцентиля будет находиться на уровне, ниже которого располагаются 95 процентов значений. |
| *Линия перцентиля (правая)* | Установите флажок и введите значение перцентиля, чтобы показать указанный перцентиль в виде линии на правой оси Y графика.<br>Например, если задан перцентиль 95%, линия перцентиля будет находиться на уровне, ниже которого располагаются 95 процентов значений. |

Период времени

Вкладка **Период времени** позволяет задать период времени, за который будут отображаться данные на графике:

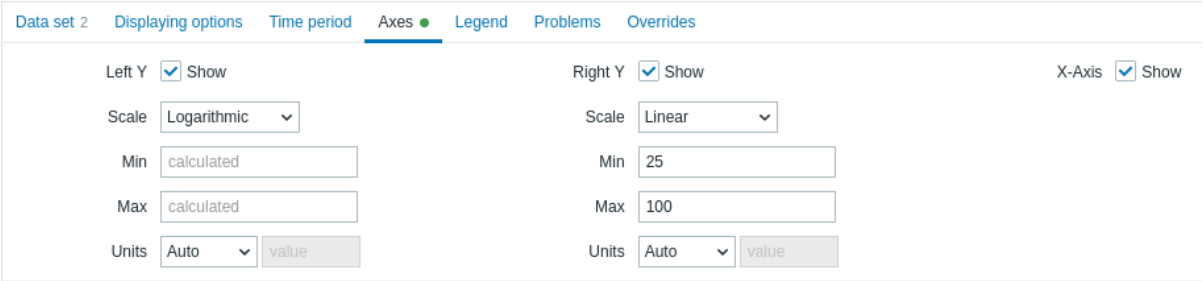



| | |
|---|---|
| *Период времени* | Выберите источник данных для периода времени:<br>**Dashboard** - задает селектор *Период времени* в качестве источника данных;<br>**Widget** - задает совместимый виджет, указанный в параметре *Widget*, в качестве источника данных;<br>**Custom** - задает период времени, указанный в параметрах *From* и *To*, в качестве источника данных; если задано, в правом верхнем углу виджета будет отображаться значок часов, показывающий установленное время при наведении курсора. |
| *Widget* | Введите или выберите совместимый виджет в качестве источника данных для периода времени.<br>Этот параметр доступен, если для *Период времени* установлено значение "Widget". |

| | |
|---|---|
| *From* | Введите или выберите начало периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w` и т. д.).<br>Этот параметр доступен, если для *Период времени* установлено значение "Custom". |
| *To* | Введите или выберите конец периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w` и т. д.).<br>Этот параметр доступен, если для *Период времени* установлено значение "Custom". |

Оси

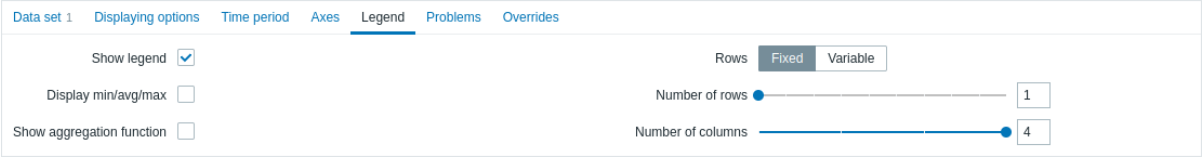
Вкладка **Оси** позволяет настроить отображение осей:




| | |
|---|---|
| *Left Y* | Отметьте этот флажок, чтобы сделать левую ось Y видимой.<br>Флажок может быть недоступен, если он не выбран либо на вкладке *Data set*, либо на вкладке *Overrides*. |
| *Right Y* | Отметьте этот флажок, чтобы сделать правую ось Y видимой.<br>Флажок может быть недоступен, если он не выбран либо на вкладке *Data set*, либо на вкладке *Overrides*. |
| *X-Axis* | Снимите этот флажок, чтобы скрыть ось X (по умолчанию он установлен). |
| *Scale* | Выберите масштаб значений оси графика в раскрывающемся списке:<br>**Linear** - значения оси увеличиваются на фиксированную величину (например, 10, 20, 30), что подходит для данных, которые изменяются равномерно или охватывают небольшой либо умеренный диапазон;<br>**Logarithmic** - значения оси увеличиваются экспоненциально (например, 10, 100, 1000), что подходит для данных, которые изменяются быстро или охватывают большой диапазон. |
| *Min* | Задайте минимальное значение соответствующей оси.<br>Указывается минимальное значение видимого диапазона оси Y. |
| *Max* | Задайте максимальное значение соответствующей оси.<br>Указывается максимальное значение видимого диапазона оси Y. |
| *Units* | Выберите единицу измерения значений оси графика в раскрывающемся списке:<br>**Auto** - значения оси отображаются с единицей измерения первого элемента данных в наборе;<br>**Static** - значения оси отображаются с единицей измерения, указанной в поле ввода *value*; если поле оставить пустым, отображаются только числовые значения. |

Легенда

Вкладка **Легенда (Legend)** позволяет настроить легенду графика:

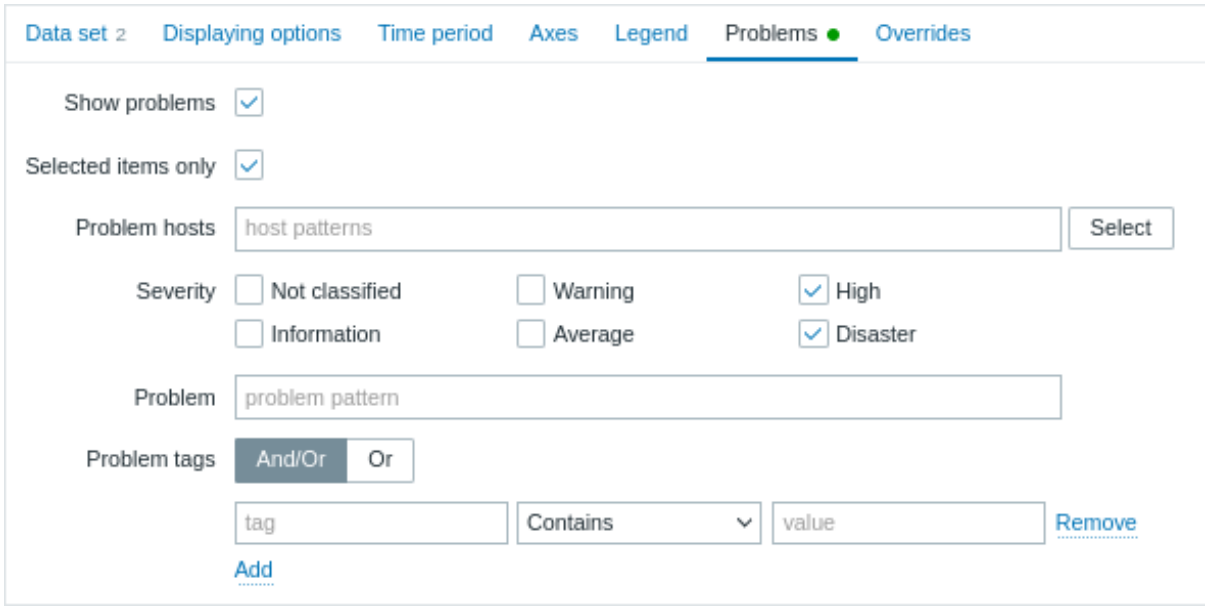



| | |
|---|---|
| *Легенда (Show legend)* | Снимите этот флажок, чтобы скрыть легенду на графике (по умолчанию флажок отмечен). |
| *Отображение мин/сред/макс (Display min/avg/max)* | Установите этот флажок для отображения в легенде минимального, среднего и максимального значений элемента данных. |

| | |
|---|---|
| *Отображение функции агрегирования (Show aggregation function)* | Установите этот флажок для отображения в легенде функции агрегирования. |
| *Количество строк (Rows)* | Выберите режим отображения строк легенды:<br>**Фиксированное (Fixed)** — количество отображаемых строк определяется значением параметра *Количество строк (Number of rows)*;<br>**Переменное (Variable)** — количество отображаемых строк определяется количеством настроенных элементов данных, в то же время не превышая значения параметра *Максимальное количество строк (Maximum number of rows)*. |
| *Количество строк/ Максимальное количество строк (Number of rows/ Maximum number of rows)* | Если *Количество строк (Rows)* имеет значение «Фиксированное (Fixed)», задать количество отображаемых строк легенды (1-10).<br>Если *Количество строк (Rows)* имеет значение «Переменное (Variable)», задать максимальное количество отображаемых строк легенды (1-10). |
| *Количество колонок (Number of columns)* | Задать количество отображаемых столбцов легенды (1-4).<br>Этот параметр доступен, если не отмечено *Отображение мин/сред/макс (Display min/avg/max)*. |

Проблемы

Вкладка **Проблемы (Problems)** позволяет настроить отображение проблем:

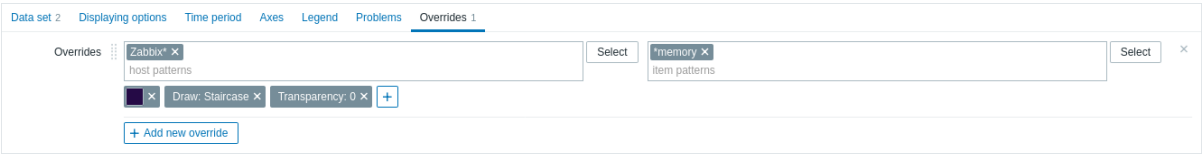



| | |
|---|---|
| *Проблемы (Show problems)* | Установите этот флажок, чтобы включить отображение проблем на графике (по умолчанию флажок снят, т.е. отключено). |
| *Только выбранные элементы данных (Selected items only)* | Установите этот флажок, чтобы включить отображение на графике только проблем по выбранным элементам данных. |
| *Узлы сети с проблемами (Problem hosts)* | Выберите узлы сети с проблемами для отображения на графике.<br>Можно использовать шаблоны с подстановочными символами (например, * вернёт результаты, соответствующие нулю или более символов).<br>Чтобы указать шаблон с подстановочными символами, просто введите строку вручную и нажмите *Enter*.<br>В процессе ввода обратите внимание, как все соответствующие узлы сети отображаются в выпадающем списке.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Важность (Severity)* | Отметьте важности проблем, чтобы отфильтровать те из них, которые будут отображаться на графике.<br>Если никакие важности не отмечены, будут отображены все проблемы. |

| | |
|---|---|
| *Проблема (Problem)* | Укажите название проблемы для отображения на графике. |
| *Теги проблем (Problem tags)* | Укажите теги проблем, чтобы ограничить количество проблем, отображаемых в виджете. Можно как включать, так и исключать определённые теги и их значения. Можно указать несколько условий. Поиск соответствия имён тегов всегда чувствителен к регистру.<br><br>По каждому условию доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием *Или (Or)*;<br>**Или (Or)** — достаточно соблюдения одного условия. |

Переопределения

Вкладка **Переопределения** позволяет добавлять пользовательские переопределения для наборов данных:




Переопределения полезны, когда для набора данных выбрано несколько элементов данных с использованием подстановочного знака *, и вы хотите изменить способ отображения элементов по умолчанию (например, базовый цвет по умолчанию или любое другое свойство).

Существующие переопределения, если они есть, отображаются в списке. Чтобы добавить новое переопределение:

- Нажмите кнопку

- Выберите узлы сети и элементы данных для переопределения. Кроме того, вы можете ввести шаблоны узлов сети и элементов данных. Можно использовать шаблоны с подстановочными знаками (например, * вернет результаты, соответствующие нулю или более символам). Чтобы указать шаблон с подстановочным знаком, просто введите строку вручную и нажмите *Enter*. По мере ввода обратите внимание, как все совпадающие узлы сети отображаются в раскрывающемся списке. Символ подстановки всегда интерпретируется, поэтому невозможно добавить, например, элемент данных с именем "item*" отдельно, если есть другие совпадающие элементы (например, item2, item3). Параметры шаблона узла сети и шаблона элемента данных обязательны. Параметр для указания шаблонов узлов сети недоступен при настройке виджета на шаблонной панели. Параметр для указания списка элементов данных позволяет выбирать только элементы данных, настроенные в шаблоне при настройке виджета на шаблонной панели.
- Нажмите , чтобы выбрать параметры переопределения. Следует выбрать как минимум один параметр переопределения. Описания параметров см. на вкладке *Набор данных* выше.

9 График (classic)

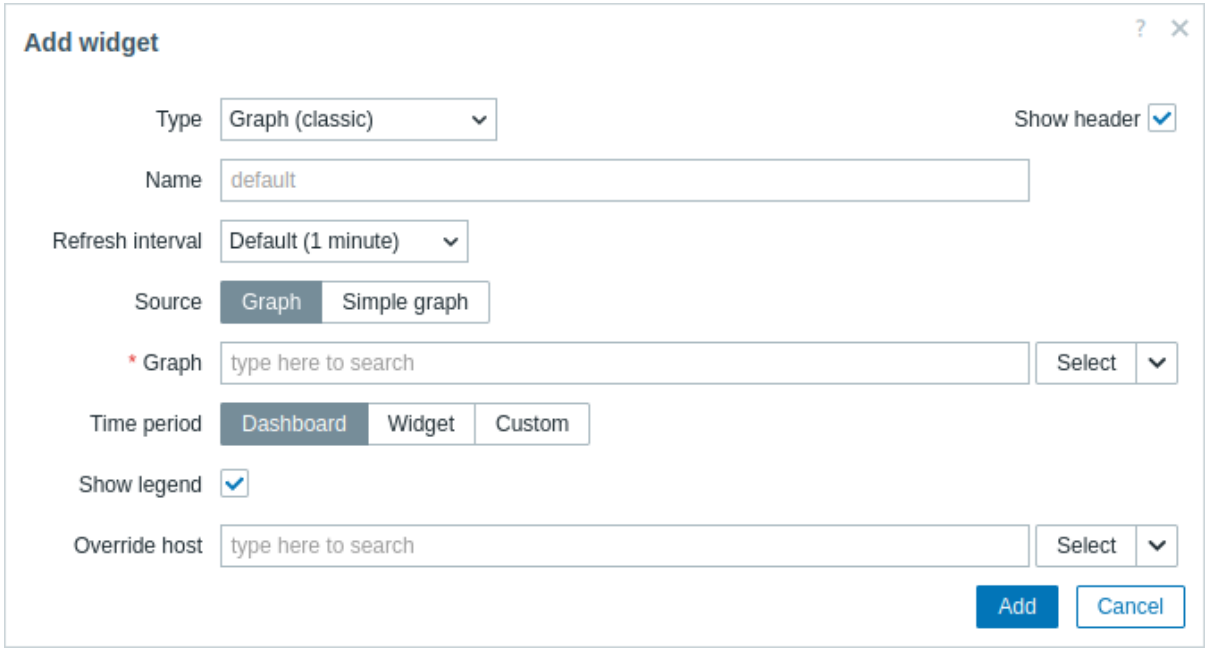
Обзор

В виджете классического графика можно отображать один пользовательский график или простой график.

Информацию, отображаемую в виджете *Graph (classic)*, можно загрузить в виде изображения PNG, выбрав параметр *Download image* в меню виджета.

Конфигурация

Чтобы настроить, выберите *Graph (classic)* в качестве типа:




Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Source* | Выберите тип графика:<br>**Graph** - custom graph;<br>**Simple graph** - simple graph. |
| *Graph* | Выберите пользовательский график для отображения.<br>Кроме того, можно выбрать совместимый виджет в качестве источника данных для графиков.<br>Этот параметр доступен, если для *Source* задано значение ”Graph”. |
| *Item* | Выберите элемент данных для отображения в простом графике.<br>Кроме того, можно выбрать совместимый виджет в качестве источника данных для элементов данных.<br>Этот параметр доступен, если для *Source* задано значение ”Simple graph”. |
| *Time period* | Задайте временной период, за который нужно отображать данные на графике. Выберите источник данных для временного периода:<br>**Dashboard** - используйте селектор *Time period* в качестве источника данных;<br>**Widget** - используйте совместимый виджет, указанный в параметре *Widget*, в качестве источника данных;<br>**Custom** - используйте временной период, указанный в параметрах *From* и *To*, в качестве источника данных; если задано, в правом верхнем углу виджета будет отображаться значок часов, показывающий заданное время при наведении курсора. |
| *Widget* | Введите или выберите совместимый виджет в качестве источника данных для временного периода.<br>Этот параметр доступен, если для *Time period* задано значение ”Widget”. |
| *From* | Введите или выберите начало временного периода.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* задано значение ”Custom”. |
| *To* | Введите или выберите конец временного периода.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* задано значение ”Custom”. |
| *Show legend* | Снимите этот флажок, чтобы скрыть легенду на графике (по умолчанию установлен). |
| *Override host* | Выберите совместимый виджет или веб-интерфейс в качестве источника данных для узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |

Легенда графика состоит из трех разделов:

- Элементы данных и их агрегированные значения
- Процентили (если настроены)
- Триггеры (если какие-либо из них связаны с отображаемыми элементами данных)

Если высоты графика в пределах виджета недостаточно, легенда может не отображаться или отображаться только частично. Сначала скрываются триггеры и процентили, затем легенда элементов данных. Чтобы отобразить полную легенду, увеличьте вертикальный размер виджета.

**Note:**
Можно отобразить не более 3 строк триггеров. Если триггеров больше, то для отображения приоритет отдается триггерам с меньшей серьезностью.

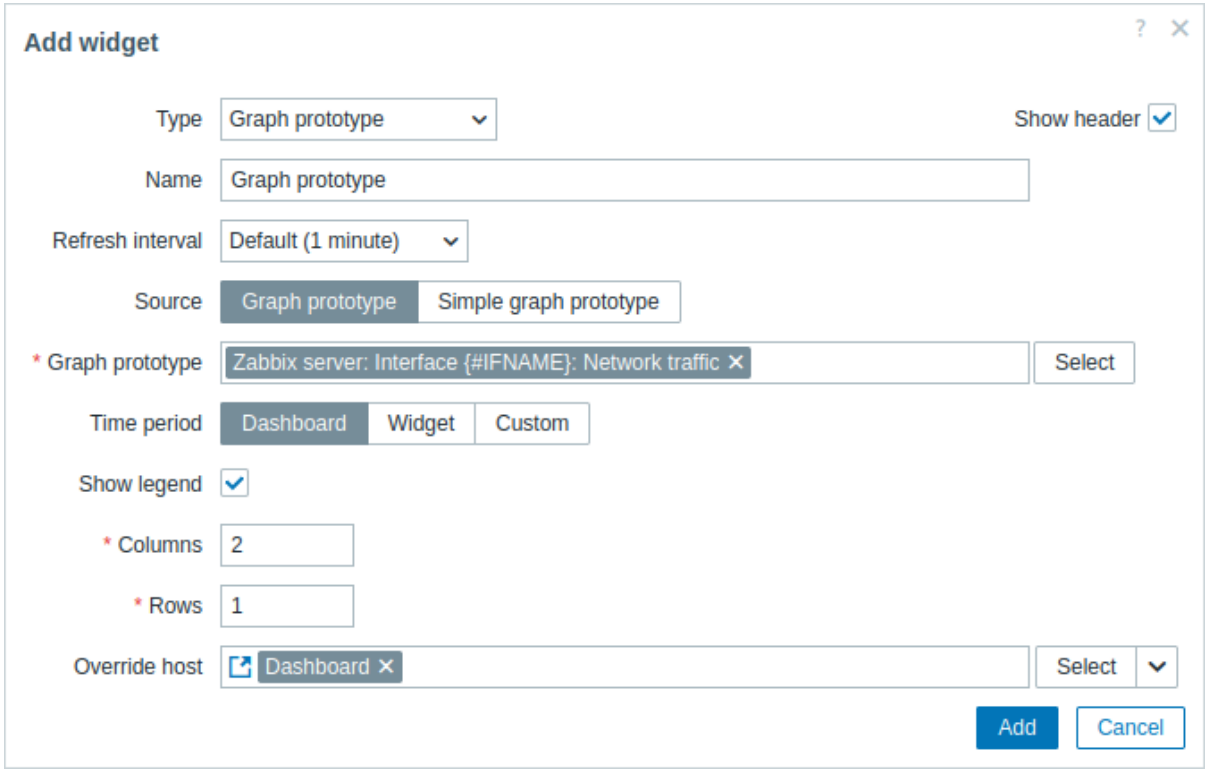
10 Прототип графика

Обзор

В виджете прототипа графиков вы можете отобразить решётку графиков, созданных правилом низкоуровневого обнаружения либо из прототипа графиков, либо из прототипа элементов данных.

Конфигурация

Чтобы настроить, выберите *Graph prototype* в качестве типа виджета:




Помимо параметров, общих для всех виджетов, можно задать следующие специфические параметры:

| | |
|---|---|
| *Source* | Выберите источник графиков: **Graph prototype** или **Simple graph prototype**. |
| *Graph prototype* | Выберите прототип графика для отображения графиков, обнаруженных прототипом графика.<br>Этот параметр доступен, если для *Source* задано значение "Graph prototype". |
| *Item prototype* | Выберите прототип элемента данных для отображения простых графиков для элементов данных, обнаруженных прототипом элемента данных.<br>Этот параметр доступен, если для *Source* задано значение "Simple graph prototype". |
| *Time period* | Задайте период времени, за который нужно отображать данные на графиках. Выберите источник данных для периода времени:<br>**Dashboard** - используйте селектор *Time period* в качестве источника данных;<br>**Widget** - используйте совместимый виджет, указанный в параметре *Widget*, в качестве источника данных;<br>**Custom** - используйте период времени, указанный в параметрах *From* и *To*, в качестве источника данных; если задано, в правом верхнем углу виджета будет отображаться значок часов, показывающий заданное время при наведении курсора мыши. |

| | |
|---|---|
| *Widget* | Введите или выберите совместимый виджет в качестве источника данных для периода времени.<br>Этот параметр доступен, если для *Time period* задано значение "Widget". |
| *From* | Введите или выберите начало периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w` и т. д.).<br>Этот параметр доступен, если для *Time period* задано значение "Custom". |
| *To* | Введите или выберите конец периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w` и т. д.).<br>Этот параметр доступен, если для *Time period* задано значение "Custom". |
| *Show legend* | Снимите этот флажок, чтобы скрыть легенду на графиках (по умолчанию установлен). |
| *Override host* | Выберите совместимый виджет или веб-интерфейс в качестве источника данных для узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Columns* | Введите количество столбцов графиков, которые нужно отображать в виджете прототипа графика. |
| *Rows* | Введите количество строк графиков, которые нужно отображать в виджете прототипа графика. |

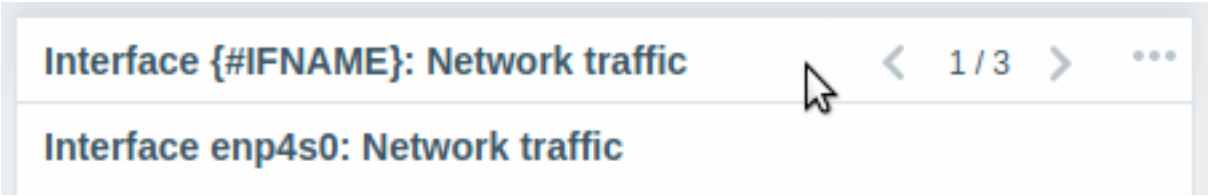
Легенда графика состоит из трех разделов:

- Элементы данных и их агрегированные значения
- Процентили (если настроены)
- Триггеры (если какие-либо из них связаны с отображаемыми элементами данных)

Если высоты графика в виджете недостаточно, легенда может не отображаться или отображаться только частично. Сначала скрываются триггеры и процентили, затем легенда элементов данных. Чтобы отобразить полную легенду, увеличьте вертикальный размер виджета.

**Note:**
Можно отобразить не более 3 линий триггеров. Если триггеров больше, то для отображения приоритет получают триггеры с меньшей важностью.

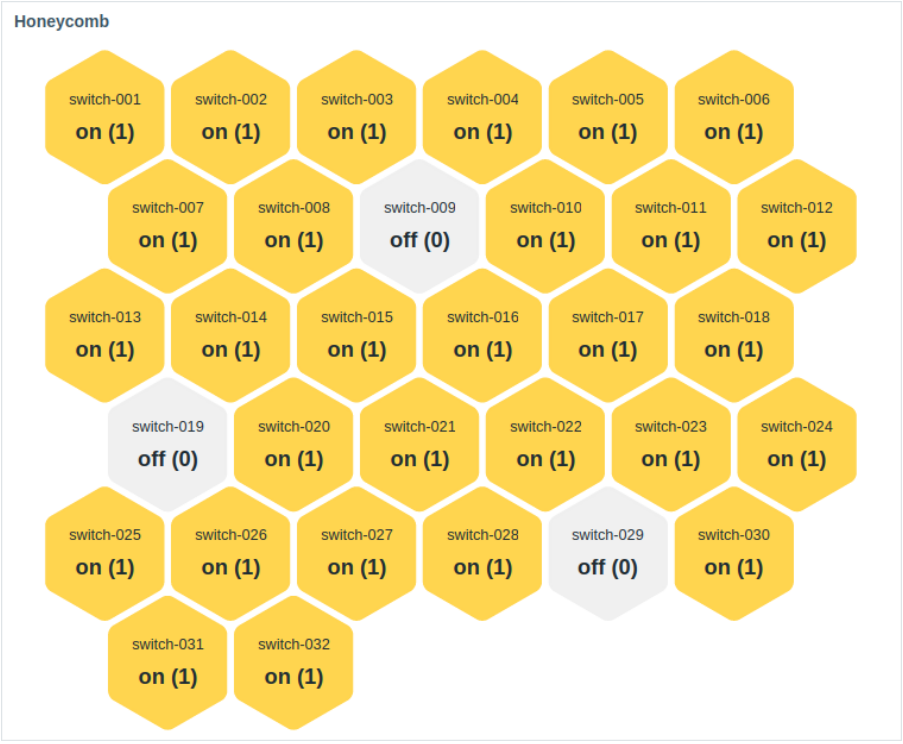
Хотя параметры *Columns* и *Rows* позволяют разместить в виджете более одного графика, графиков, обнаруженных системой, все равно может быть больше, чем столбцов/строк в виджете. В этом случае в виджете становится доступна пагинация, а выдвижной заголовок позволяет переключаться между страницами с помощью стрелок влево и вправо:



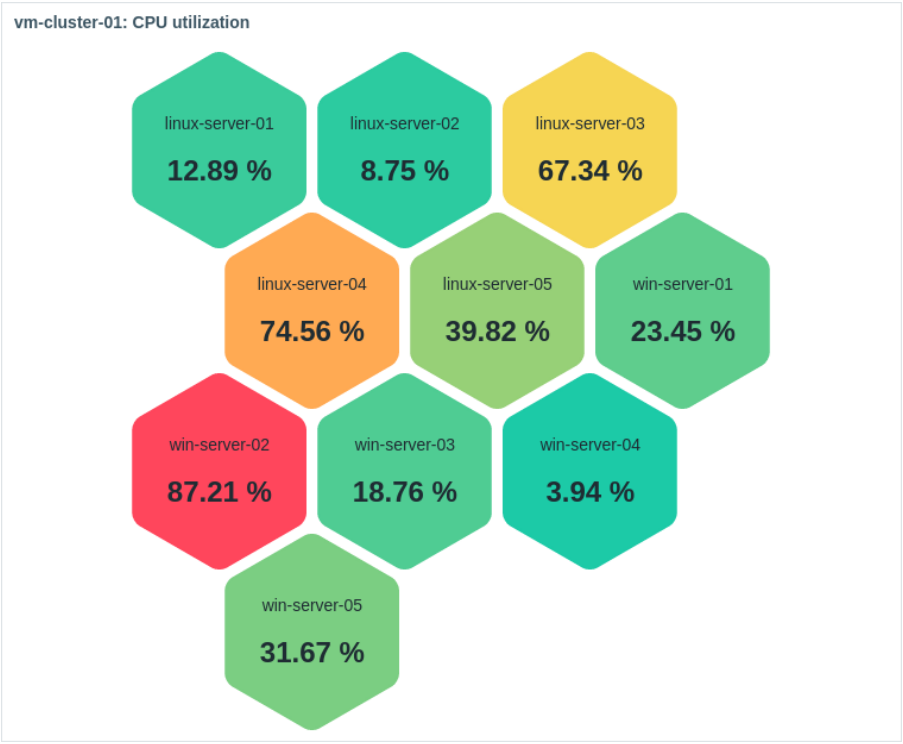

11 Honeycomb

Обзор

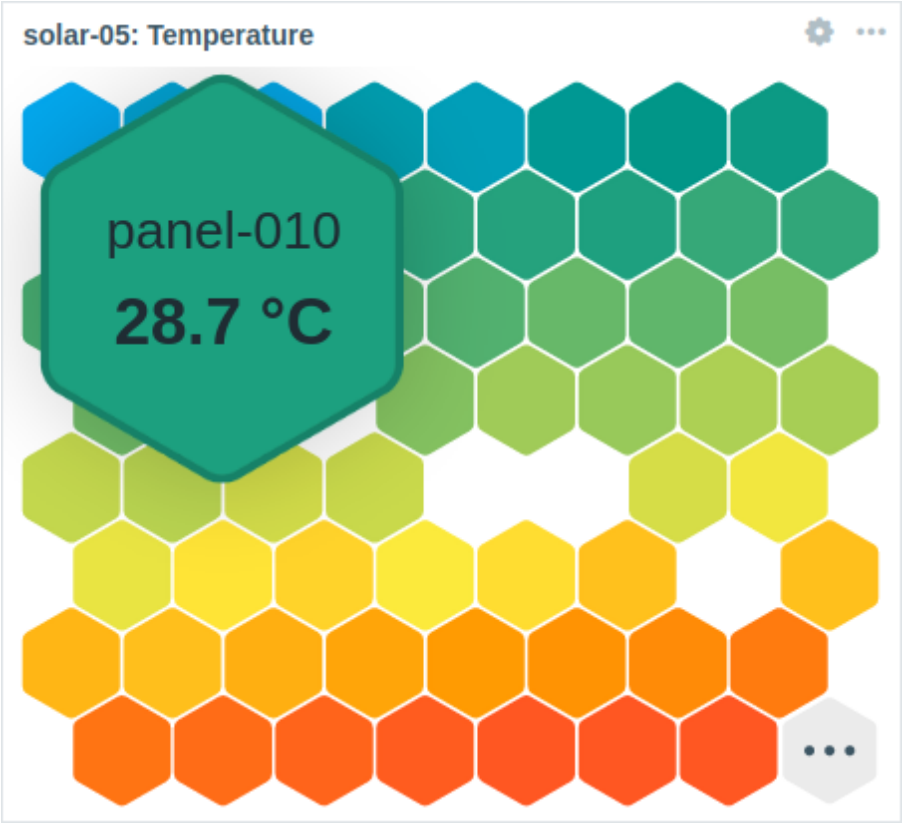
Виджет honeycomb предоставляет динамический обзор контролируемой сетевой инфраструктуры и ресурсов, где группы узлов сети, такие как виртуальные машины и сетевые устройства, вместе с соответствующими им элементами данных, визуально представлены в виде интерактивных шестиугольных ячеек.




Виджет можно визуально тонко настраивать с помощью параметров расширенной настройки, чтобы создавать широкий спектр визуальных стилей.




При наведении указателя мыши выделенная ячейка honeycomb увеличивается для лучшей видимости. При щелчке по ячейке ее граница подсвечивается, пока не будет выбрана другая ячейка.



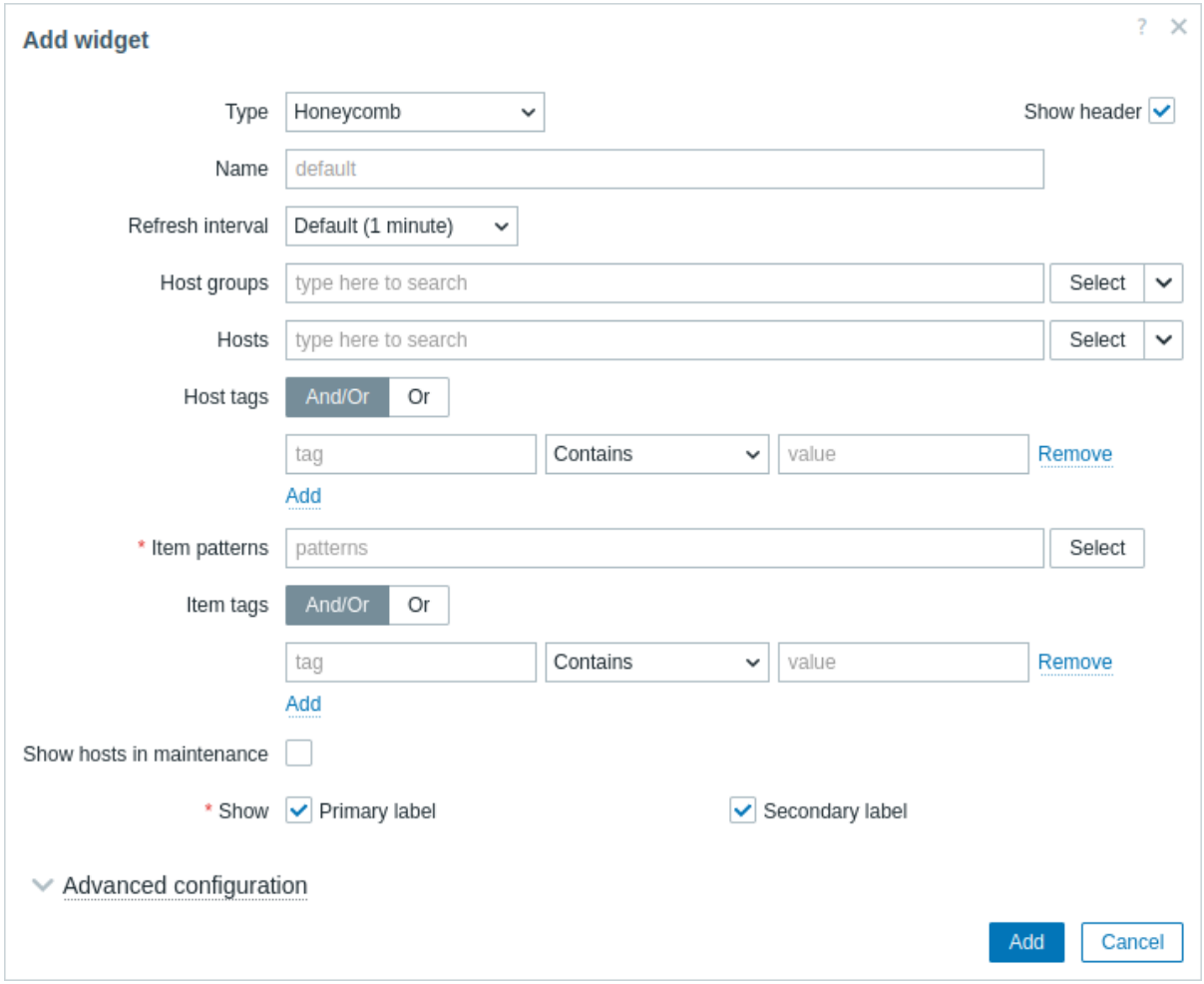

Количество отображаемых ячеек honeycomb ограничено размером виджета и минимальным размером ячейки (32 пикселя). Если все ячейки не помещаются в виджет, в качестве последней ячейки отображается многоточие.

Размер виджета можно изменять, чтобы вместить больше ячеек. При изменении размера размер и расположение ячеек honeycomb динамически корректируются. Каждая строка в honeycomb будет содержать одинаковое количество ячеек, за исключением последней строки, если общее количество ячеек не делится на количество ячеек в строке.

Информацию, отображаемую в виджете *Honeycomb*, можно скачать в виде изображения PNG, выбрав опцию *Download image* в меню виджета.

Конфигурация

Чтобы настроить, выберите *Honeycomb* в качестве типа:



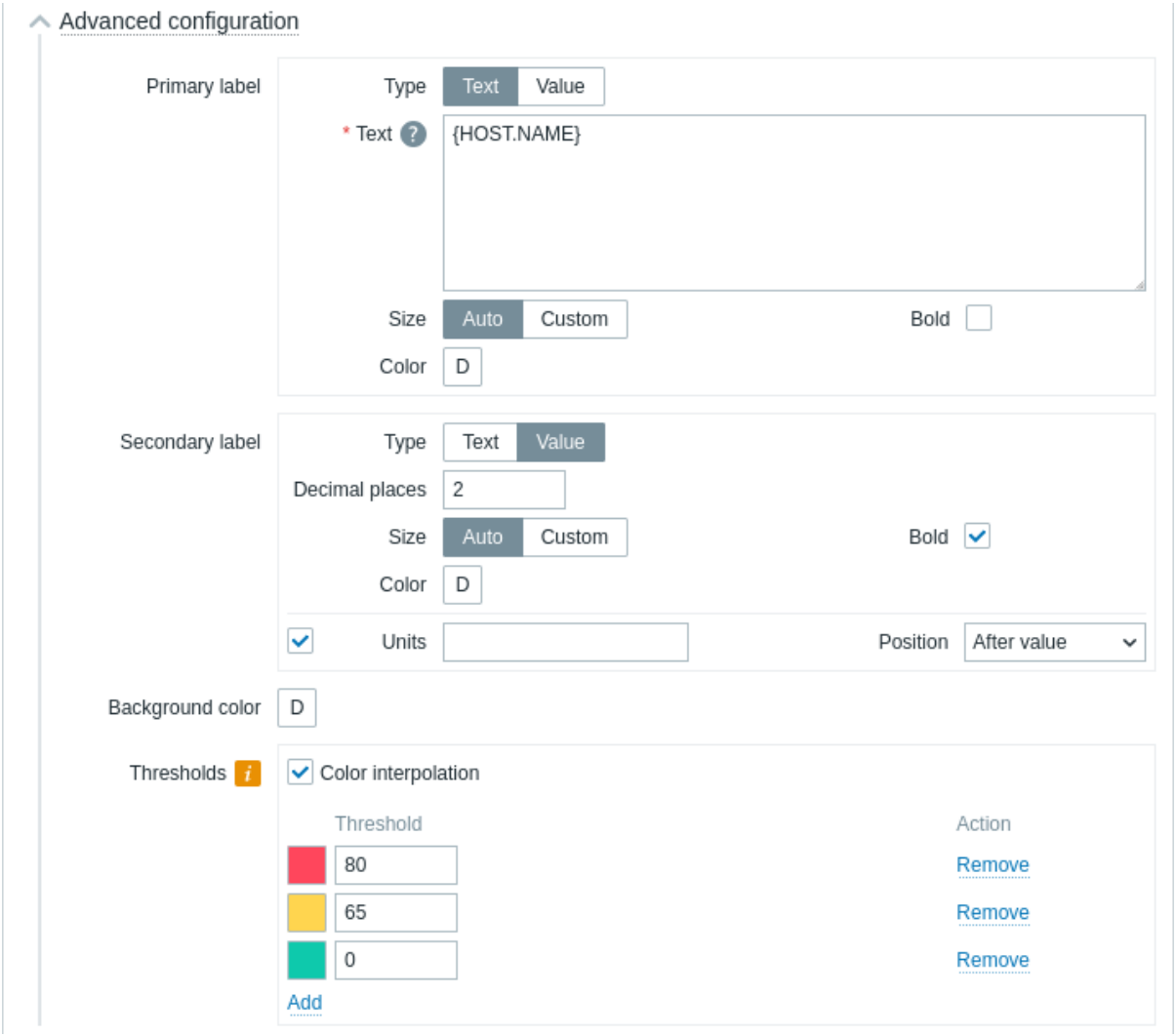

Помимо параметров, общих для всех виджетов, можно задать следующие специфические параметры:

| | |
|---|---|
| *Группы узлов сети* | Выберите группы узлов сети.<br>Кроме того, выберите совместимый виджет в качестве источника данных для групп узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет показан выпадающий список подходящих групп.<br>При выборе родительской группы узлов сети автоматически выбираются все вложенные группы узлов сети; если группы узлов сети не выбраны, виджет отобразит все группы узлов сети, содержащие узлы сети с элементами данных, соответствующими выбранному *Шаблону элемента данных* (см. ниже).<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Узлы сети* | Выберите узлы сети.<br>Кроме того, выберите совместимый виджет или панель в качестве источника данных для узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет показан выпадающий список подходящих узлов сети.<br>Если узлы сети не выбраны, виджет отобразит все узлы сети с элементами данных, соответствующими выбранному *Шаблону элемента данных* (см. ниже).<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |

| | |
|---|---|
| *Теги узлов сети* | Укажите теги, чтобы ограничить число узлов сети, отображаемых в виджете.<br>Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий.<br>Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия.<br><br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Шаблоны элементов данных* | Введите шаблоны элементов данных или выберите существующие элементы данных в качестве шаблонов элементов данных. На honeycomb будут отображаться данные элементов данных, соответствующих введенным или выбранным шаблонам. Параметр *Шаблоны элементов данных* обязателен.<br><br>Для выбора можно использовать шаблоны с подстановочными знаками (например, `*` вернет элементы данных, соответствующие нулю или более символам; `Zabbix*` вернет элементы данных, начинающиеся с ”Zabbix”).<br>Чтобы указать шаблон с подстановочным знаком, введите строку вручную и нажмите *Enter*. При вводе будет показан выпадающий список подходящих элементов данных, ограниченный теми, которые принадлежат выбранным *Узлам сети* или узлам сети из выбранных *Групп узлов сети*, если они указаны. Символ подстановки всегда интерпретируется, поэтому невозможно добавить, например, элемент данных с именем *item** отдельно, если есть другие подходящие элементы данных (например, item2, item3).<br><br>При настройке виджета на шаблонной панели этот параметр позволяет выбирать только элементы данных, настроенные на шаблоне. |
| *Теги элементов данных* | Укажите теги, чтобы ограничить число элементов данных, отображаемых в виджете.<br>Дополнительные сведения см. выше в разделе *Теги узлов сети*. |
| *Показывать узлы сети в обслуживании* | Установите этот флажок, чтобы отображать узлы сети в обслуживании (в этом случае рядом с именем узла сети будет показан значок обслуживания).<br>При настройке виджета на шаблонной панели этот параметр называется *Показывать данные в обслуживании*. |
| *Показывать* | Установите этот флажок, чтобы отображать соответствующий элемент ячейки honeycomb - основную метку, вторичную метку.<br>Должен быть выбран как минимум один элемент. |
| *Дополнительная конфигурация* | Нажмите заголовок *Дополнительная конфигурация*, чтобы отобразить параметры дополнительной конфигурации. |

Расширенная настройка

Параметры расширенной настройки доступны а сворачиваемом разделе *Расширенная настройка (Advanced configuration)* и только для элементов, отмеченных в поле *Показать (Show)* (см. выше), а также для цвета фона или пороговых значений для ячеек сот.




---

| | |
|---|---|
| ***Первичная/Вторичная метка (Primary/Secondary label)*** | |
| *Тип (Type)* | Выберите тип метки:<br>**Текст (Text)** — метка будет отображать текст, указанный для параметра *Текст (Text)*;<br>**Значение (Value)** — метка будет отображать значение элемента данных с количеством десятичных знаков, как указано для параметра *Число десятичных знаков (Decimal places)*. |
| *Текст (Text)* | Введите текст метки. Этот текст может переопределять имя элемента данных по умолчанию.<br>Поддерживается многострочный текст. Возможна комбинация текста и поддерживаемых макросов.<br>Поддерживаются `{HOST.\*}`, `{ITEM.\*}`, `{INVENTORY.\*}`, а также пользовательские макросы.<br>Ячейки сот сортируются в алфавитном порядке по имени узла сети и затем, для каждого узла сети — по имени элемента данных.<br>Этот параметр доступен, если *Тип (Type)* имеет значение *Текст (Text)*. |
| *Число десятичных знаков (Decimal places)* | Введите количество десятичных знаков для отображения значения.<br>Этот параметр доступен, если *Тип (Type)* имеет значение *Значение (Value)*, и имеет смысл только для элементов данных, которые возвращают числовые (с плавающей точкой) значения. |
| *Размер (Size)* | Выберите размер метки:<br>**Автоматически (Auto)** — использовать размер метки, настраиваемый автоматически;<br>**Пользовательское (Custom)** — ввести пользовательский размер метки (в процентах, относительно размера ячейки сот).<br>Обратите внимание, что метки, которые не вмещаются в размер ячейки сот, усекаются. |
| *Жирный (Bold)* | Установите флажок для отображения подписи данных жирным шрифтом. |

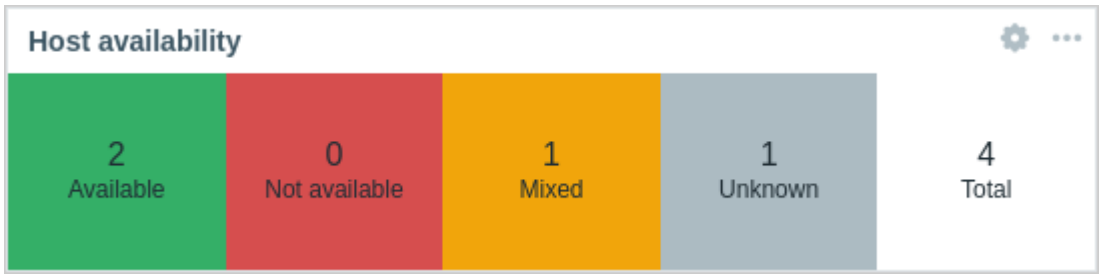
| | |
|---|---|
| *Цвет (Color)* | Выберите цвет подписи из палитры.<br>«D» означает цвет по умолчанию, который зависит от темы веб-интерфейса. Чтобы вернуться к цвету по умолчанию, нажмите в палитре на кнопку *Использовать умолчание (Use default)*. |
| ***Единицы измерения (Units)*** | |
| *Единицы измерения (Units)* | Установите флажок для отображения единиц измерения со значением элемента данных.<br>Если вы вводите название единиц измерения, оно будет переопределять единицы измерения, заданные в настройках элемента данных.<br>Этот параметр доступен, если *Тип (Type)* имеет значение «Значение (Value)». |
| *Расположение (Position)* | Выберите расположение единиц измерения элемента данных (до или после значения элемента данных).<br>Этот параметр игнорируется для следующих единиц измерения, относящихся ко времени: unixtime, uptime, s.<br>Этот параметр доступен, если *Тип (Type)* имеет значение «Значение (Value)», а параметр *Единицы измерения (Units)* отмечен. |
| ***Цвет фона (Background color)*** | |
| *Цвет фона (Background color)* | Выберите из палитры цвет фона ячейки сот.<br>«D» означает цвет по умолчанию, который зависит от темы веб-интерфейса. Чтобы вернуться к цвету по умолчанию, нажмите в палитре на кнопку *Использовать умолчание (Use default)*. |
| ***Пороги (Thresholds)*** | |
| *Интерполяция цветов (Color interpolation)* | Установите флажок, чтобы разрешить для ячеек сот плавный переход между цветами пороговых значений.<br>Этот параметр доступен, если заданы два или более пороговых значения. |
| *Порог (Threshold)* | Нажмите *Добавить (Add)* для добавления порога, выберите из палитры цвет порога и укажите числовое значение.<br>Список порогов будет при сохранении отсортирован в возрастающем порядке.<br>Обратите внимание, что цвета, настроенные для порогов, будут правильно отображаться только для числовых элементов данных.<br>Поддерживаются суффиксы (например, «1d», «2w», «4K», «8G»). Поддерживаются преобразования значений. |

12 Доступность узла сети

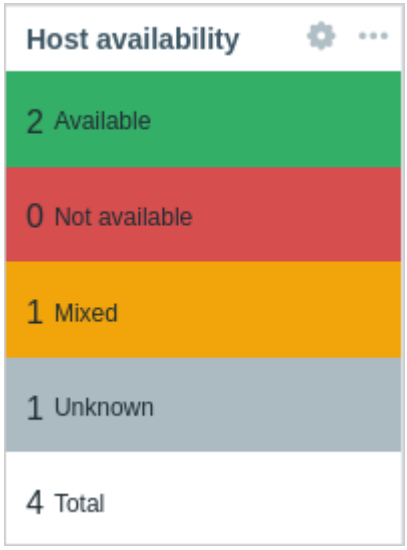
Обзор

На виджете доступности узла сети общая статистика о доступности узла сети отображается в подкрашенных столбцах/строках в зависимости от выбранного макета.

Горизонтальное отображение (столбцы):




Вертикальное отображение (строки):



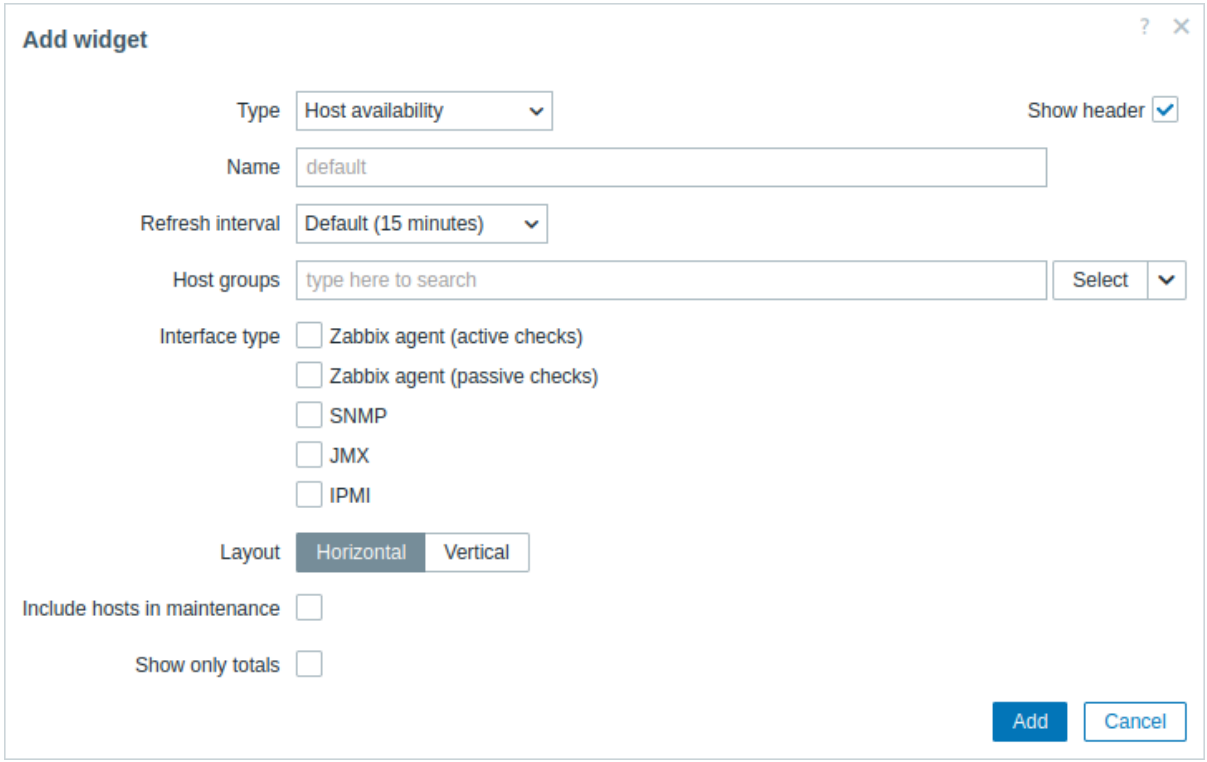

Доступность узла сети в каждом столбце/строке учитывается следующим образом:

- *Доступен (Available)* — узлы сети со всеми доступными интерфейсами
- *Недоступен (Not available)* — узлы сети со всеми недоступными интерфейсами
- *Смешанные (Mixed)* — узлы сети с хотя бы одним недоступным интерфейсом и хотя бы одним интерфейсом либо доступным, либо неизвестным; остальные могут иметь любое значение, включая «неизвестно (unknown)»
- *Неизвестно (Unknown)* — узлы сети с хотя бы одним интерфейсом в неизвестном состоянии (не недоступен)
- *Всего (Total)* — общее количество узлов сети

**Note:**
В случае Zabbix агента (активные проверки), ячейка *Смешанные (Mixed)* всегда будет пустой, т.к. этот тип элементов данных не может иметь несколько интерфейсов.

Настройка

Для настройки выберите *Доступность узла сети (Host availability)* в качестве типа:



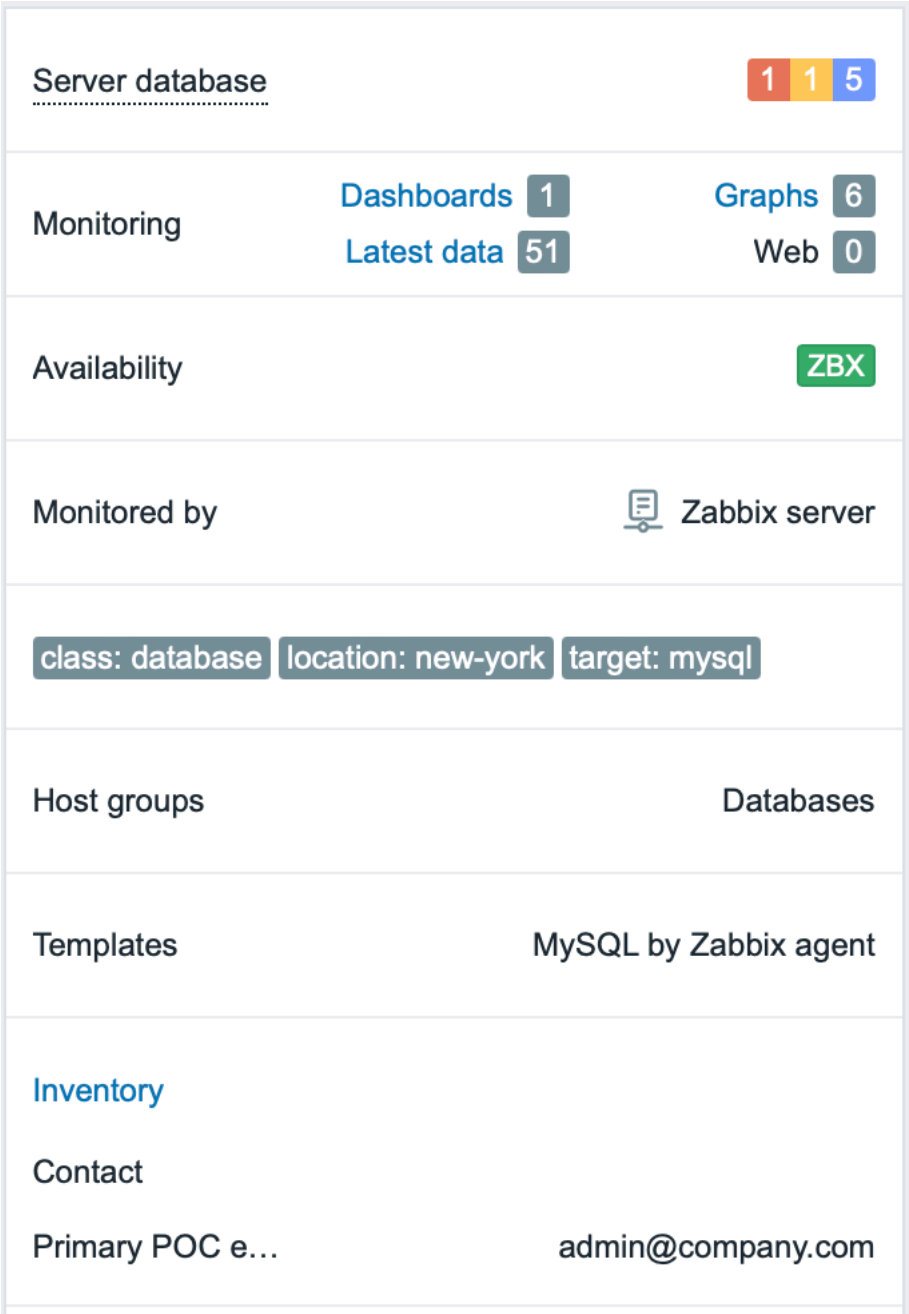

В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Типы интерфейсов (Interface type)* | Выберите, для каких интерфейсов узлов сети вы хотите видеть данные о доступности.<br>Если ничего не выбрано, то по умолчанию отображается доступность всех интерфейсов. |
| *Размещение (Layout)* | Выберите горизонтальное отображение (столбцы) или вертикальное отображение (строки). |
| *Включить узлы сети в обслуживании (Include hosts in maintenance)* | Включать ли в статистику узлы сети в состоянии обслуживания.<br>Этот параметр помечен *Показывать данные в обслуживании (Show data in maintenance)* при настройке виджета на панели шаблона. |
| *Только итоговые значения (Show only totals)* | Если отмечено, то отображается общее количество узлов сети, без разбиения по интерфейсам. Эта опция отключена, если выбран только один интерфейс. |

## 13 Карточка узла сети

### Обзор

Виджет карточки узла сети предоставляет полную и настраиваемую информацию об одном узле сети. Узел сети можно указать через конфигурацию виджета, выбрать из другого совместимого виджета или найти с помощью селектора узлов сети на уровне панели мониторинга.

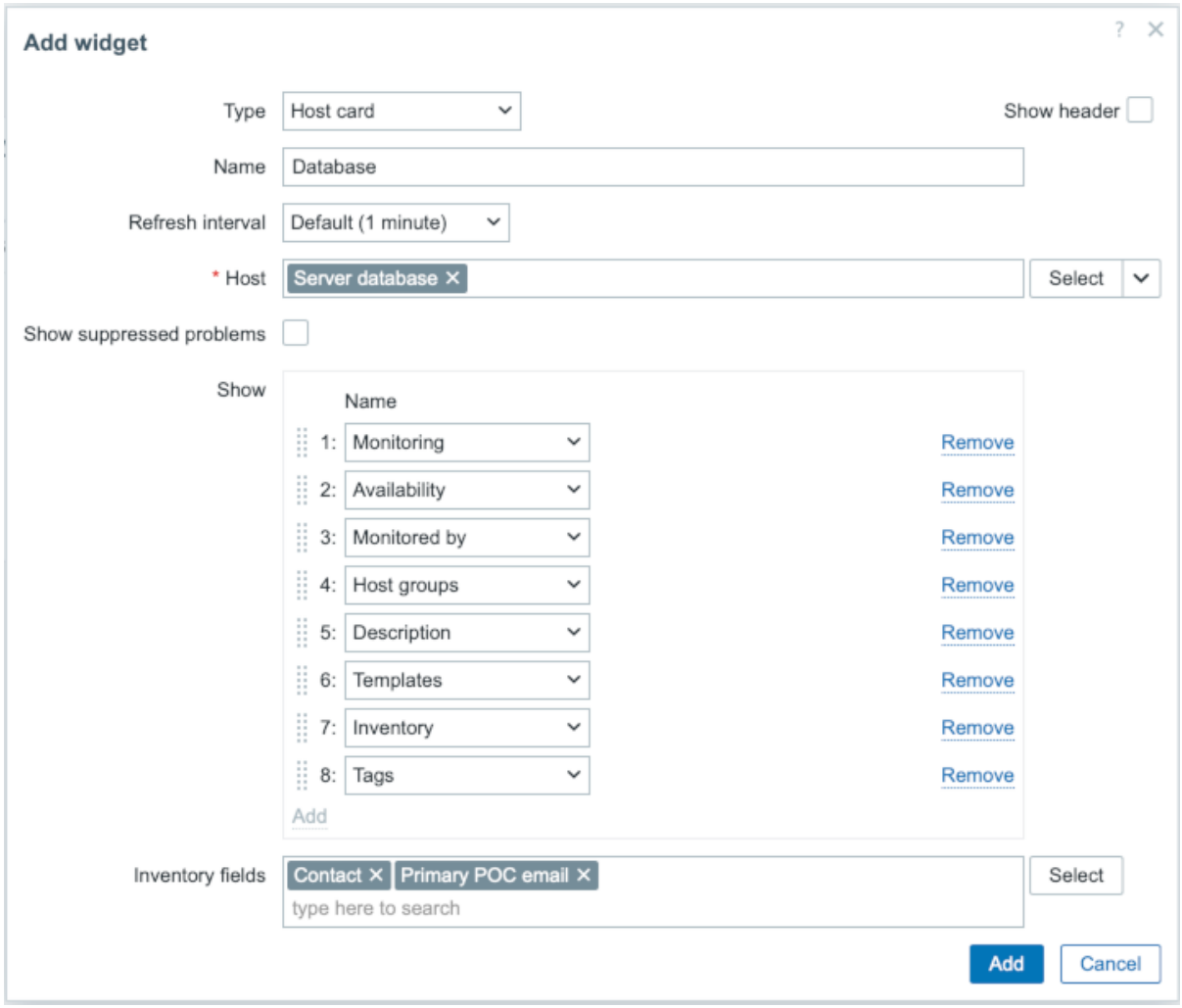



Верхняя часть виджета содержит имя узла сети и количество проблем узла сети, сгруппированных по степени важности. Если узел сети отключен, рядом с его именем будет отображаться текст *Disabled*. Если узел сети находится на обслуживании, рядом с его именем будет отображаться значок обслуживания. Другие разделы виджета можно настраивать по мере необходимости.

Этот виджет автоматически изменяет расположение элементов в зависимости от ширины. Чтобы разместить разделы виджета в несколько столбцов, расширьте виджет по горизонтали.

Конфигурация

Чтобы настроить, выберите тип *Host card*:




Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

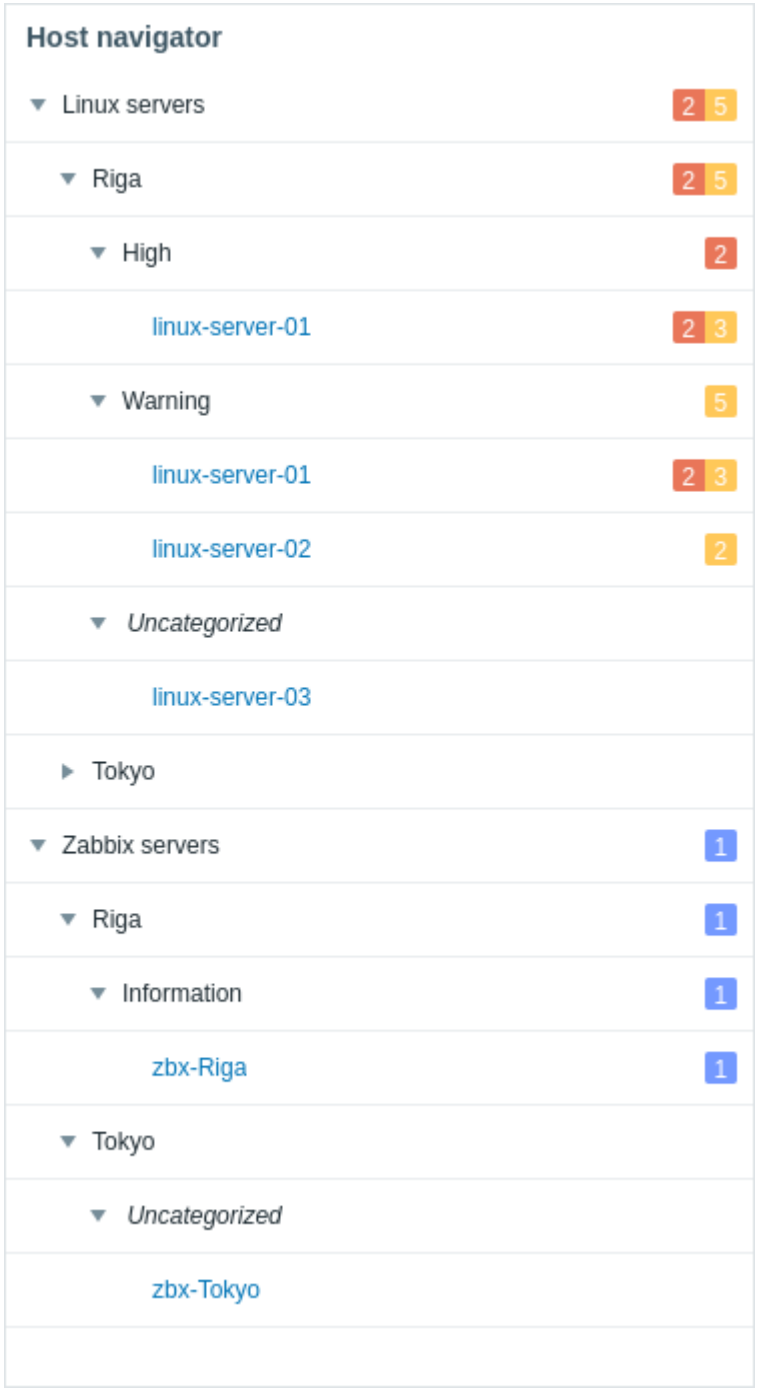
| | |
|---|---|
| *Host* | Выберите узел сети.<br>Либо выберите совместимый виджет или панель как источник данных.<br>По мере ввода имени это поле будет предлагать подходящие варианты. Также можно нажать кнопку *Select*, чтобы выбрать вручную.<br><br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Show suppressed problems* | Отметьте этот флажок, чтобы включить подавленные проблемы в расчет проблем. |
| *Show* | Настройте до восьми разделов, которые будут отображаться виджетом. Разделы можно менять местами перетаскиванием. |
| *Host groups* | Группы узлов сети, к которым принадлежит узел сети. |
| *Description* | Описание узла сети. |
| *Monitoring* | Быстрые ссылки для перехода к панелям, последним данным, графикам и веб-сценариям узла сети. Число рядом с каждой ссылкой указывает количество сущностей. |
| *Availability* | Доступность узла сети по интерфейсам. |
| *Monitored by* | Сервер Zabbix, прокси или группа прокси, которыми отслеживается узел сети. |
| *Templates* | Шаблоны, связанные с узлом сети. |
| *Inventory* | Показывать поля инвентаризации узла сети. Если выбрано, появится параметр *Inventory fields*, позволяющий указать, какие поля следует отображать. |
| *Tags* | Теги узла сети. |

| | |
|---|---|
| *Inventory fields* | Укажите поля инвентаризации, которые нужно отображать.<br>По мере ввода имени это поле будет предлагать подходящие варианты. Либо можно нажать кнопку *Select*, чтобы выбрать вручную.<br>Если поля инвентаризации не указаны, будут отображаться все заполненные поля инвентаризации.<br><br>Этот параметр становится доступен только тогда, когда в разделе *Show* выбрано *Inventory*. |

14. Навигатор по узлам сети

Обзор

Виджет навигатора по узлам сети отображает узлы сети на основе различных параметров фильтрации и группировки.




Этот виджет также позволяет управлять информацией, отображаемой в других виджетах, в зависимости от выбранного

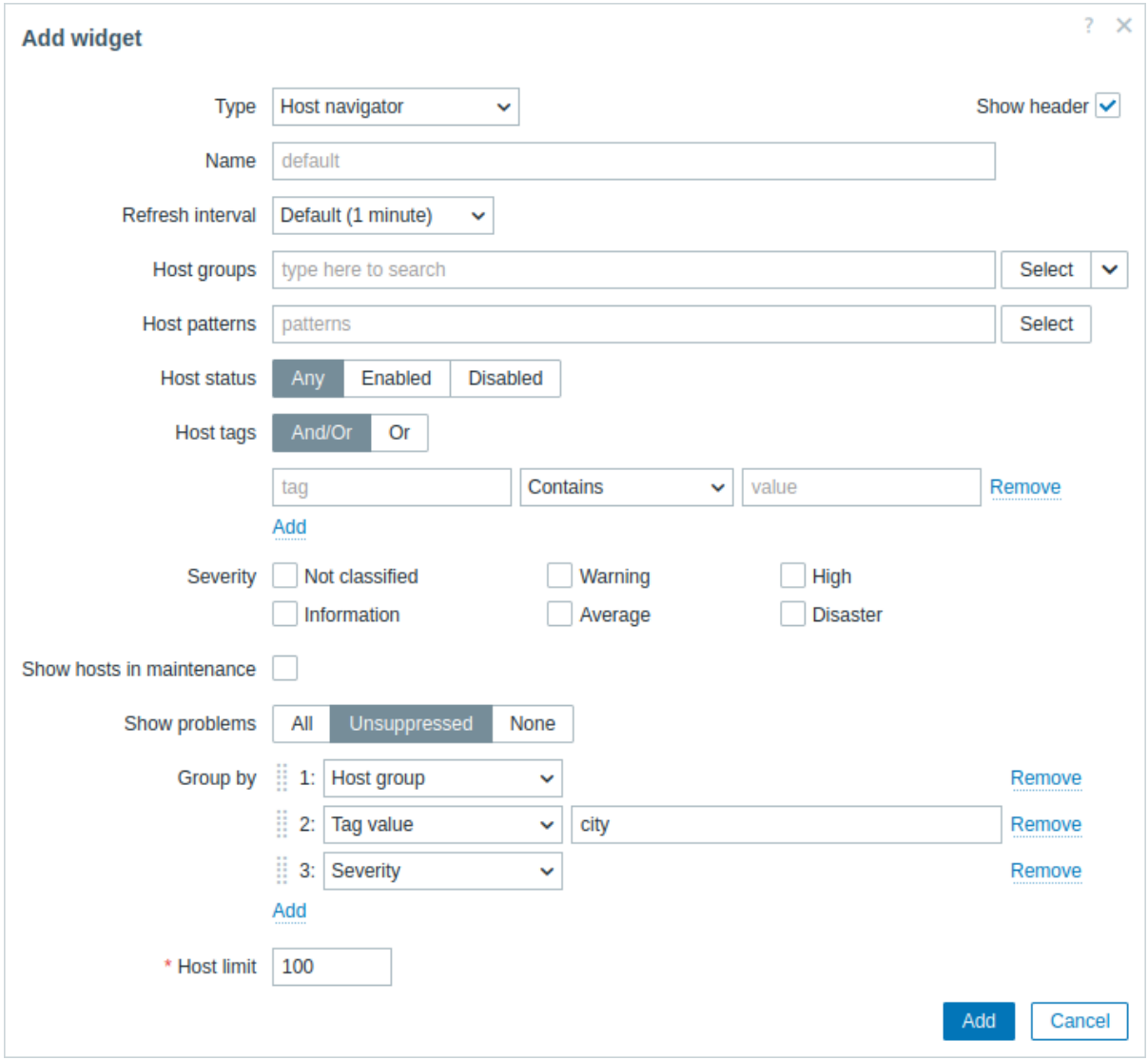
узла сети.

Группы, по которым организованы узлы сети, могут раскрываться или сворачиваться.

Для групп, проблем и узлов сети в состоянии обслуживания дополнительные сведения доступны при наведении курсора мыши.

Настройка

Для настройки выберите *Навигатор по узлам сети (Host navigator)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительскойгруппы узлов сети неявно выбираются все вложенные группы узлов сети; если не выбрана ни одна группа, то виджет будет отображать все узлы сети из всех групп.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |

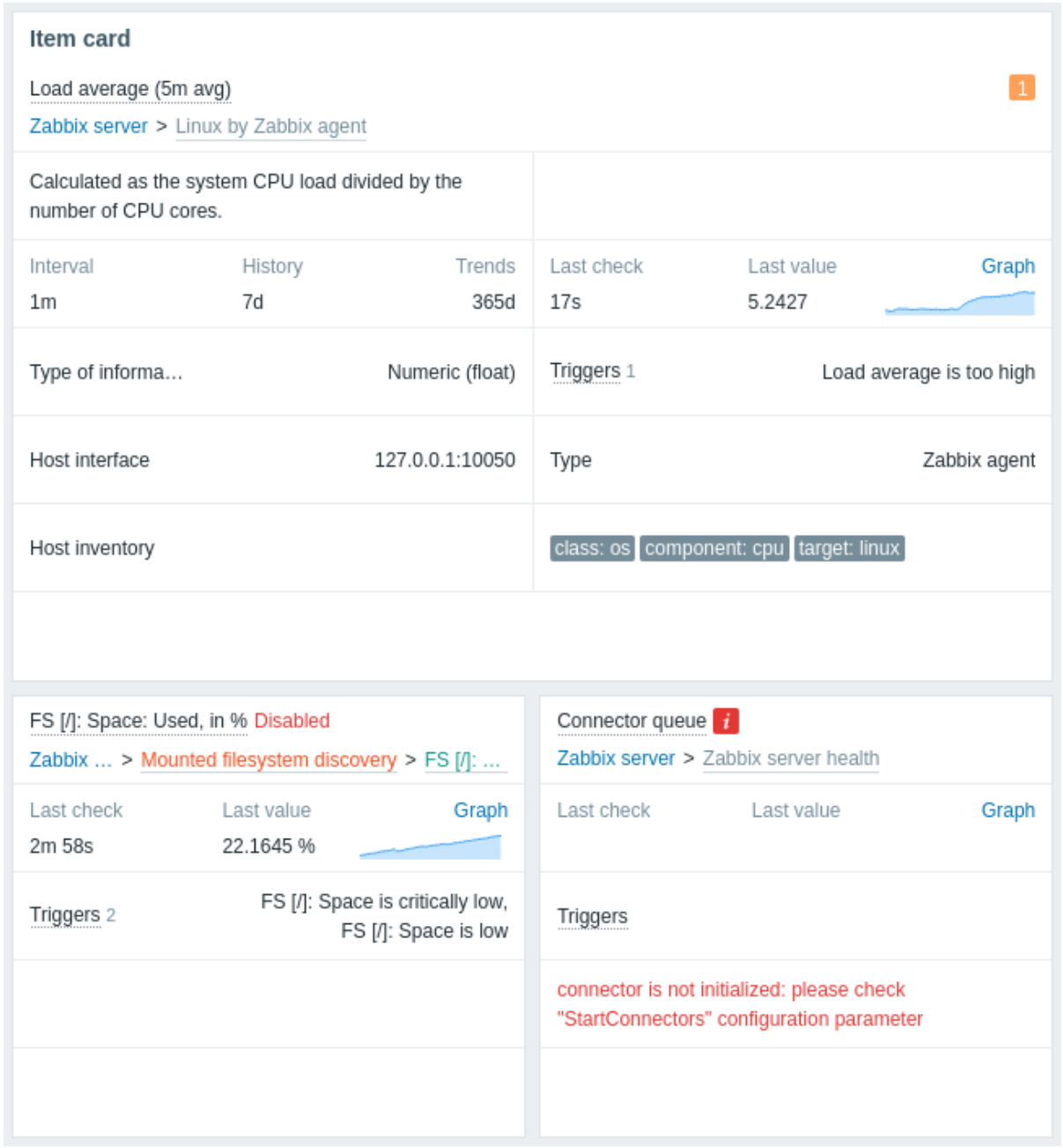
| | |
|---|---|
| *Шаблоны узлов сети (Host patterns)* | Введите шаблоны узлов сети или выберите существующие узлы сети в качестве шаблонов узлов сети. Узлы сети, соответствующие указанным шаблонам, будут отображаться в навигаторе по узлам сети.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени узла сети будет автоматически предложен список совпадающих узлов сети в выпадающем меню.<br>Если никаких узлов сети не выбрано, то виджет будет отображать все узлы сети.<br><br>Для выбора могут использоваться шаблоны с подстановочными символами (например, `*` вернёт узлы сети, соответствующие нулю или более символов; `Zabbix*` вернёт узлы сети, которые начинаются на «Zabbix»).<br>Чтобы указать шаблон с подстановочными символами, введите строку вручную и нажмите *Enter*. При начале вводавыпадающий список отобразит соответствующие узлы сети, ограниченные теми, которые относятся к узлас сети из выбранных *Групп узлов сети (Host groups*), если имеются.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Состояние узлов сети (Host status)* | Фильтровать, какие узлы сети отображать, на основе их состояния (любое, деактивировано, деактивировано — any, enabled, disabled).<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Теги узлов сети (Host tags)* | Укажите теги для фильтрации узлов сети, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно указать несколько условий.<br>Поиск соответствия имён тегов всегда чувствителен к регистру.<br><br>По каждому условию доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием Или (Or);<br>**Или (Or)** — достаточно соблюдения одного условия.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Важность (Severity)* | Отметьте важности проблем для фильтрации узлов сети с проблемами при отображении в виджете.<br>Если никакие важности не отмечены, то будут отображаться все узлы сети со всеми проблемами. |
| *Узлы сети в обслуживании (Show hosts in maintenance)* | Установите этот флажок для отображения узлов сети в режиме обслуживания (в этом случае рядом с именем узла сети будет отображаться значок режима обслуживания).<br><br>При настройке виджета на панели шаблона этот параметр помечен *Show data in maintenance*. |
| *Проблемы (Show problems)* | Фильтровать, какие проблемы отображать в виджете с узлами сети на основе их состояния (все, подавление отменено, нет — all, unsuppressed, none). |

| | |
|---|---|
| *Группировка по (Group by)* | Добавить атрибут группировки, по которому группировать выбранные узлы сети:<br>**Группа узлов сети (Host group)** — группировать узлы сети по их группе узлов сети;<br>**Значение тега (Tag value)** — введите имя тега для группировки узлов сети по значению этого тега (например, введите «city», чтобы группировать узлы сети по значениям «Riga», «Tokyo» и т.д.);<br>**Важность (Severity)** — группировать узлы сети по важности их проблем.<br><br>Если параметр *Проблемы (Show problems)* настроен на отображение проблем, они отображаются следующим образом:<br>- для каждой группы важности отображается только количество соответствующих проблем;<br>- для каждого узла сети отображается количество всех его проблем.<br>Обратите внимание, что узлы сети будут сгруппированы только по тем уровням важности, которые отмечены в параметре *Важность (Severity)*; если никакие уровни важности не выбраны, все узлы сети будут сгруппированы по всем важностям.<br><br>Атрибуты группировки могут быть переупорядочены путём перетаскивания вверх или вниз за ползунок перед именем группы. Обратите внимание, что порядок атрибутов группировки определяет порядок вложенности групп. Например, указание нескольких имён тегов (1: цвет, 2: город) приведёт к тому, что узлы сети будут сгруппированы по цвету (красный, синий и т. д.), а затем по городу (Рига, Токио и т. д.).<br><br>В зависимости от настроенных атрибутов группировки узел сети может отображаться в нескольких группах (например, при группировке по группам узлов сети, когда узел сети принадлежит нескольким группам). Щелчок по таким узлам сети выделяет и подсвечивает их во всех группах.<br>Узлы сети, не соответствующие заданным атрибутам группировки, отображаются в группе *Без категории (Uncategorized)*.<br><br>Можно указать до 10 атрибутов группировки, и все они должны быть уникальными.<br>Если атрибуты группировки не указаны, узлы сети не будут сгруппированы. |
| *Лимит узлов сети (Host limit)* | Введите максимальное количество отображаемых узлов сети. Возможные значения: 1-9999.<br><br>Когда для отображения доступно больше узлов сети, чем задано лимитом, то ниже отображаемых узлов сети показывается соответствующее сообщение (например: «100 of 100+ hosts are shown (показаны 100 из 100+ узлов сети)»).<br>Обратите внимание, что настроенный лимит узлов сети также влияет на отображение настроенных групп; например, если лимит узлов сети составляет 100 и узлы сети группируются по значениям тегов (более 200), то в виджете будут отображены только первые 100 значений тегов с соответствующими узлами сети.<br><br>Этот параметр не зависит от параметра *Лимит элементов в результатах поиска и фильтрах (Limit for search and filter results)* в разделе *Administration (Администрирование)* → *Общие (General)* → *Веб-интерфейс (GUI)*.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |

15 Карточка элемента данных

Обзор

Виджет карточки элемента данных предоставляет полную и настраиваемую информацию об одном элементе данных.

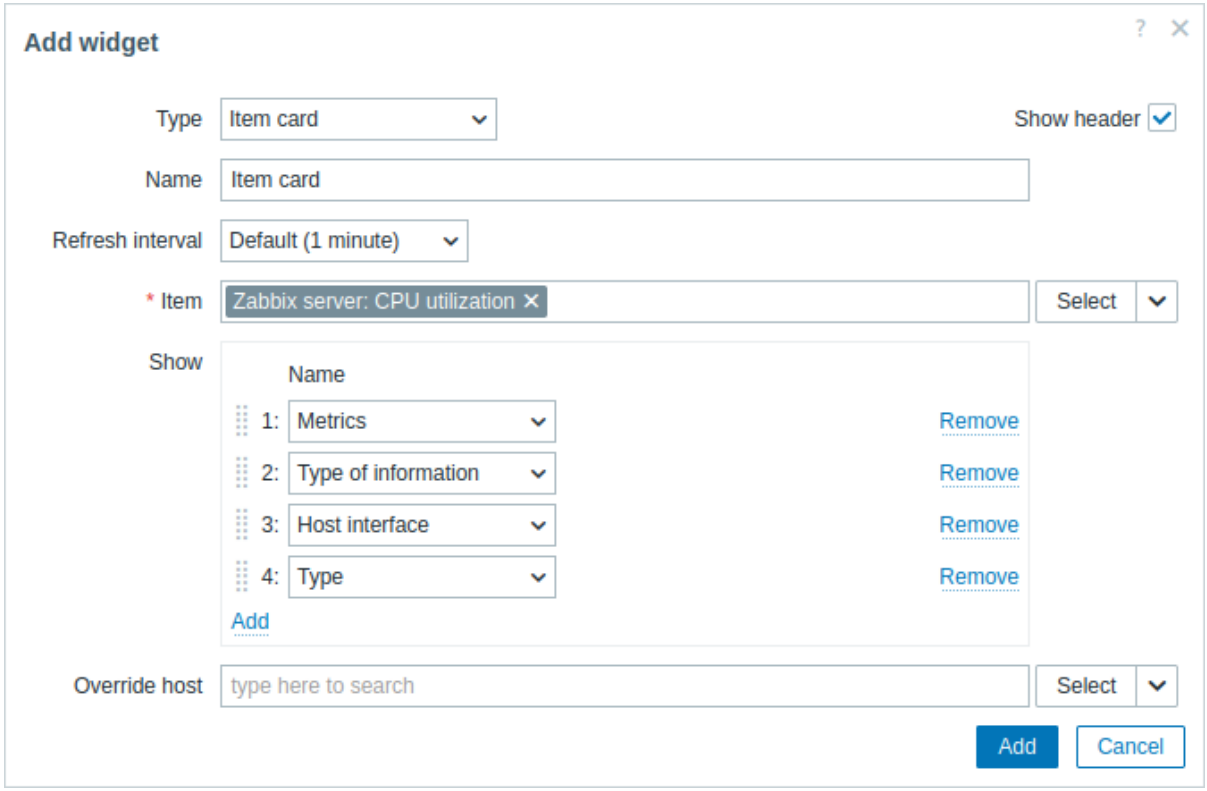



Элемент данных можно задать напрямую в конфигурации виджета или выбрать из совместимого виджета на панели мониторинга.

Виджет может отображать несколько разделов, каждый из которых показывает разный тип информации (описание элемента данных, последние данные и т. д.). Макет разделов автоматически подстраивается под ширину виджета. При горизонтальном расширении разделы перестраиваются в несколько столбцов, упорядоченных слева направо в каждой строке.

Конфигурация

Чтобы настроить, выберите *Item card* в качестве типа:



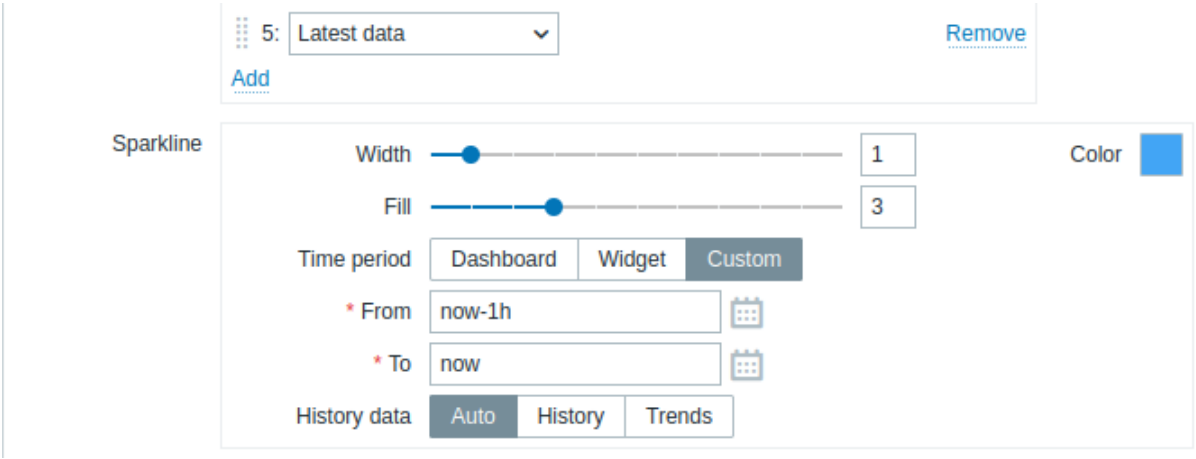

Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Item* | Выберите элемент данных.<br>Либо выберите совместимый виджет в качестве источника данных для элементов данных.<br>По мере ввода отображаются подходящие варианты.<br><br>Верхняя часть виджета всегда отображает **имя элемента данных** (щелчок по нему открывает меню элемента данных):<br>- Если элемент данных отключен, рядом с его именем отображается красная метка *Disabled*.<br>- В случае ошибок отображается квадратный значок с буквой "i". В случае проблем отображается квадратный значок с количеством проблем для каждой степени серьезности. При наведении на значок открывается всплывающее окно с подробностями.<br>- На шаблонной панели отображается техническое имя элемента данных вместо отображаемого имени.<br><br>Верхняя часть виджета также отображает **путь элемента данных** — его узел сети и, если применимо, шаблон, правило низкоуровневого обнаружения (LLD) или мастер-элемент (щелчок по сущности открывает ее настройки).<br>Путь элемента данных отображается только пользователям, у которых есть права на соответствующие сущности. |
| *Show* | Добавьте разделы для отображения в виджете. Разделы можно менять местами перетаскиванием. |
| *Description* | Отображает описание элемента данных. Макросы подставляются. |
| *Error text* | Отображает ошибку элемента данных, если она есть. |
| *Metrics* | Отображает следующую информацию:<br>- Интервал обновления элемента данных (кроме элементов данных trapper);<br>- Настройки истории элемента данных, если они есть;<br>- Настройки трендов элемента данных, если они есть.<br>Макросы подставляются.<br><br>Если элемент данных использует пользовательские интервалы, рядом с интервалом обновления отображается значок справки. При наведении на значок открывается всплывающее окно с подробностями о пользовательском интервале. |

| | |
|---|---|
| *Latest data* | Отображает следующую информацию:<br>- Время с момента последней проверки элемента данных;<br>- Последнее значение элемента данных;<br>- Ссылку на простой график/историю значений элемента данных (если история или тренды сохраняются);<br>- Настраиваемый график sparkline (если элемент данных числовой).<br><br>Для элементов данных двоичного типа вместо последнего значения отображается миниатюра или параметр *Show*. При наведении на миниатюру открывается всплывающее окно с изображением. При наведении на *Show* открывается всплывающее окно со значением элемента данных (строка Base64).<br><br>На шаблонной панели вместо последнего значения элемента данных отображается *No data*. |
| *Type of information* | Отображает тип информации элемента данных. |
| *Triggers* | Отображает следующую информацию:<br>- Метка *Triggers* с количеством триггеров (при наведении на *Triggers* открывается всплывающее окно с подробностями о триггерах);<br>- Список имен триггеров (макросы подставляются). |
| *Host interface* | Отображает интерфейс узла сети, используемый элементом данных, или *No data*, если у элемента данных нет интерфейса узла сети (например, у зависимых элементов данных, простых проверок).<br><br>На шаблонной панели отображается *No data*. |
| *Type* | Отображает тип элемента данных. |
| *Host inventory* | Отображает поле инвентаря узла сети, которое заполняется значением элемента данных. |
| *Tags* | Отображает теги элемента данных. |
| *Override host* | Выберите совместимый виджет или панель в качестве источника данных для узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |

Sparkline

Параметры настройки графика sparkline доступны только если в виджет добавлен раздел *Latest data*.



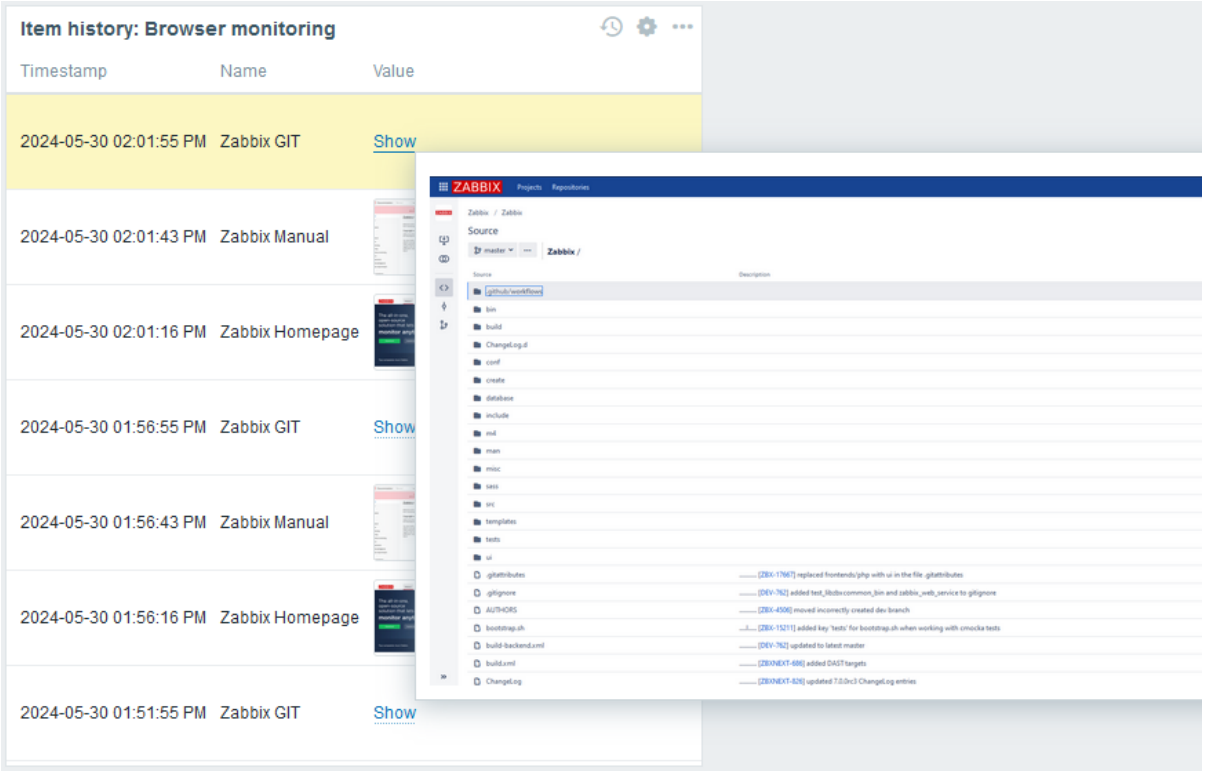

| | |
|---|---|
| *Width* | Задайте толщину линии графика с помощью ползунка или вручную введите значение в диапазоне от 0 до 10. |
| *Color* | Выберите цвет линии и заливки. |
| *Fill* | Задайте уровень прозрачности цвета заливки с помощью ползунка или вручную введите значение в диапазоне от 0 до 10. |

| | |
|---|---|
| *Time period* | Выберите источник данных для периода времени:<br>**Dashboard** - используйте селектор *Time period* в качестве источника данных;<br>**Widget** - используйте совместимый виджет, указанный в параметре *Widget*, в качестве источника данных;<br>**Custom** - используйте период времени, указанный в параметрах *From* и *To*, в качестве источника данных; если задано, в правом верхнем углу виджета будет отображаться значок часов, указывающий установленное время при наведении курсора. |
| *Widget* | Введите или выберите совместимый виджет в качестве источника данных для периода времени.<br>Этот параметр доступен, если для *Time period* задано значение "Widget". |
| *From* | Введите или выберите начало периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* задано значение "Custom". |
| *To* | Введите или выберите конец периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* задано значение "Custom". |
| *History data* | Брать данные из истории или трендов:<br>**Auto** - автоматический выбор;<br>**History** - брать данные истории;<br>**Trends** - брать данные трендов.<br>Этот параметр применяется только к числовым данным. Нечисловые данные всегда будут браться из истории. |

16 История элемента данных

Обзор

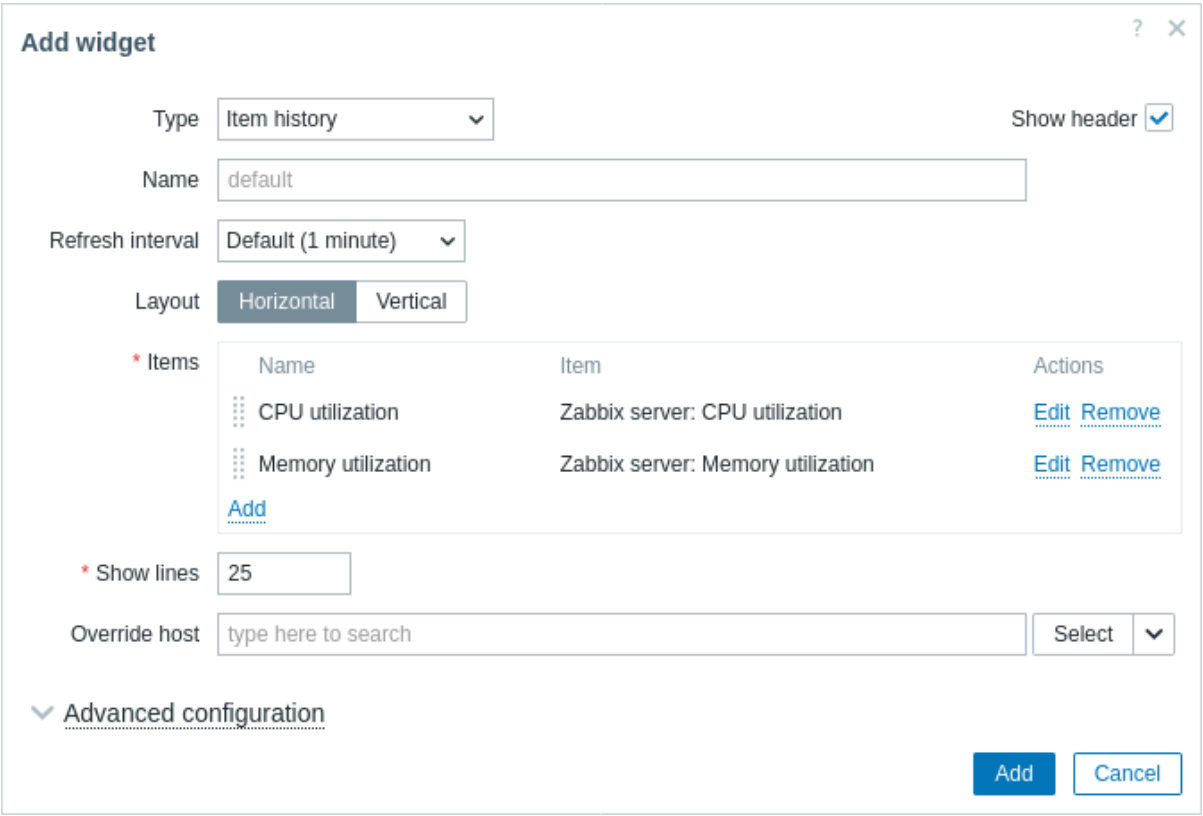
Виджет истории элемента данных отображает последние данные для различных типов элементов данных (числовые, символьные, логовые, текстовые и двоичные) в табличном формате. Он также может показывать индикаторы выполнения, изображения для двоичных типов данных (полезно для элементов браузера) и выделять значения (полезно для мониторинга лог-файлов).




Можно отображать до 1000 записей.

Конфигурация

Чтобы настроить, выберите *Item history* в качестве типа:



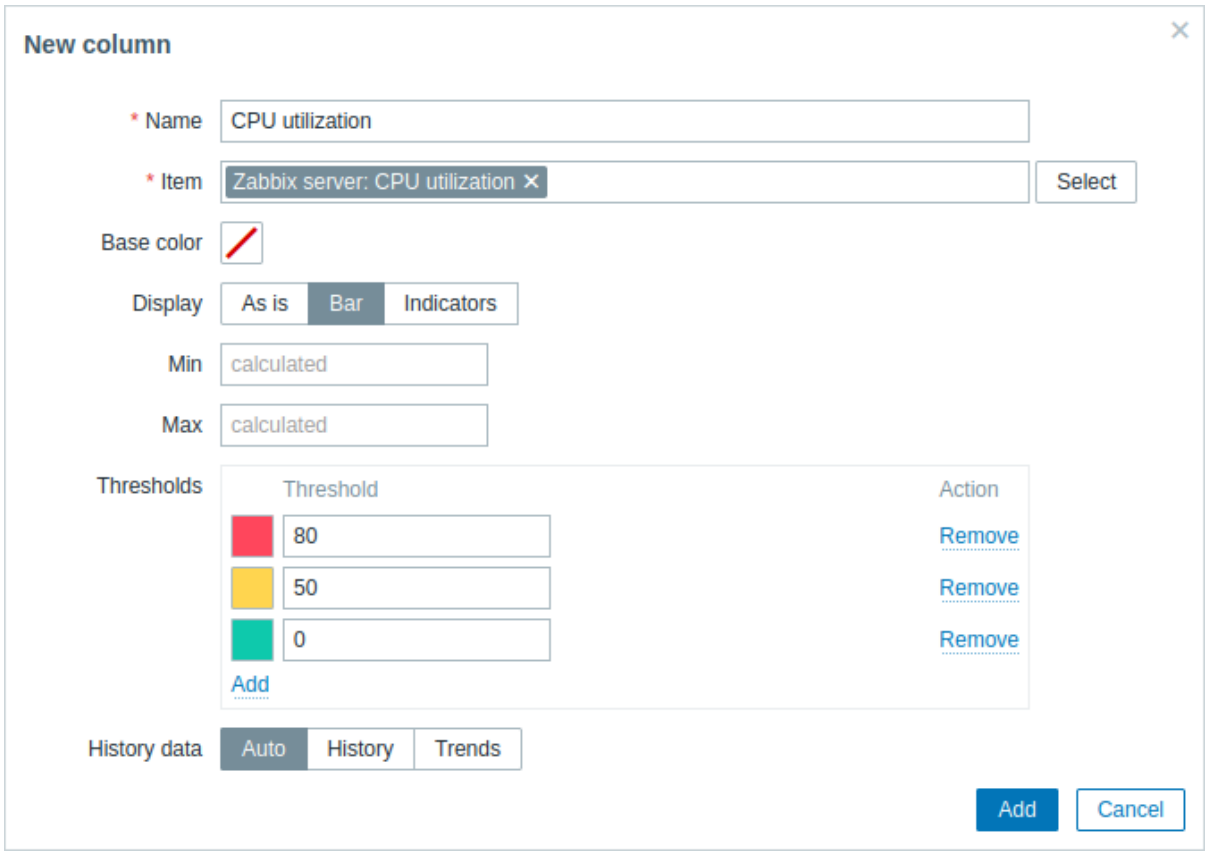

Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Layout* | Выберите вариант расположения для столбцов элементов данных:<br>**Horizontal** - элементы данных будут отображаться горизонтально, значения - вертикально;<br>**Vertical** - элементы данных будут отображаться вертикально, значения - горизонтально. |

| | |
|---|---|
| *Items* | Добавьте столбцы элементов данных для отображения.<br>Порядок элементов данных определяет порядок их отображения.<br>Элементы данных можно менять местами, перетаскивая их вверх или вниз за маркер перед именем элемента данных. |
| *Show lines* | Укажите количество строк значений элемента данных для отображения. |
| *Override host* | Выберите совместимый виджет или панель мониторинга в качестве источника данных для узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели мониторинга. |
| *Advanced configuration* | Щелкните заголовок *Advanced configuration*, чтобы отобразить параметры расширенной настройки. |

Настройка столбцов

Чтобы настроить столбцы элементов данных, нажмите *Add* в параметре *Items*:




Общие параметры колонок:

| | |
|---|---|
| *Имя (Name)* | Введите имя колонки.<br>Если оставить пустым, используется имя элемента данных из параметра *Элемент данных (Item)*. |
| *Элемент данных (Item)* | Выберите элемент данных.<br>Обратите внимание, что параметры настройки колонки отличаются в зависимости от типа информации выбранного элемента данных; для более подробных сведений смотрите отдельные параметры ниже.<br>При настройке виджета на панели шаблона могут быть выбраны только элементы данных, настроенные в этом шаблоне. |
| *Основной цвет (Base color)* | Выберите цвет фона колонки или цвет заливки, если параметр *Отображение (Display)* имеет значения «Столбцы (Bar)» или «Индикатор (Indicators)».<br>Обратите внимание, что основной цвет может быть переопределён цветом порога или подсветки. |

Параметры колонок, специфичные для числовых типов элементов данных:

| | |
|---|---|
| *Отображение (Display)* | Выберите, как следует отображать значение элемента данных:<br>**Как есть (As is)** — в виде обычного текста;<br>**Столбцы (Bar)** — в виде сплошной цветной полосы;<br>**Индикаторы (Indicators)** — в виде сегментированной цветной полосы. |
| *Мин (Min)* | Введите минимальное значение для полосы/индикаторов.<br>Если оставить пустым, виджет будет использовать минимальное значение элемента данных.<br>Этот параметр доступен, только когда параметр *Отображение (Display)* имеет значения «Столбцы (Bar)» или «Индикаторы (Indicators)». |
| *Макс (Max)* | Введите максимальное значение для полосы/индикаторов.<br>Если оставить пустым, виджет будет использовать максимальное значение элемента данных.<br>Этот параметр доступен, только когда параметр *Отображение (Display)* имеет значения «Столбцы (Bar)» или «Индикаторы (Indicators)». |
| *Пороги (Thresholds)* | Нажмите *Добавить (Add)* для добавления порога, выберите цвет порога из палитры и укажите числовое значение.<br>При сохранении список порого будет отсортирован по возрастанию.<br>Поддерживаются суффиксы (например, «1d», «2w», «4K», «8G»). Поддерживаются преобразования значений. |
| *Данные истории (History data)* | Выберите, брать данные из истории или из динамики изменений:<br>**Автоматически (Auto)** — автоматический выбор;<br>**История (History)** — брать данные истории;<br>**Динамика изменений (Trends)** — брать данные динамики изменений. |

Параметры столбца, специфичные для элементов данных типа character, text и log:

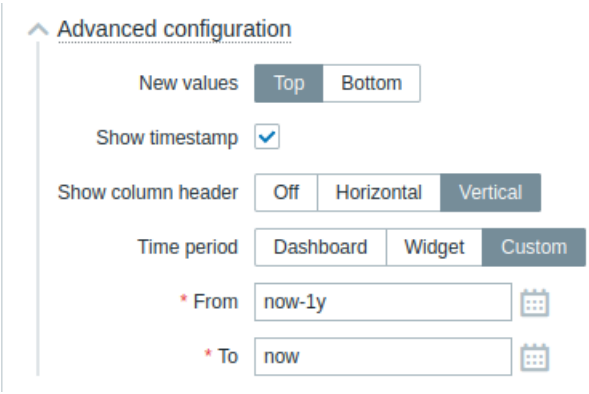
| | |
|---|---|
| *Подсветка* | Нажмите *Добавить*, чтобы добавить подсветку, выберите цвет подсветки в палитре цветов и укажите регулярное выражение.<br>Выбранный цвет будет использоваться как цвет фона для значений элемента данных, где указанное регулярное выражение совпадает с текстом. |
| *Отображение* | Выберите, как должно отображаться значение элемента данных:<br>**Как есть** - точно в том виде, в котором оно получено. Разрывы строк сохраняются, если значение содержит символы перевода строки. Перенос слов применяется, если для *Макет* задано значение ”Вертикальный” и настроено более одного столбца;<br>**HTML** - как текст, отформатированный в HTML;<br>**Одна строка** - как одна строка, обрезанная до указанной длины (1-500 символов).<br>При наведении курсора или щелчке по обрезанному значению открывается всплывающее окно с полным значением. |
| *Использовать моноширинный шрифт* | Отметьте этот флажок, чтобы отображать значение элемента данных моноширинным шрифтом (по умолчанию не отмечен). |
| *Отображать локальное время* | Отметьте этот флажок, чтобы отображать локальное время вместо временной метки в столбце временной метки.<br>Обратите внимание, что флажок *Показывать временную метку* в расширенной настройке также должен быть отмечен.<br>Этот параметр доступен только для элементов данных типа log. |

Параметры колонок, специфичные для элементов данных бинарного типа:

| | |
|---|---|
| *Миниатюра (Show thumbnail)* | Установите этот флажок для отображения миниатюры для бинарных значений с изображениями или параметром «Показать (Show)» для бинарных значений, не являющихся изображениями.<br>Снимите этот флажок для отображения параметра «Показать (Show)» для всех бинарных значений элементов данных.<br>При наведении курсора мыши или нажатии на кнопку «Показать (Show)» открывается всплывающее окно со значением элемента данных (изображением или строкой Base64).<br>Если значение элемента данных является пустой строкой, параметр «Показать (Show)» отображается; при наведении курсора мыши или нажатии открывается всплывающее окно, содержащее «Пустая строка» (”Empty string”). |

Дополнительная настройка

Параметры дополнительной настройки доступны в сворачиваемом разделе *Дополнительная настройка*:

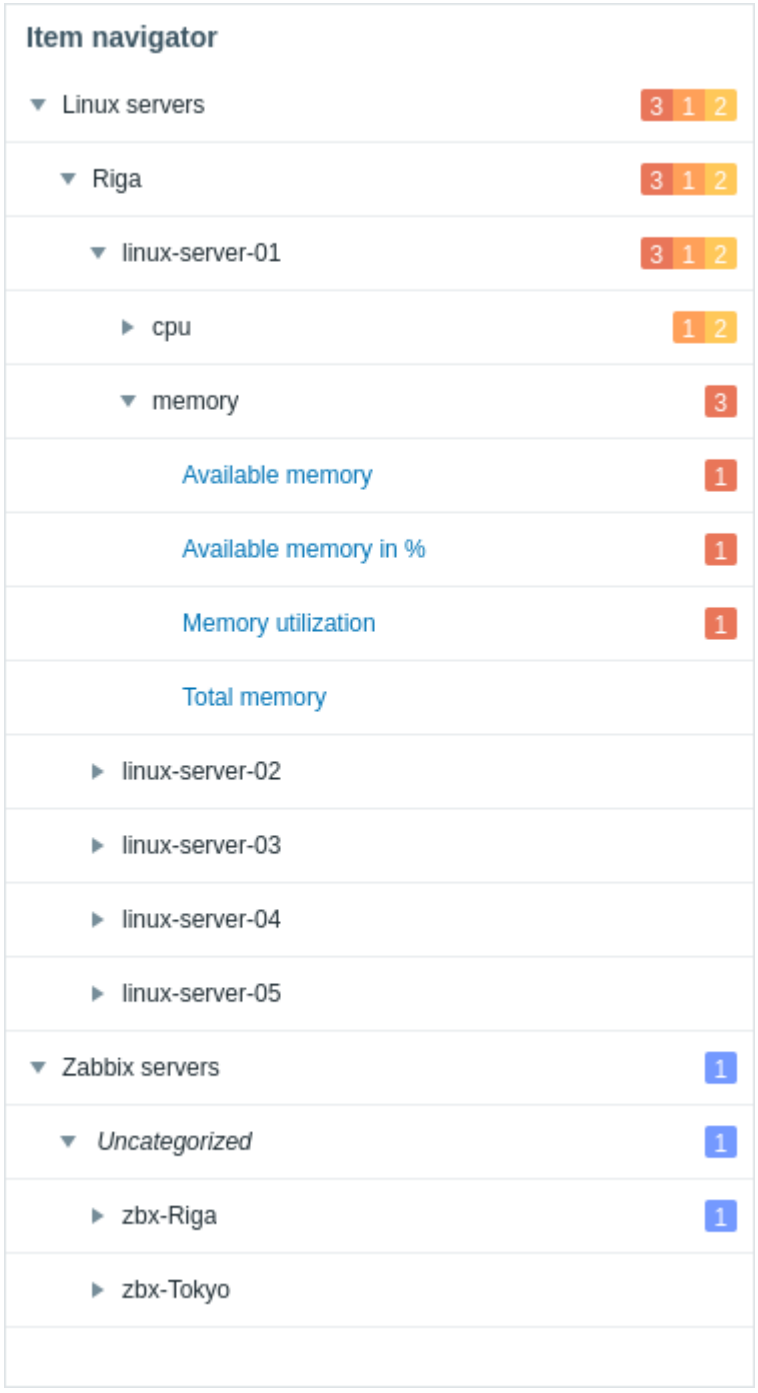



| | |
|---|---|
| *New values* | Выберите, куда будут добавляться новые значения элемента данных:<br>**Top** - в верхнюю часть столбцов;<br>**Bottom** - в нижнюю часть столбцов. |
| *Show timestamp* | Установите этот флажок, чтобы отображать столбец с временной меткой (по умолчанию не установлен). |
| *Show column header* | Выберите ориентацию заголовка столбца:<br>**Off** - скрыть заголовок;<br>**Horizontal** - отображать заголовок горизонтально;<br>**Vertical** - отображать заголовок вертикально. |
| *Time period* | Выберите источник данных для периода времени:<br>**Dashboard** - использовать селектор *Time period* в качестве источника данных;<br>**Widget** - использовать совместимый виджет, указанный в параметре *Widget*, в качестве источника данных;<br>**Custom** - использовать период времени, указанный в параметрах *From* и *To*, в качестве источника данных; если этот вариант выбран, в правом верхнем углу виджета будет отображаться значок часов, показывающий заданное время при наведении курсора. |
| *Widget* | Введите или выберите совместимый виджет (*Graph*, *Graph (classic)*, *Graph prototype*) в качестве источника данных для периода времени.<br>Этот параметр доступен, если для *Time period* установлено значение "Widget". |
| *From* | Введите или выберите начало периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* установлено значение "Custom". |
| *To* | Введите или выберите конец периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Time period* установлено значение "Custom". |

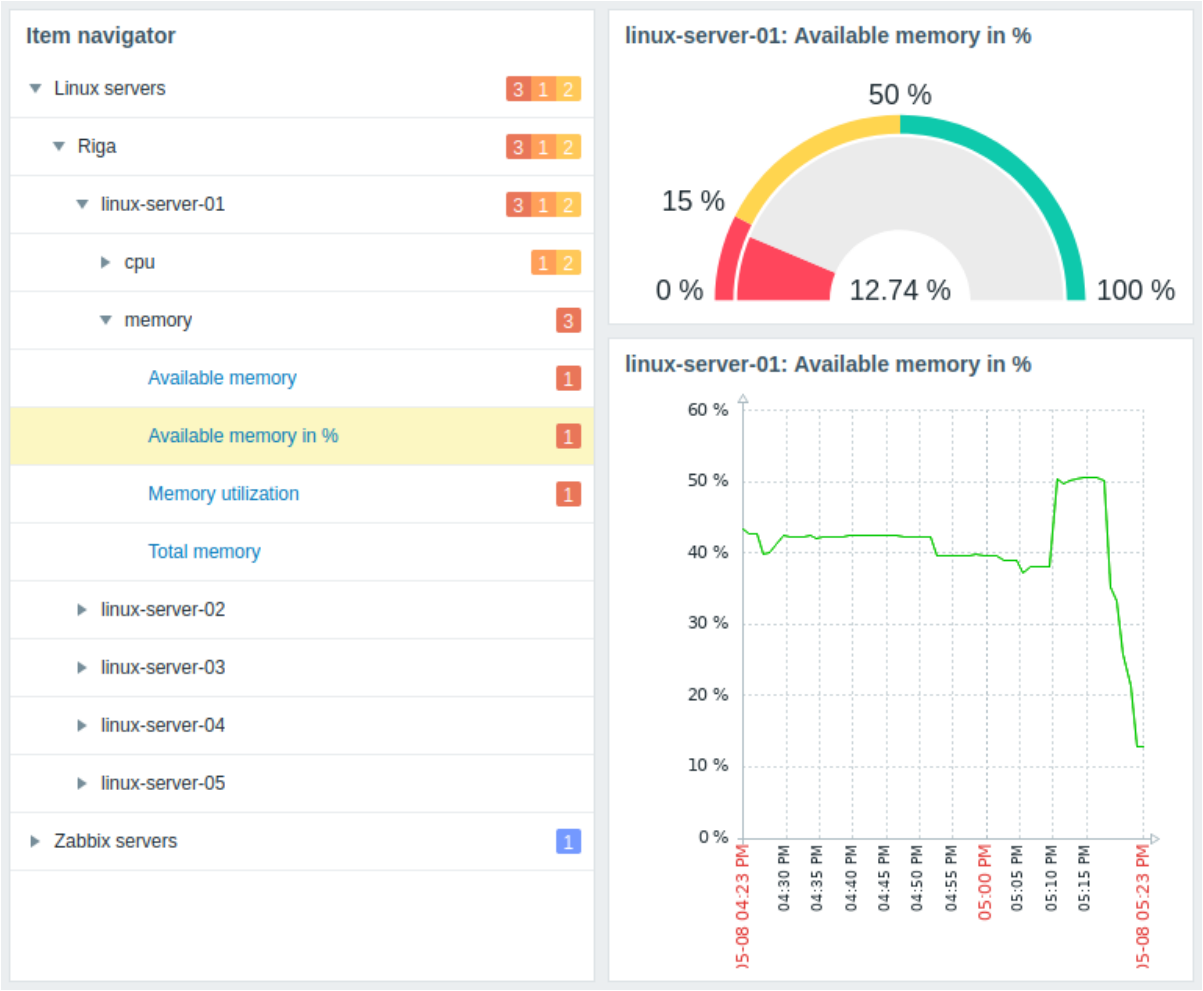
## 17 Навигатор элементов данных

### Обзор

Виджет навигатора по элементам данных отображает элементы данных на основе различных параметров фильтрации и группировки.




Этот виджет также позволяет управлять информацией, отображаемой в других виджетах, в зависимости от выбранного элемента данных.

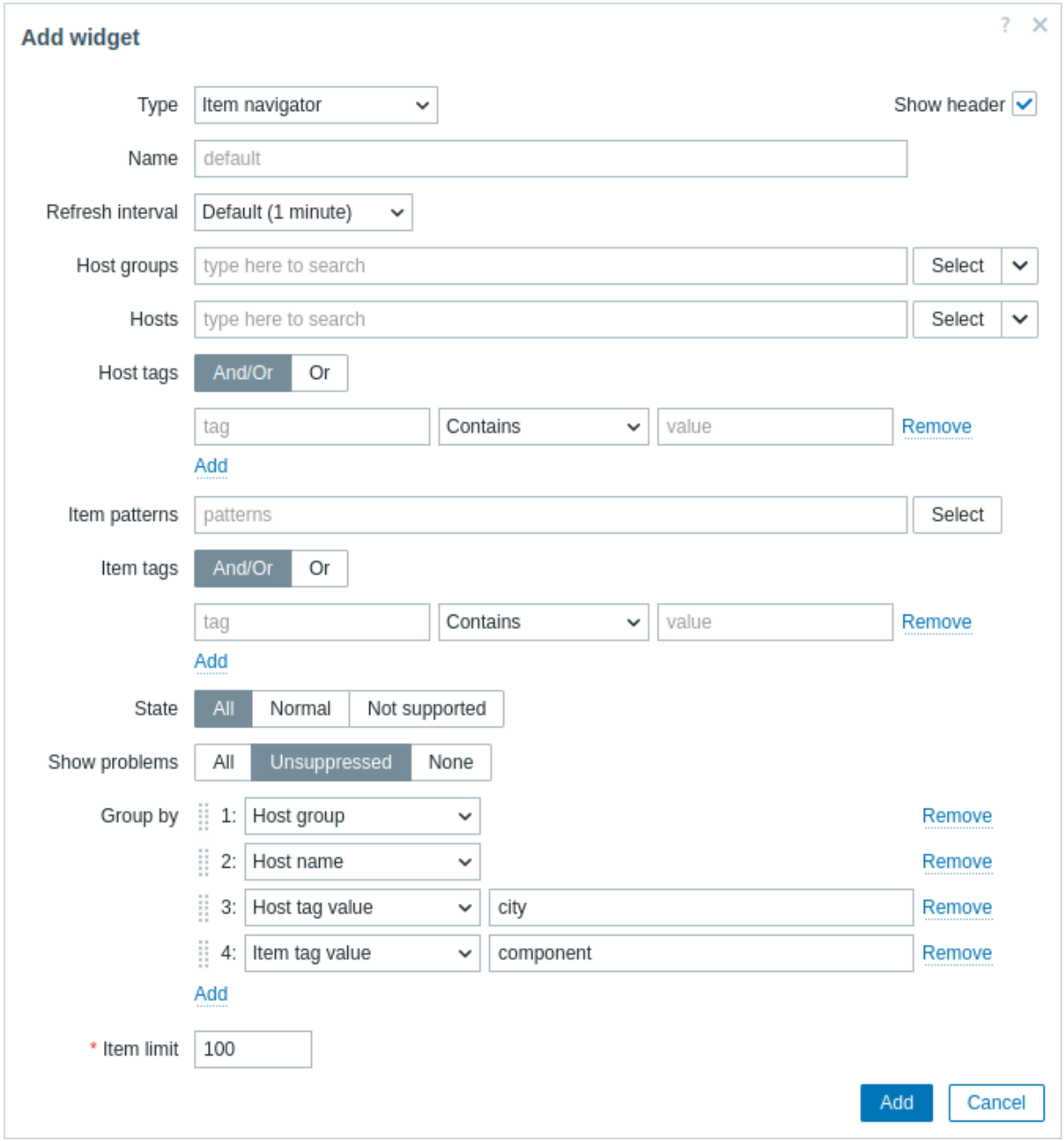



Группы, по которым организованы элементы данных, могут раскрываться или сворачиваться.

Для групп и проблем дополнительные сведения доступны при наведении курсора мыши.

Настройка

Для настройки выберите *Навигатор по элементам данных (Item navigator)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети; если не выбрана ни одна группа узлов сети, то виджет будет отображать элементы данных, относящиеся ко всем узлам сети из всех групп.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Узлы сети (Hosts)* | Выберите узлы сети.<br>В качестве альтернативы выберите для узлов сети в качестве источника данных совместимый виджет или панель.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени узла сети будет автоматически предложен список совпадающих узлов сети в выпадающем меню.<br>Если не выбран ни один узел сети, то виджет будет отображать элементы данных, относящиеся ко всем узлам сети.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |

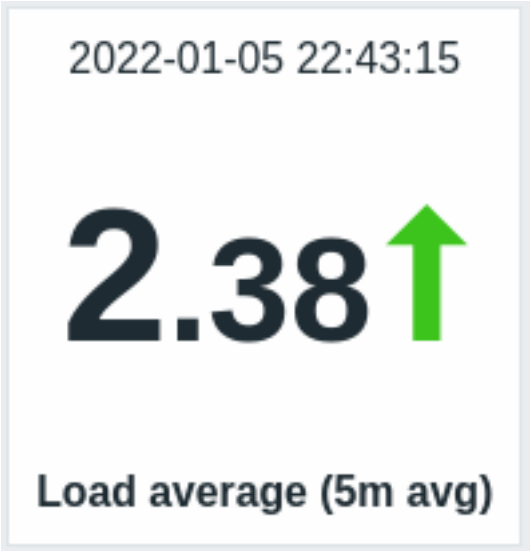
| | |
|---|---|
| *Теги узлов сети (Host tags)* | Укажите теги узлов сети для фильтрации элементов данных, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно задать несколько условий.<br>Имена тегов всегда сравниваются с учётом регистра.<br><br>Для каждого условия доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием ИЛИ;<br>**Или (Or)** — достаточно соблюдения одного условия.<br><br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Шаблоны элементов данных (Item patterns)* | Введите шаблоны элементов данных или выберите существующие элементы данных в качестве таких шаблонов. Элементы данных, совпадающие с указанными шаблонами, будут отображаться в навигаторе.<br><br>Можно использовать для выбора шаблоны с подстановочными символами (например, `*` вернёт элементы данных, которые соответствуют нулю или более символов; `Zabbix*` вернёт элементы данных, которые начинаются с «Zabbix»).<br>Чтобы указать шаблон с подстановочными символами, введите строку вручную и нажмите *Enter*. После начала ввода выпадающий список будет отображать совпадающие элементы данных, ограниченные теми, которые относятся к выбранным *Узлам сети* или же узлам сети из выбранных *Групп узлов сети*, если они имеются. Подстановочные символы всегда интерпретируются, поэтому нет возможности добавить, скажем, элемент данных с именем *item** отдельно, если имеются другие соответствующие элементы данных (например, item2, item3).<br><br>При настройке виджета на панели шаблона, этот параметр позволяет выбирать только элементы данных, настроенные в этом шаблоне. |
| *Теги элементов данных (Item tags)* | Укажите теги элементов данных для фильтрации элементов данных, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно задать несколько условий.<br>Имена тегов всегда сравниваются с учётом регистра.<br><br>Для каждого условия доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием ИЛИ;<br>**Или (Or)** — достаточно соблюдения одного условия. |
| *Состояние (State)* | Фильтровать отображаемые элементы данных по их состоянию (все, нормальные, неподдерживаемые). |

| | |
|---|---|
| *Проблемы (Show problems)* | Фильтровать отображаемые с элементам данных проблемы по их состоянию (все, неподавленные, никакие). |
| *Группировка по (Group by)* | Добавить атрибут группировки, по которому группировать выбранные элементы данных:<br>**Группа узлов сети (Host group)** — группировать элементы данных по группам их узлов сети;<br>**Имя узла сети (Host name)** — группировать элементы данных по их узлам сети;<br>**Значение тега узлов сети (Host tag value)** — введите имя тега для группировки элементов данных по значению этого тега узла сети (например, введите «city», чтобы группировать элементы данных по значениям «Riga», «Tokyo» и т.д.);<br>**Важность (Severity)** — группировать узлы сети по важности их проблем.<br>**Значение тега элементов (Item tag value)** — введите имя тега для группировки элементов данных по значению этого тега элементов данных (например, введите «component», чтобы группировать элементы данных по значениям «cpu», «memory» и т.д.).<br><br>Атрибуты группировки могут быть переупорядочены путём перетаскивания вверх или вниз за ползунок перед именем группы. Обратите внимание, что порядок атрибутов группировки определяет порядок вложенности групп. Например, указание нескольких имён тегов узов сети (1: цвет, 2: город) приведёт к тому, что элементы данных будут сгруппированы по цвету (красный, синий и т. д.), а затем по городу (Рига, Токио и т. д.).<br><br>В зависимости от настроенных атрибутов группировки элемент данных может отображаться в нескольких группах (например, при группировке по группам узлов сети, когда узел сети элемента данных принадлежит нескольким группам). Щелчок по таким элементам данных выделяет и подсвечивает их во всех группах.<br>Элементы данных, не соответствующие заданным атрибутам группировки, отображаются в группе *Без категории (Uncategorized)*.<br><br>Если в параметре *Проблемы (Show problems)* указано отображать проблемы, то они отображаются следующим образом:<br>- для каждой группы отображается общее количество проблем в нижележащих элементах данных;<br>- для каждого элемента данных отображается только его количество проблем.<br><br>Можно указать до 10 атрибутов группировки, и все они должны быть уникальными.<br>Если атрибуты группировки не указаны, элементы данных не будут сгруппированы. |
| *Лимит элементов данных (Item limit)* | Введите максимальное количество отображаемых элементов данных. Возможные значения: 1-9999.<br><br>Когда для отображения доступно больше элементов данных, чем задано лимитом, то ниже отображаемых элементов данных показывается соответствующее сообщение (например: «100 of 100+ items are shown (показаны 100 из 100+ элементов данных)»).<br>Обратите внимание, что настроенный лимит элементов данных также влияет на отображение настроенных групп; например, если лимит элементов данных составляет 100 и элементы данных группируются по их узлам сети (каждый содержит по 200 элементов данных), то в виджете будет отображён только первый узел сети со своими 100 элементов данных.<br><br>Этот параметр не зависит от параметра *Лимит элементов в результатах поиска и фильтрах (Limit for search and filter results)* в разделе *Administration (Администрирование)* → *Общие (General)* → *Веб-интерфейс (GUI)*. |

18 Значение элемента данных

Обзор

Этот виджет полезен для наглядного отображения значения одного элемента данных. Это может быть как последнее значение, так и агрегированное значение за некоторый период в прошлом.




Помимо самого значения, при необходимости можно отображать дополнительные элементы:

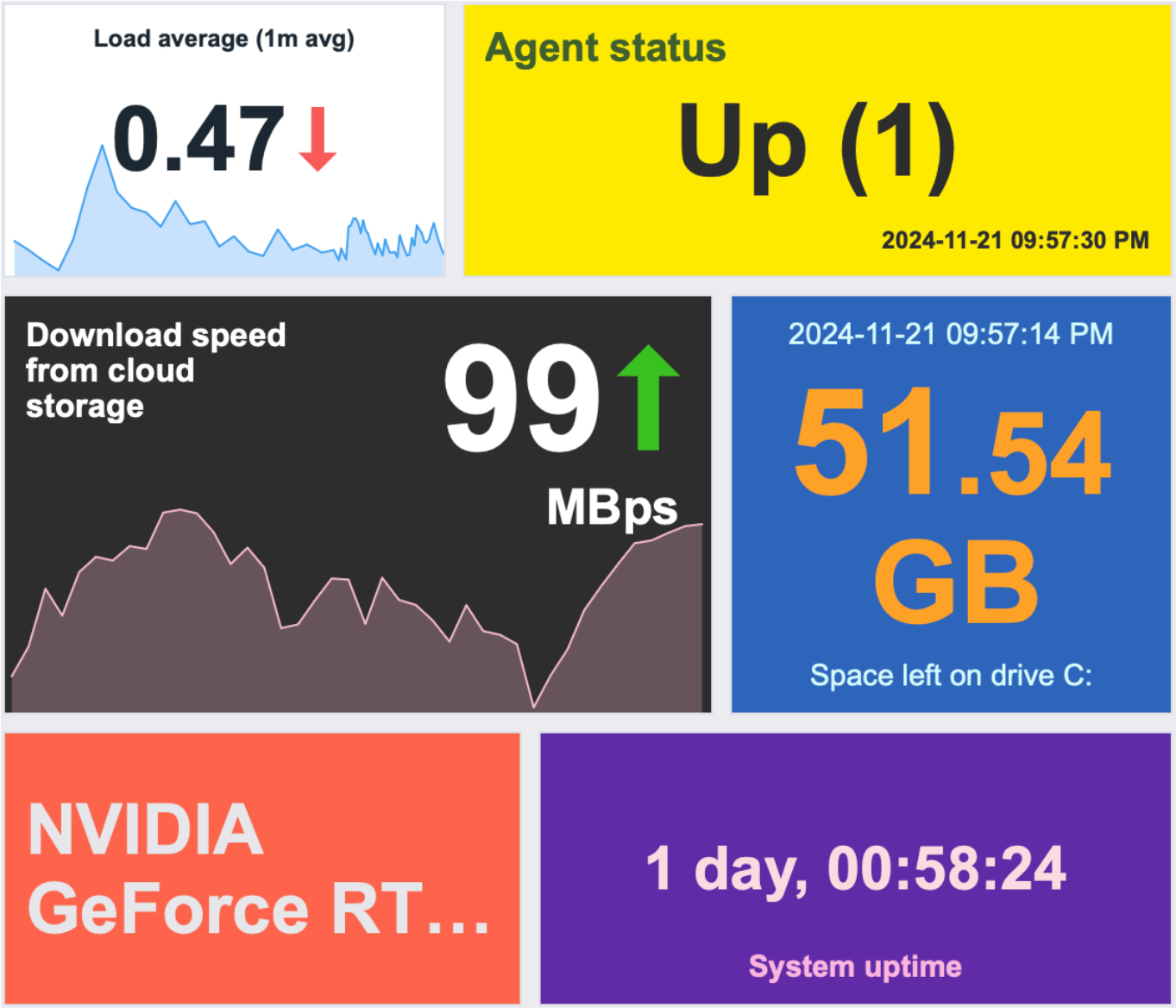
- Время метрики
- Описание элемента данных
- Индикатор изменения значения
- Цвет фона для значения
- Единица измерения элемента данных
- Мини-график значений за указанный период времени

Виджет может отображать числовые и строковые значения. Отображение двоичных значений не поддерживается. Строковые значения отображаются в одной строке и при необходимости обрезаются. Если значения нет, отображается ”No data”.

Индикатор изменения всегда сравнивает значение с тем же периодом в прошлом. Например, последнее значение будет сравниваться с предыдущим значением, а значение за последний месяц — с месяцем до него. Обратите внимание, что предыдущий период для агрегированных значений вычисляется как период той же длины, что и исходный, с временем окончания непосредственно перед временем начала исходного периода.

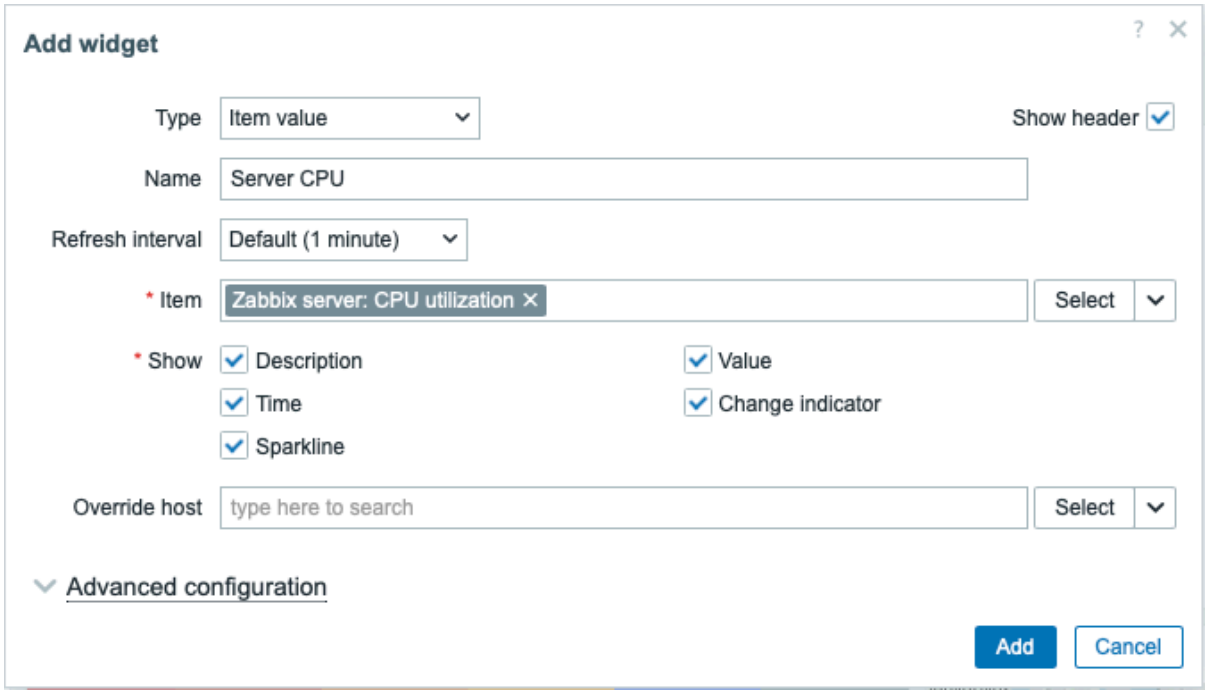
При нажатии на значение открывается график по запросу для числовых элементов данных или последние данные для строковых элементов данных.

Виджет и все его элементы можно визуально тонко настроить с помощью параметров расширенной настройки, что позволяет создавать широкий спектр визуальных стилей:




Конфигурация

Чтобы настроить, выберите *Item value* в качестве типа виджета:




Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Item* | Выберите элемент данных.<br>Либо выберите совместимый виджет в качестве источника данных для элемента данных. |
| *Show* | Установите флажок, чтобы отображать соответствующий элемент: description, value, time, change indicator, sparkline. Снимите флажок, чтобы скрыть. Должен быть выбран как минимум один элемент. |
| *Override host* | Выберите совместимый виджет или веб-интерфейс в качестве источника данных для узлов сети.<br>Этот параметр недоступен при настройке виджета на [шаблонной панели]((/manual/config/templates/template#adding-dashboards). |
| *Advanced configuration* | Щелкните заголовок *Advanced configuration*, чтобы отобразить параметры advanced configuration. |

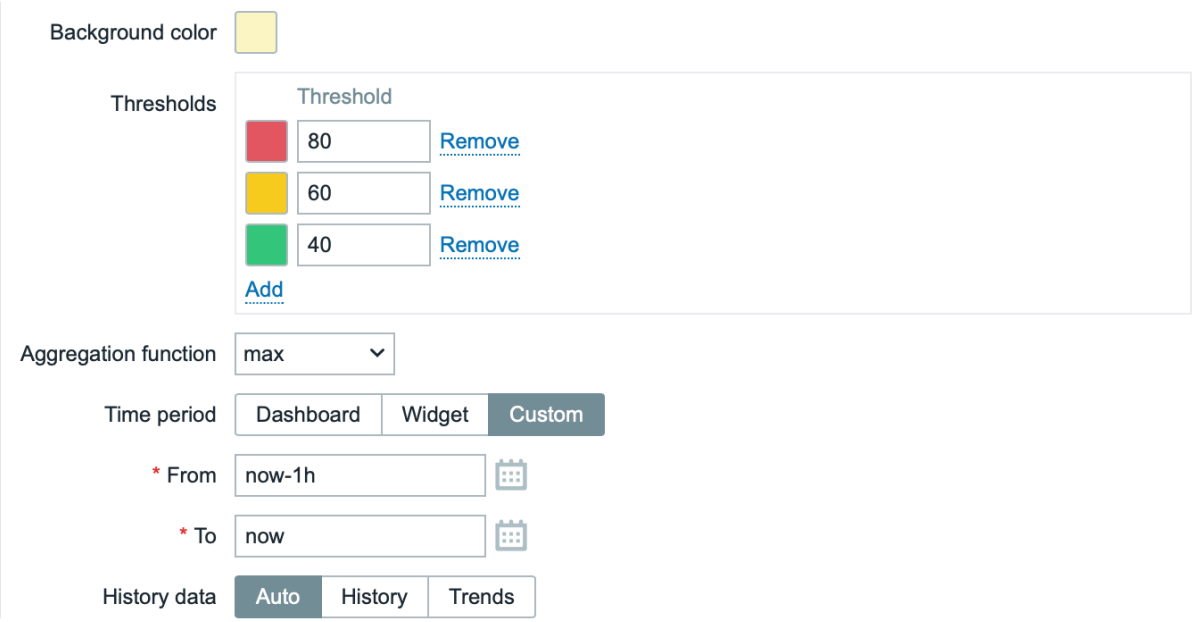
Расширенная конфигурация

Параметры расширенной конфигурации доступны в сворачиваемом разделе *Расширенная конфигурация*, и только для тех элементов, которые выбраны в поле *Показать* (см. выше).

Обратите внимание, что несколько элементов не могут занимать одно и то же пространство; если они размещены в одном пространстве, будет отображено сообщение об ошибке.

Общие параметры виджета

Эти параметры определяют цвет фона (статический или динамический) для всего виджета и функцию агрегации для отображения значений.

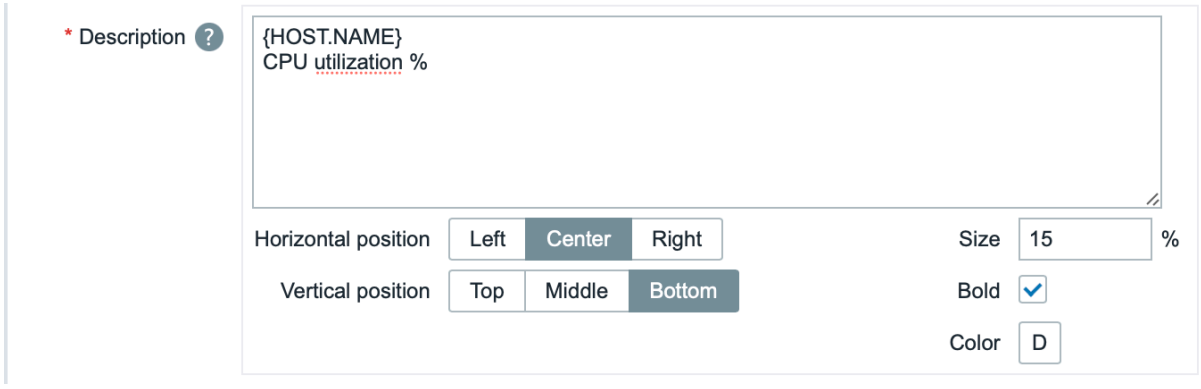



| | |
|---|---|
| *Цвет фона* | Выберите цвет фона для всего виджета в палитре цветов.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите кнопку *Использовать по умолчанию* в палитре цветов. |
| *Пороги* | Настройте динамический цвет фона для всего виджета. Нажмите *Добавить*, чтобы добавить порог, выберите цвет фона в палитре цветов и укажите числовое значение. Когда значение элемента данных станет равно пороговому значению или превысит его, цвет фона изменится.<br>При сохранении список будет отсортирован по возрастанию.<br>Обратите внимание, что динамический цвет фона корректно отображается только для числовых элементов данных. |

| | |
|---|---|
| *Функция агрегации* | Укажите, какую функцию агрегации использовать в пределах выбранного *Периода времени*:<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение;<br>**count** - отображать количество значений;<br>**sum** - отображать сумму значений;<br>**first** - отображать первое значение;<br>**last** - отображать последнее значение.<br>Если выбрано **не используется**, отображается самое последнее значение (без агрегации).<br><br>Агрегация позволяет отображать агрегированное значение за выбранный интервал (5 минут, час, день) вместо самого последнего значения.<br>Только числовые данные могут отображаться для *min*, *max*, *avg* и *sum*. Для *count* нечисловые данные будут преобразованы в числовые. |
| *Период времени* | Укажите период времени, который будет использоваться для агрегации значений:<br>**Dashboard** - использовать период времени панели;<br>**Widget** - использовать период времени указанного виджета;<br>**Custom** - использовать пользовательский период времени.<br>Этот параметр скрыт, когда для *Функция агрегации* выбрано "не используется". |
| *Виджет* | Выберите виджет.<br>Этот параметр отображается только если для *Период времени* выбрано "Widget". |
| *От* | Выберите начало периода времени (значение по умолчанию `now-1h`). См. синтаксис относительного времени.<br>Этот параметр отображается только если для *Период времени* выбрано "Custom". |
| *До* | Выберите конец периода времени (значение по умолчанию `now`). См. синтаксис относительного времени.<br>Этот параметр отображается только если для *Период времени* выбрано "Custom". |
| *Исторические данные* | Брать данные из истории или трендов:<br>**Auto** - автоматический выбор;<br>**History** - брать данные истории;<br>**Trends** - брать данные трендов.<br><br>Этот параметр применяется только к числовым данным. Нечисловые данные всегда будут браться из истории. |

Описание

Эти параметры определяют, как должно отображаться описание элемента данных.

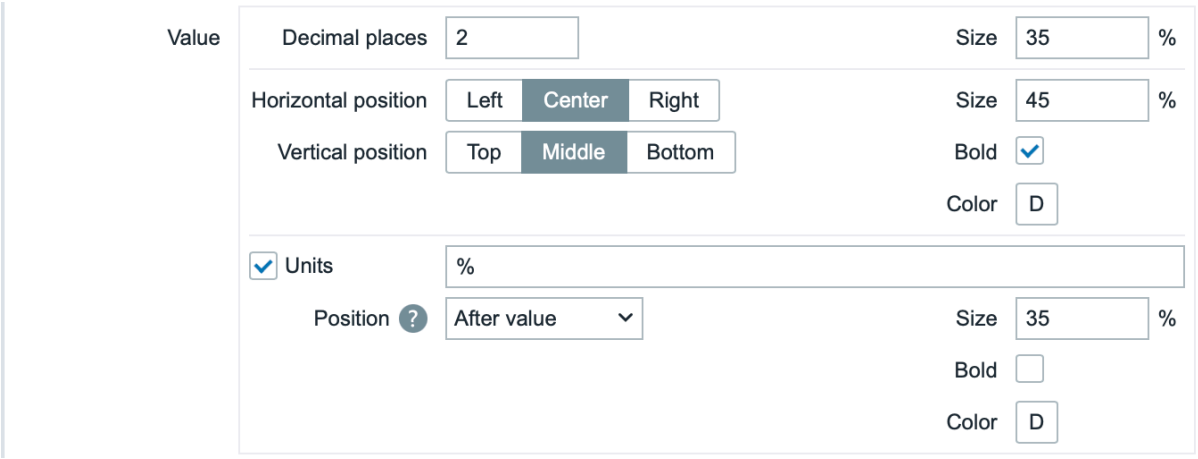



| | |
|---|---|
| *Описание* | Введите описание элемента данных. Это описание может переопределять имя элемента данных по умолчанию. Поддерживаются многострочные описания. Возможна комбинация текста и поддерживаемых макросов.<br>{HOST.*}, {ITEM.*}, {INVENTORY.*} и пользовательские макросы поддерживаются. |
| *Горизонтальное положение* | Выберите горизонтальное положение описания элемента данных — слева, справа или по центру. |
| *Вертикальное положение* | Выберите вертикальное положение описания элемента данных — сверху, снизу или по середине. |

| | |
|---|---|
| *Размер* | Введите высоту размера шрифта для описания элемента данных (в процентах относительно общей высоты виджета). |
| *Полужирный* | Установите флажок, чтобы отображать описание элемента данных полужирным шрифтом. |
| *Цвет* | Выберите цвет описания элемента данных в палитре цветов.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите кнопку *Использовать по умолчанию* в палитре цветов. |

Значение

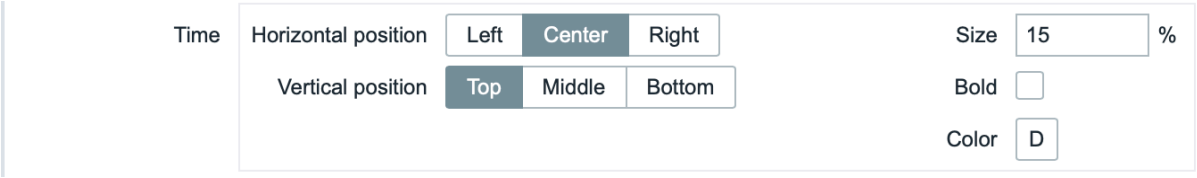
Эти параметры определяют, как должно отображаться значение элемента данных.




| | |
|---|---|
| *Десятичные знаки* | Выберите, сколько десятичных знаков будет отображаться вместе со значением. Этот параметр влияет только на элементы данных с плавающей точкой. Для элементов данных с единицами измерения, заданными как "s", если это значение больше 0 (по умолчанию 2), виджет округляет наиболее значимую единицу времени и отображает числовое значение (например, "10.43m") вместо преобразования значения в полную строку времени. Если установлено 0, секунды преобразуются в удобочитаемую строку времени (например, "4h 56m 30s"). |
| *Размер* | Введите высоту размера шрифта для десятичных знаков (в процентах от общей высоты виджета). |
| *Горизонтальное положение* | Выберите горизонтальное положение значения элемента данных - слева, справа или по центру. |
| *Вертикальное положение* | Выберите вертикальное положение значения элемента данных - сверху, снизу или по центру. |
| *Размер* | Введите высоту размера шрифта для значения элемента данных (в процентах от общей высоты виджета).<br>Обратите внимание, что размер значения элемента данных имеет приоритет; другим элементам необходимо уступить место для значения. Однако при наличии индикатора изменения, если значение слишком велико, оно будет обрезано, чтобы отобразить индикатор изменения. |
| *Жирный* | Установите флажок, чтобы отображать значение элемента данных полужирным шрифтом. |
| *Цвет* | Выберите цвет значения элемента данных в палитре цветов.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите кнопку *Использовать по умолчанию* в палитре цветов. |
| ***Единицы*** | |
| *Единицы* | Установите флажок, чтобы отображать единицы вместе со значением элемента данных. Если вы введете имя единицы, оно переопределит единицу из конфигурации элемента данных. |
| *Положение* | Выберите положение единицы элемента данных - над, под, перед или после значения. |
| *Размер* | Введите высоту размера шрифта для единицы элемента данных (в процентах от общей высоты виджета). |
| *Жирный* | Установите флажок, чтобы отображать единицу элемента данных полужирным шрифтом. |
| *Цвет* | Выберите цвет единицы элемента данных в палитре цветов.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуться к значению по умолчанию, нажмите кнопку *Использовать по умолчанию* в палитре цветов. |

Время

Эти параметры определяют, как должно отображаться время (значение часов из истории элемента данных).




| | |
|---|---|
| *Горизонтальное положение* | Выберите горизонтальное положение времени - слева, справа или по центру. |
| *Вертикальное положение* | Выберите вертикальное положение времени - сверху, снизу или по центру. |
| *Размер* | Введите высоту размера шрифта для времени (в процентах относительно общей высоты виджета). |
| *Полужирный* | Установите флажок, чтобы отображать время полужирным шрифтом. |
| *Цвет* | Выберите цвет времени в палитре цветов.<br>D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуть значение по умолчанию, нажмите кнопку *Использовать по умолчанию* в палитре цветов. |

Индикатор изменения

В этом разделе можно выбрать цвет индикаторов изменения с помощью палитры цветов.

Индикаторы изменения:

- ↑ - значение элемента данных увеличилось (для числовых элементов данных)
- ↓ - значение элемента данных уменьшилось (для числовых элементов данных)
- ↕ - значение элемента данных изменилось (для строковых элементов данных и элементов данных с преобразованием значений)

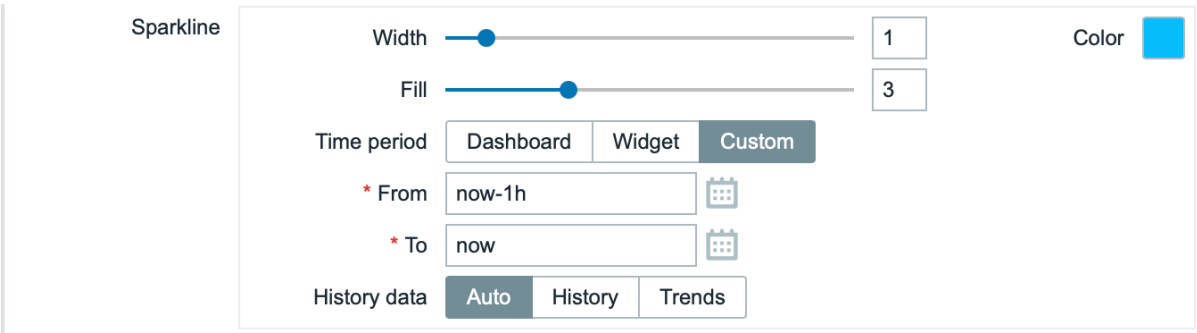
D означает цвет по умолчанию (зависит от темы веб-интерфейса). Чтобы вернуть значение по умолчанию, нажмите кнопку *Use default* в палитре цветов.

Вертикальный размер индикатора изменения равен размеру значения (целой части значения для числовых элементов данных).

Обратите внимание, что индикаторы увеличения и уменьшения не отображаются при наличии только одного значения.

Sparkline

Эти параметры определяют, как должен отображаться график sparkline.




| | |
|---|---|
| *Width* | Задайте толщину линии графика с помощью ползунка или вручную введите значение в диапазоне от 0 до 10. |
| *Color* | Выберите цвет линии и заливки. |
| *Fill* | Задайте уровень прозрачности цвета заливки с помощью ползунка или вручную введите значение в диапазоне от 0 до 10. |
| *Time period* | Укажите период времени, значения за который будут включены в график sparkline:<br>**Dashboard** - использовать период времени панели;<br>**Widget** - использовать период времени указанного виджета;<br>**Custom** - использовать пользовательский период времени. |

| | |
|---|---|
| *Widget* | Выберите виджет.<br>Этот параметр будет отображаться только если для *Time period* установлено значение "Widget". |
| *From* | Выберите начальный момент периода времени (значение по умолчанию `now-1h`). См. синтаксис относительного времени.<br>Этот параметр будет отображаться только если для *Time period* установлено значение "Custom". |
| *To* | Выберите конечный момент периода времени (значение по умолчанию `now`). См. синтаксис относительного времени.<br>Этот параметр будет отображаться только если для *Time period* установлено значение "Custom". |
| *History data* | Брать данные из history или trends:<br>**Auto** - автоматический выбор;<br>**History** - брать данные history;<br>**Trends** - брать данные trends. |

19 Карта

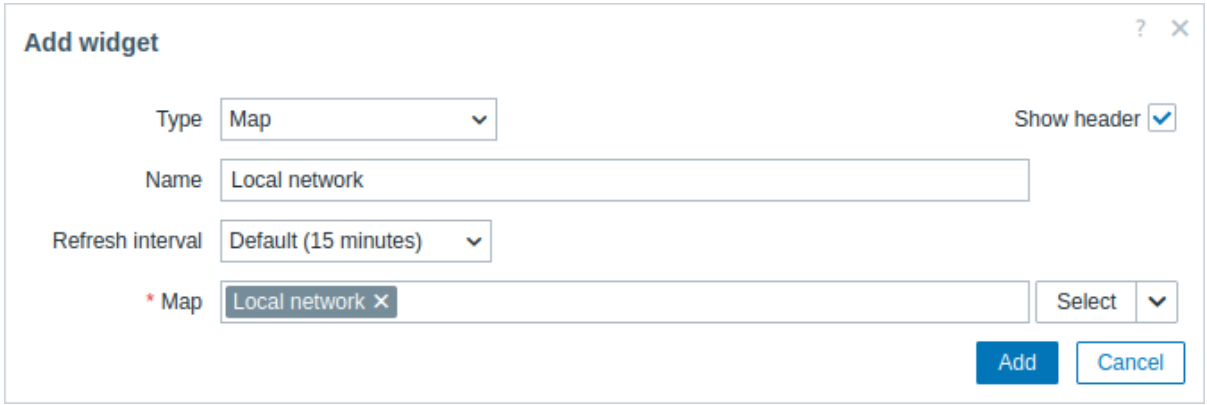
Обзор

В виджете карты можно отображать либо:

- одну настроенную карту сети;
- одну из настроенных карт сети в дереве навигации по картам (при нажатии на имя карты в дереве).

Настройка

Для настройки выберите *Карта сети (Map)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

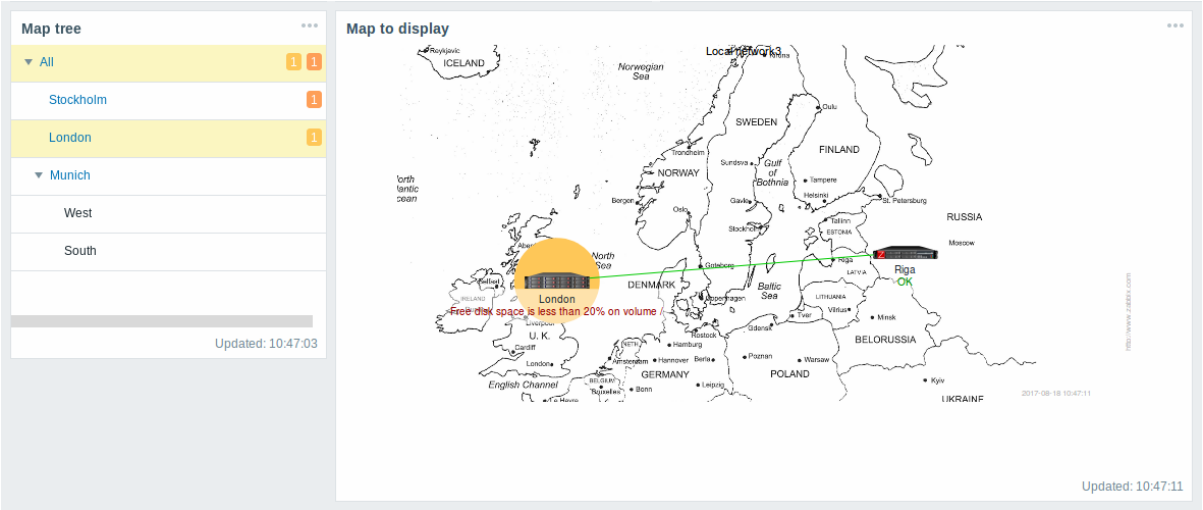
| | |
|---|---|
| *Карта сети (Map)* | Задайте карту для отображения.<br>В качестве альтернативы выберите для отображаемой карты сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени карты или виджета будет автоматически предложен список совпадающих карт сети или виджетов в выпадающем меню. |

20 Дерево навигации по карте

Обзор

Этот виджет позволяет строить иерархию существующих карт, а также отображать статистику проблем для каждой включенной карты и группы карт.

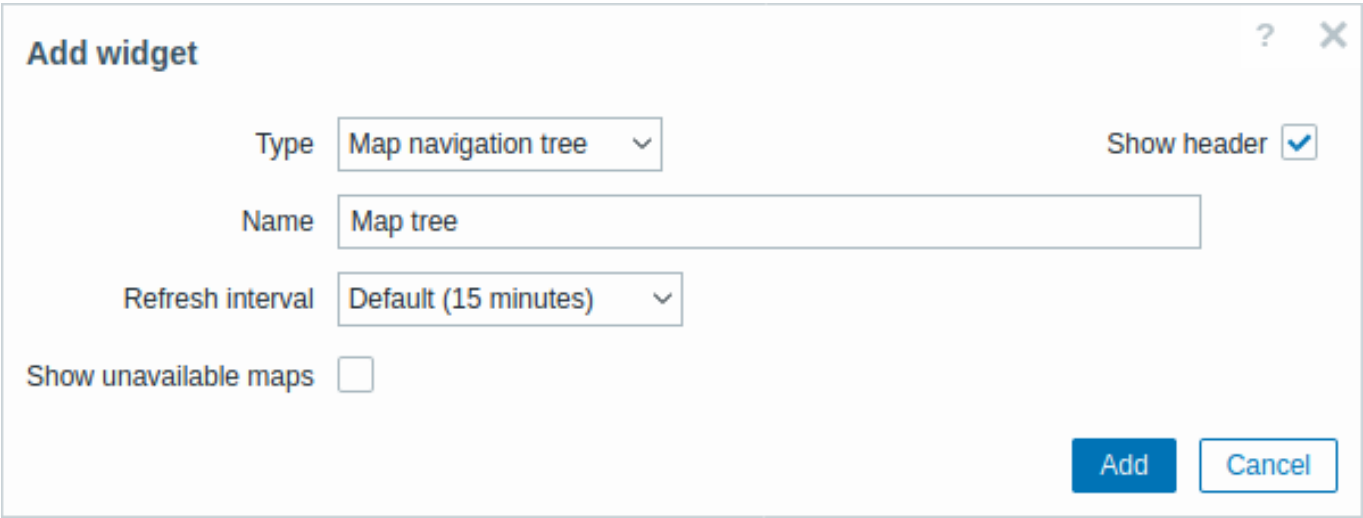
Он становится еще более мощным, если связать виджет *Map* с деревом навигации. В этом случае при щелчке по названию карты в дереве навигации карта полностью отображается в виджете *Map*.




Статистика для карты верхнего уровня в иерархии отображает сумму проблем всех подкар и собственных проблем.

Настройка

Для настройки выберите *Дерево навигации карт сетей (Map navigation tree)* в качестве типа:

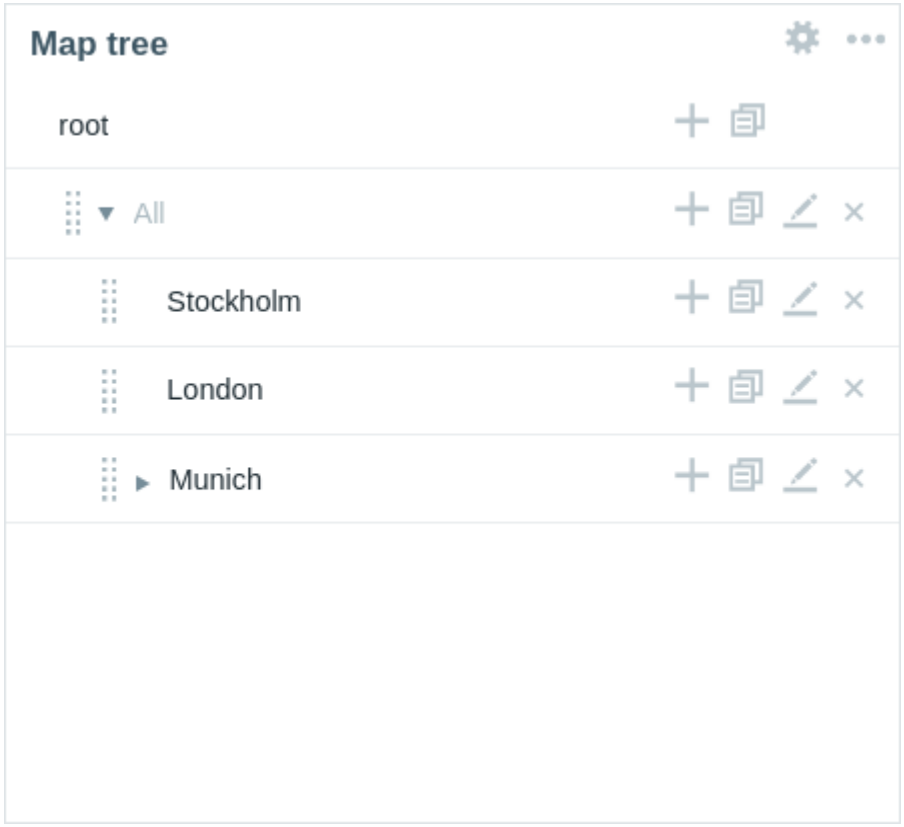



В дополнение к параметрам, которые являются общими для всех виджетов, вы можете выставить следующие специфичные опции:

| | |
|---|---|
| *Отображать недоступные карты (Show unavailable maps)* | Установите этот флажок для отображения карт, на которые у пользователя нет прав на чтение.<br>Недоступные карты в дереве навигации будут отображаться серым значком.<br>Обратите внимание что если этот флажок отмечен, доступные карты нижнего уровня отображаются, даже если карта вышестоящего уровня недоступна. Если флажок снят, доступные карты нижних уровней при недоступности карт вышестоящих уровней вообще не будут отображаться.<br>Количество проблем рассчитывается на основе доступных карт и доступных элементов карты. |

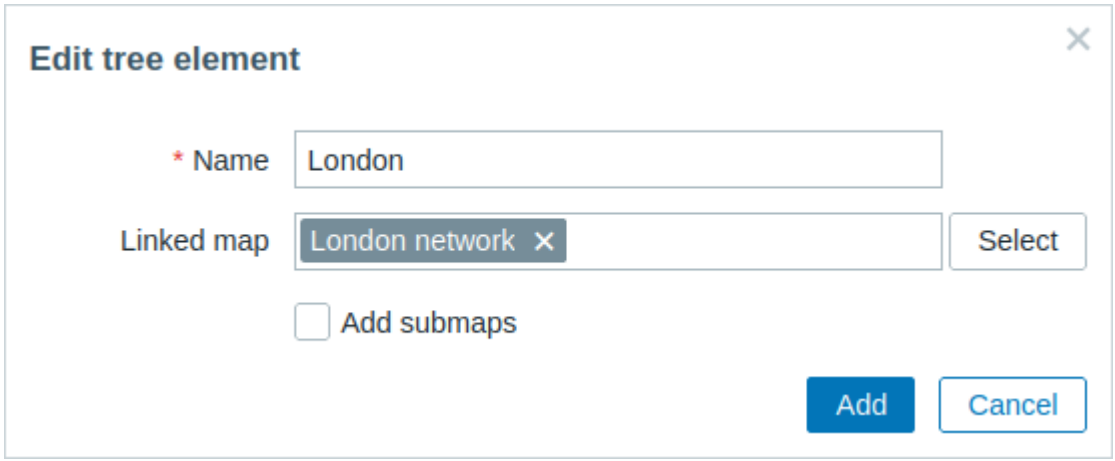
Элементы дерева навигации отображаются в виде списка. Вы можете:

- перетащить элемент (включая его дочерние элементы) на новое место в списке;
- развернуть или свернуть элемент, чтобы показать или скрыть его дочерние элементы;
- добавить к элементу дочерний элемент (со связанной картой или без нее);
- добавить к элементу несколько дочерних элементов (со связанными картами);
- редактировать элемент;
- удалить элемент (включая его дочерние элементы).




Настройка элемента

Чтобы настроить элемент дерева навигации, добавьте новый элемент или отредактируйте существующий элемент.



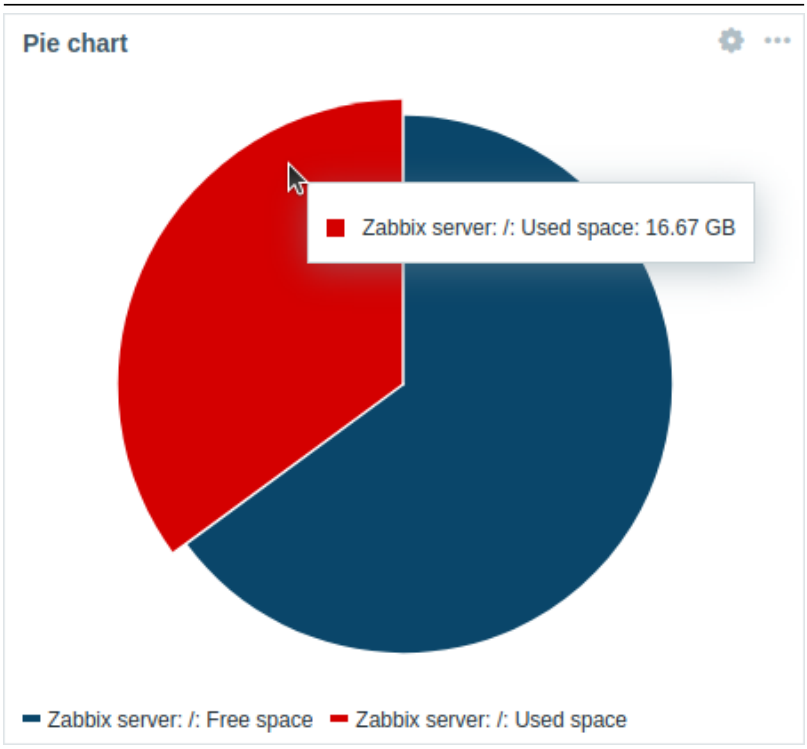

Доступны следующие параметры конфигурации элемента дерева навигации:

| | |
|---|---|
| *Имя (Name)* | Введите имя элемента дерева навигации. |
| *Связанная карта (Linked map)* | Выберите карту, которая будет связана с элементом дерева навигации.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени карты сети будет автоматически предложен список совпадающих карт в выпадающем меню. |
| *Добавить подкарты (Add submaps)* | Отметьте этот флажок, чтобы добавить к элементу дерева навигации подкарты связанной карты в качестве дочерних элементов. |

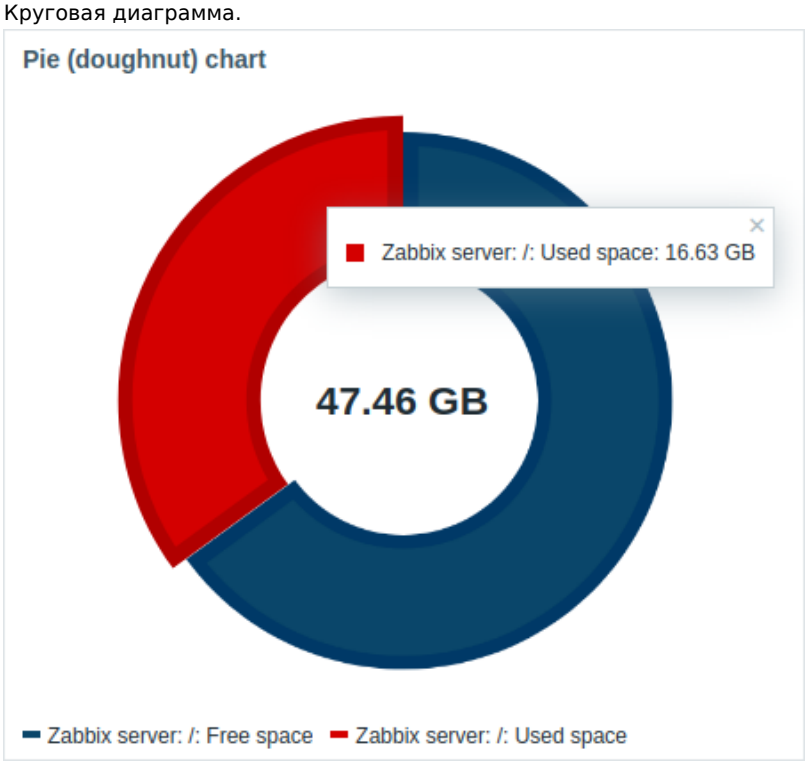
21 Круговая диаграмма

Обзор

Виджет круговой диаграммы позволяет отображать значения выбранных элементов данных в виде диаграммы **pie** или **doughnut**.




Круговая диаграмма.




Кольцевая диаграмма.

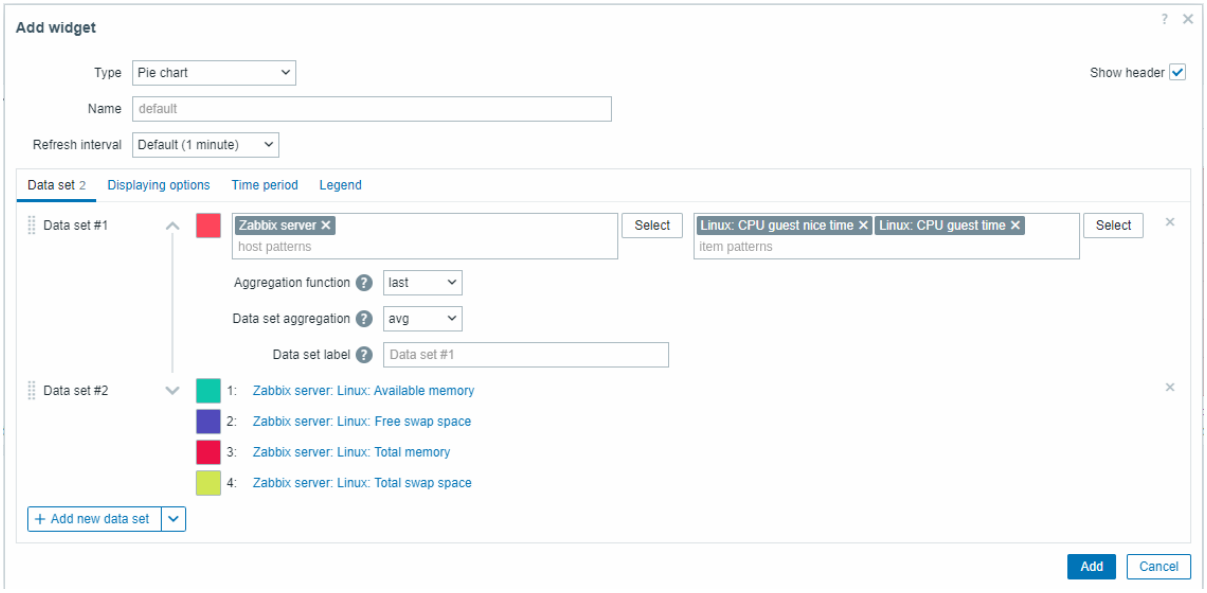
При наведении курсора выделенный сектор увеличивается наружу, и для этого сектора отображается легенда. При щелчке по выделенному сектору эффект выдвижения сохраняется до закрытия с помощью ”x”.

Диаграммы рисуются с использованием техники векторной отрисовки изображений.

Информацию, отображаемую в виджете *Pie chart*, можно загрузить в виде изображения PNG, выбрав параметр *Download image* в меню виджета.

Настройка

Для настройки выберите *Круговая диаграмма (Pie chart)* в качестве типа:




Набор данных

Вкладка **Набор данных** позволяет выбирать данные для круговой диаграммы путем добавления наборов данных. Можно добавить два типа наборов данных:

- *Шаблоны элементов данных* - отображаются данные из совпадающих элементов данных. Можно выбрать один базовый цвет или строку палитры, чтобы назначить разные цвета каждому совпавшему элементу данных.
- *Список элементов данных* - отображаются данные из выбранных элементов данных. Для каждого элемента можно выбрать цвет отдельно с помощью палитры.

По умолчанию добавляется набор данных типа *Шаблоны элементов данных*.

| | |
|---|---|
| *Набор данных* | Для набора данных **Шаблоны элементов данных**:<br>Выберите или введите шаблоны узла сети и элемента данных; на круговой диаграмме будут отображаться данные элементов, которые соответствуют введенным шаблонам; может быть отображено до 50 элементов.<br>Для выбора можно использовать шаблоны с подстановочными знаками (например, * вернет результаты, соответствующие нулю или более символам).<br>Чтобы указать шаблон с подстановочным знаком, введите строку вручную и нажмите *Enter*.<br>Символ подстановки всегда интерпретируется, поэтому нельзя добавить, например, элемент с именем *item** отдельно, если есть другие совпадающие элементы (например, *item2*, *item3*).<br>Указание шаблонов узла сети и элемента данных обязательно для наборов данных типа ”Шаблоны элементов данных”.<br>См. также: Подробности настройки набора данных.<br><br>Для набора данных **Список элементов данных**:<br>Выберите элементы для круговой диаграммы, нажав кнопку *Добавить элемент*.<br>Выпадающий список *Тип* после имени элемента позволяет выбрать тип отображения для каждого элемента:<br>**Normal** - значение элемента отображается пропорционально на круговой диаграмме (по умолчанию);<br>**Total** - значение элемента занимает всю круговую диаграмму. Обратите внимание, что на одной круговой диаграмме может существовать только один элемент типа ”Total”, и он будет первым в легенде круговой диаграммы. Если для элемента задано значение ”Total”, параметр *Агрегация набора данных* (см. ниже) будет отключен и установлен в значение ”not used”.<br>Также можно выбрать совместимые виджеты в качестве источника данных для элементов, нажав кнопку *Добавить виджет*. К виджетам применяются те же параметры, что и к отдельным элементам.<br>Указание элементов или виджетов обязательно для наборов данных типа ”Список элементов данных”.<br>См. также: Подробности настройки набора данных.<br><br>Обратите внимание, что допускаются только числовые типы элементов данных.<br><br>При настройке виджета на панели мониторинга шаблона параметр для указания шаблонов узла сети недоступен, а параметр для указания списка элементов данных позволяет выбрать только элементы данных, настроенные на шаблоне. |
| *Функция агрегации* | Укажите, какую функцию агрегации использовать для каждого элемента в наборе данных в пределах выбранного *Периода времени*:<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение;<br>**sum** - отображать сумму значений;<br>**count** - отображать количество значений;<br>**first** - отображать первое значение;<br>**last** - отображать последнее значение (по умолчанию).<br><br>Агрегация позволяет отображать агрегированное значение за выбранный интервал (5 минут, час, день).<br>Только числовые данные могут отображаться для *min*, *max*, *avg* и *sum*. Для *count* нечисловые данные будут преобразованы в числовые. |

| | |
|---|---|
| *Агрегация набора данных* | Укажите, какую функцию агрегации использовать для всего набора данных:<br>**not used** - агрегация не используется, элементы отображаются отдельно (по умолчанию);<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение;<br>**sum** - отображать сумму значений;<br>**count** - отображать количество значений.<br><br>Агрегация позволяет отображать агрегированное значение за интервал (5 минут, час, день), выбранный на вкладке *Период времени*, или используемый для всей панели мониторинга. |
| *Метка набора данных* | Укажите пользовательскую метку для набора данных.<br>Метка отображается в настройках набора данных и в легенде круговой диаграммы (для агрегированных наборов данных).<br>Все наборы данных нумеруются, включая те, для которых указана *Метка набора данных*. Если метка не указана, набор данных будет автоматически помечен в соответствии со своим номером (например, ”Data set #2”, ”Data set #3” и т. д.).<br>Нумерация наборов данных пересчитывается после изменения порядка/перетаскивания наборов данных.<br>Слишком длинные метки наборов данных будут сокращены, чтобы поместиться в месте отображения (например, ”Number of proc...”). |

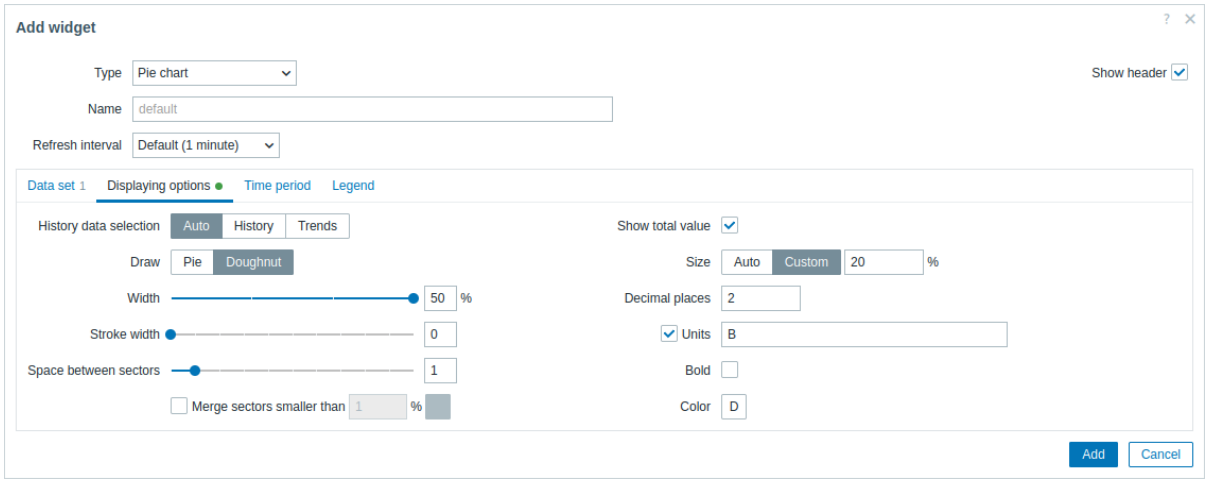
Подробности настройки набора данных

Существующие наборы данных отображаются в виде списка. Вы можете переупорядочить, развернуть/свернуть, изменить цвет и склонировать эти наборы данных.

Для дополнительных сведений смотрите подробности настройки набора данных в виджете *График (Graph)*. Эти подробности применимы также и к виджету *Круговая диаграмма (Pie chart)*.

Параметры отображения

Вкладка **Параметры отображения** позволяет задать параметры выбора данных истории и визуализации для круговой диаграммы:

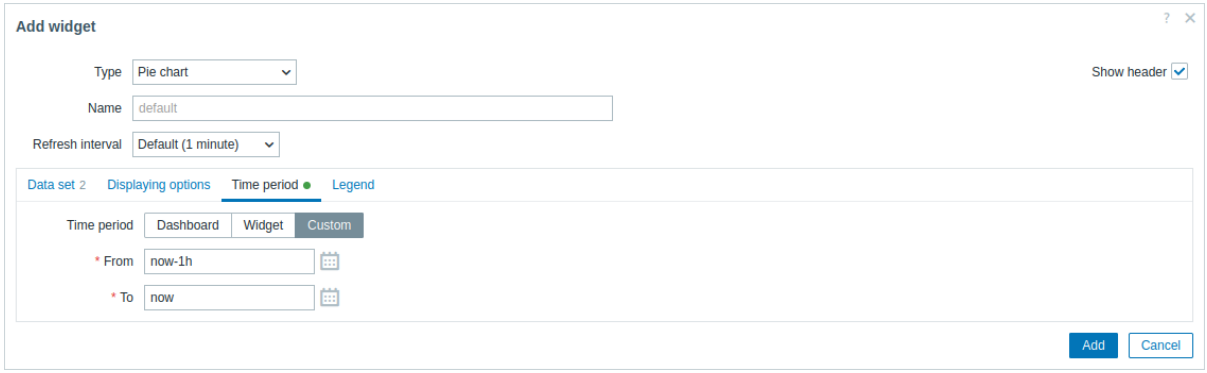



| | |
|---|---|
| *Выбор данных истории* | Выберите источник данных:<br>**Auto** - данные берутся согласно классическому алгоритму (по умолчанию);<br>**History** - данные из истории;<br>**Trends** - данные из трендов. |
| *Draw* | Выберите стиль отображения круговой диаграммы:<br>**Pie** - полная круговая диаграмма (секторы занимают 100% радиуса);<br>**Doughnut** - круговая диаграмма с пустым кругом в центре (секторы занимают до 50% радиуса). |

| | |
|---|---|
| *Space between sectors* | Выберите размер промежутка (в единицах 0-10) между секторами (по умолчанию "1"). |
| *Merge sectors smaller than N%* | Установите флажок, чтобы объединять сектора меньше N%.<br>Если включено, выберите цвет для объединенных секторов и пороговое значение процента (N) для объединения мелких секторов. |
| *Draw style:* ***Doughnut*** | |
| *Width* | Выберите ширину doughnut: 20, 30, 40 или 50% (по умолчанию) радиуса. |
| *Stroke width* | Выберите ширину границы сектора doughnut (0-10). |
| *Show total value* | Установите флажок, чтобы отображать итоговое значение в центре диаграммы doughnut. |
| *Size* | Выберите параметр размера для итогового значения:<br>**Auto** - размер текста подбирается автоматически, чтобы итоговое значение в центре doughnut читалось удобно;<br>**Custom** - укажите размер текста как процент высоты от общей высоты виджета. |
| *Decimal places* | Укажите количество знаков после запятой для итогового значения (0-6). |
| *Units* | Укажите единицы для итогового значения. |
| *Bold* | Установите флажок, чтобы отображать итоговое значение полужирным шрифтом. |
| *Color* | Выберите цвет для итогового значения. |

Период времени

Вкладка **Период времени** позволяет задать пользовательский период времени для настроек агрегации круговой диаграммы:

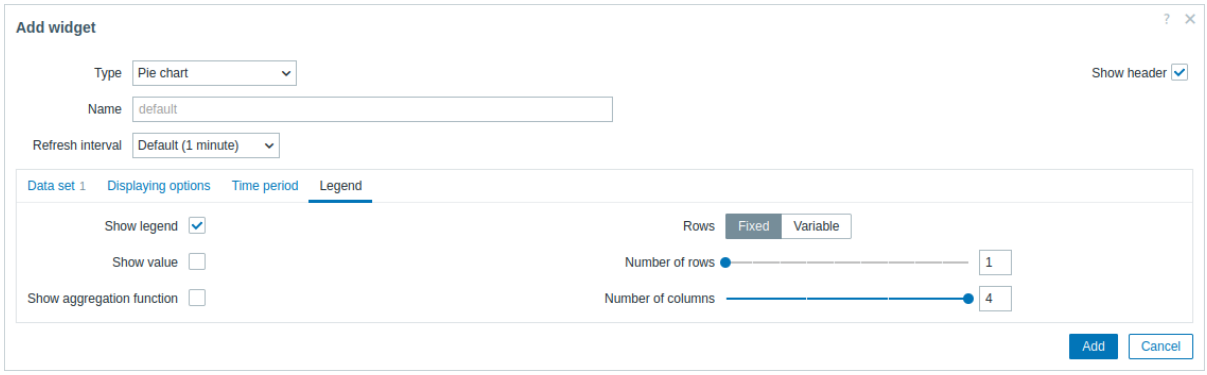



| | |
|---|---|
| *Период времени* | Выберите источник данных для периода времени:<br>**Панель** - задает селектор *Период времени* в качестве источника данных;<br>**Виджет** - задает совместимый виджет, указанный в параметре *Виджет*, в качестве источника данных;<br>**Пользовательский** - задает период времени, указанный в параметрах *С* и *По*, в качестве источника данных; если задан, в правом верхнем углу виджета будет отображаться значок часов, указывающий установленное время при наведении курсора. |
| *Виджет* | Введите или выберите совместимый виджет (*Graph*, *Graph (classic)*, *Graph prototype*) в качестве источника данных для периода времени.<br>Этот параметр доступен, если для *Период времени* установлено значение "Виджет". |
| *С* | Введите или выберите начало периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w`, и т. д.).<br>Этот параметр доступен, если для *Период времени* установлено значение "Пользовательский". |

| | |
|---|---|
| *По* | Введите или выберите конец периода времени.<br>Поддерживается синтаксис относительного времени (now, now/d, now/w-1w, и т. д.).<br>Этот параметр доступен, если для *Период времени* установлено значение "Пользовательский". |

Легенда

Вкладка **Легенда (Legend)** позволяет настроить легенду круговой диаграммы:



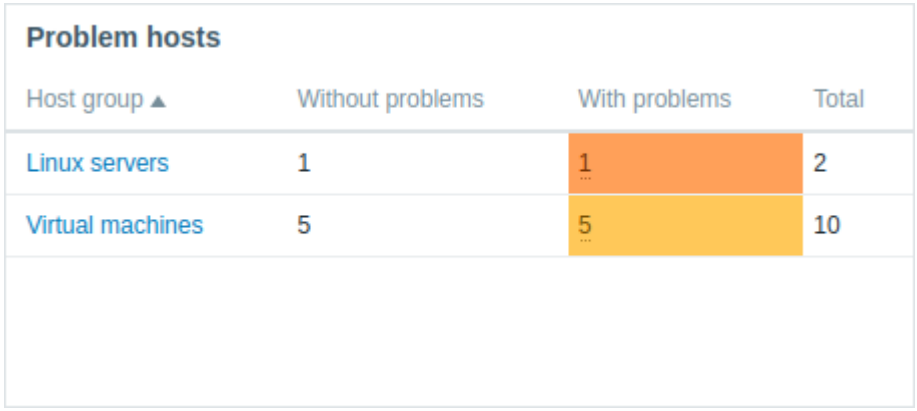

| | |
|---|---|
| *Легенда (Show legend)* | Снимите этот флажок, чтобы скрыть легенду на графике (по умолчанию флажок отмечен). |
| *Отображение значения (Show value)* | Установите этот флажок для отображения в легенде значения элемента данных. |
| *Отображение функции агрегирования (Show aggregation function)* | Установите этот флажок для отображения в легенде функции агрегирования. |
| *Количество строк (Rows)* | Выберите режим отображения строк легенды:<br>**Фиксированное (Fixed)** — количество отображаемых строк определяется значением параметра *Количество строк (Number of rows)*;<br>**Переменное (Variable)** — количество отображаемых строк определяется количеством настроенных элементов данных, в то же время не превышая значения параметра *Максимальное количество строк (Maximum number of rows)*. |
| *Количество строк/ Максимальное количество строк (Number of rows/ Maximum number of rows)* | Если *Количество строк (Rows)* имеет значение «Фиксированное (Fixed)», задать количество отображаемых строк легенды (1-10).<br>Если *Количество строк (Rows)* имеет значение «Переменное (Variable)», задать максимальное количество отображаемых строк легенды (1-10). |
| *Количество колонок (Number of columns)* | Задать количество отображаемых столбцов легенды (1-4).<br>Этот параметр доступен, если *Отображение значения (Show value)* не отмечено. |

22 проблемных узла сети

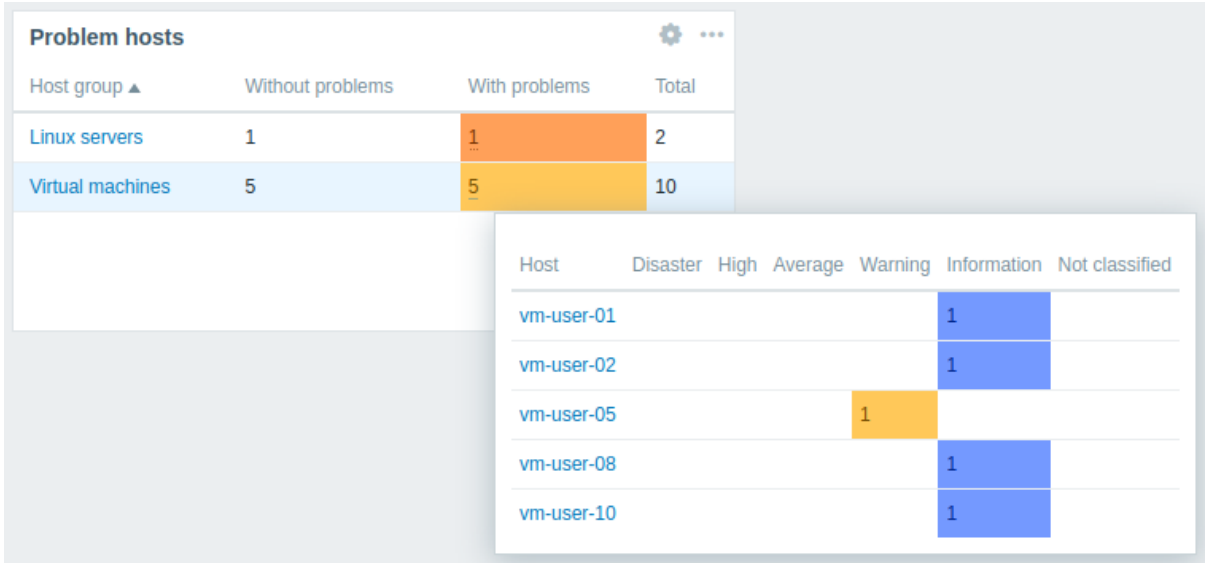
Обзор

В виджете узлов сети с проблемами вы можете отобразить количество проблем по группам узлов сети и наивысшую важность проблем в группе.

Количество проблем отображается только для проблем, являющихся причинами.

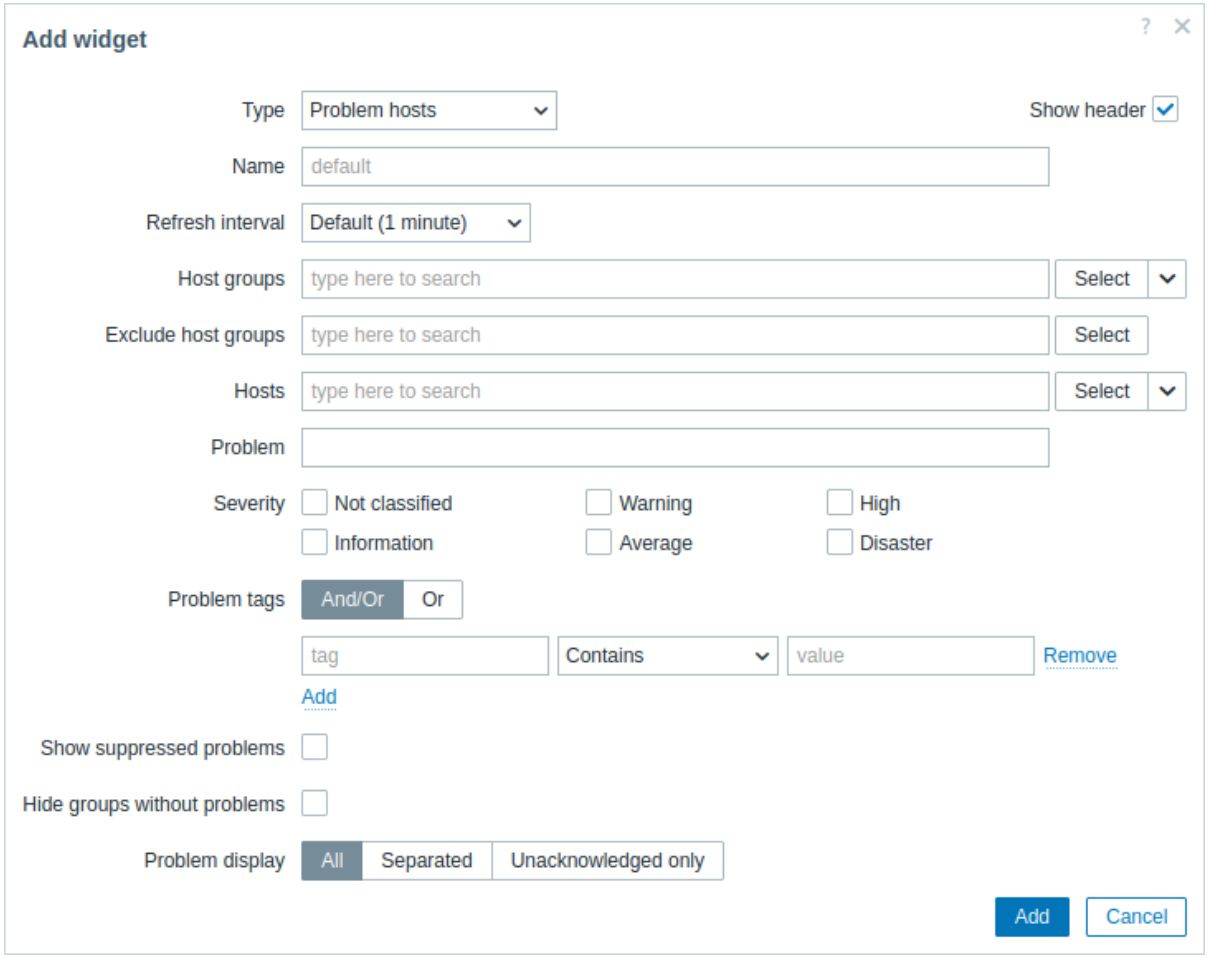



При наведении курсора мыши на количество проблем отображается список узлов сети, имеющих проблемы:




Настройка

Для настройки выберите *Узлы сети с проблемами (Problem hosts)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети для отображения в виджете.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети.<br>Данные от этих групп узлов сети будут отображаться в виджете; если не выбрана ни одна группа, то будут отображаться все группы узлов сети.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Исключить группы узлов сети (Exclude host groups)* | Выберите группы узлов сети, которые нужно скрыть от виджета.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети.<br>Данные узлов сети из этих групп не будут отображаться в виджете. Например, узлы сети 001, 002, 003 могут быть в группе «Group A», а узлы сети 002, 003 — также и в группе «Group B». Если мы одновременно выбираем *показать* группу «Group A» и *исключить* группу «Group B», то на панели будут отображены данные только с узла сети 001.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Узлы сети (Hosts)* | Выберите узлы сети для отображения в виджете.<br>В качестве альтернативы выберите для узлов сети в качестве источника данных совместимый виджет или панель.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени узла сети будет автоматически предложен список совпадающих узлов сети в выпадающем меню.<br>Если не введено никаких узлов сети, будут отображаться все узлы сети.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |

| | |
|---|---|
| *Проблема (Problem)* | Количество отображаемых узлов сети с проблемами можно ограничить по имени проблемы.<br>Если ввести здесь строку, будут отображаться только те узлы сети, чьи имена проблем содержат введённую строку.<br>Макросы не раскрываются. |
| *Важность (Severity)* | Укажите важности проблем, чтобы отфильтровать проблемы для отображения в виджете.<br>Если никакие важности не отмечены, будут отображаться все проблемы. |
| *Теги проблем (Problem tags)* | Укажите теги проблем для ограничения количества проблем, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно указать несколько условий. Поиск соответствия имён тегов всегда чувствителен к регистру.<br><br>По каждому условию доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием *Или (Or)*;<br>**Или (Or)** — достаточно соблюдения одного условия. |
| *Подавленные проблемы (Show suppressed problems)* | Установите флажок для отображения проблем, которые в противном случае могут быть подавлены (не показаны) по причине режима обслуживания узла сети. |
| *Скрыть группы без проблем (Hide groups without problems)* | Отметьте опцию *Скрыть группы без проблем (Hide groups without problems)*, чтобы скрыть в виджете данные от групп узлов сети, в которых проблемы отсутствуют.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Отображение проблем (Problem display)* | Отображать количество проблем как:<br>**Все (All)** — будет отображаться общее количество проблем;<br>**Раздельно (Separated)** — количество неподтвержденных проблем будет отображаться отдельным числом от общего количества проблем;<br>**Только неподтверждённые (Unacknowledged only)** — будет отображаться только количество неподтверждённых проблем. |

23 проблемы

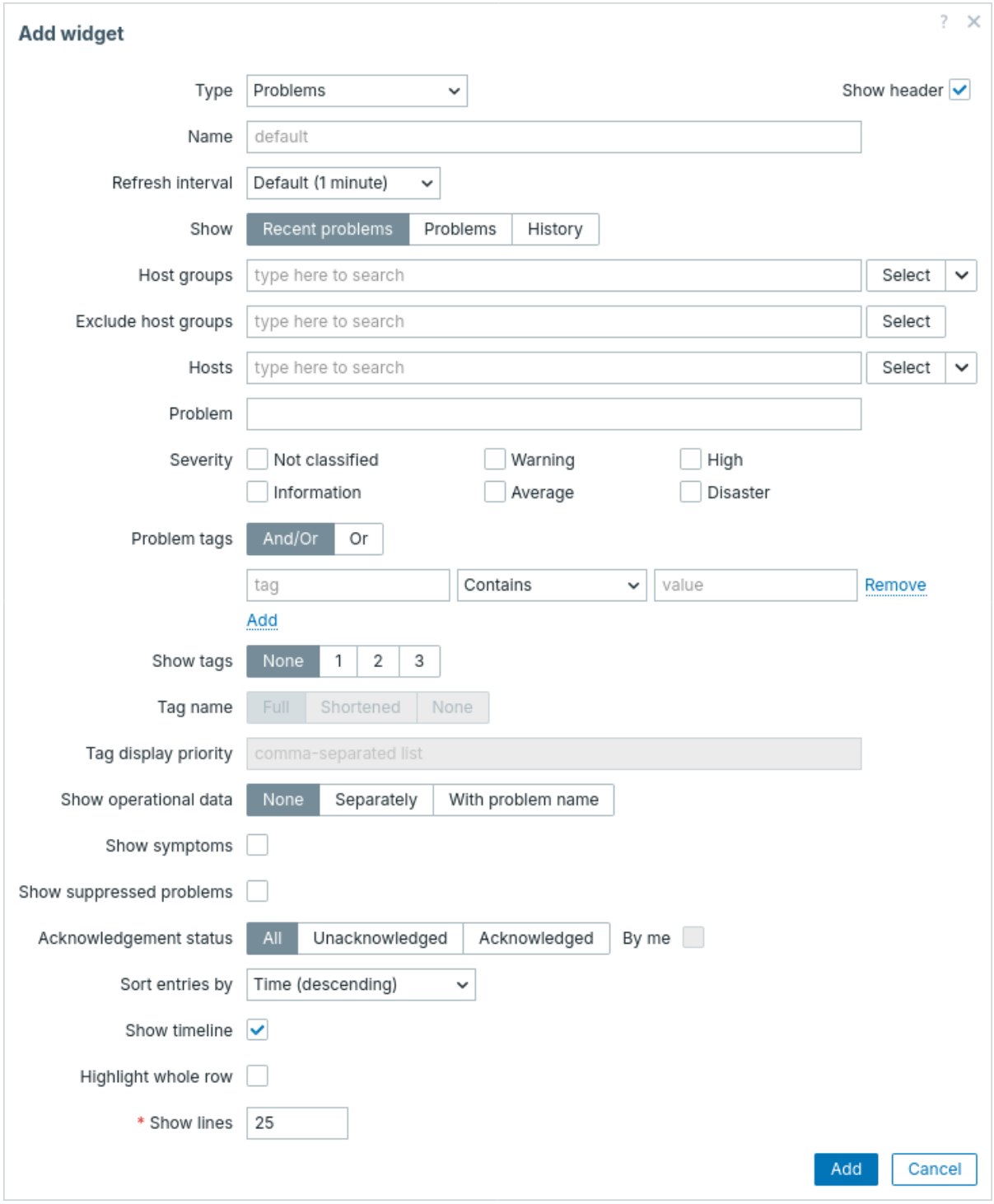
Обзор

В этом виджете можно отображать текущие проблемы. Информация в этом виджете аналогична разделу *Monitoring* > *Problems*.

Можно отображать до 1000 записей.

Конфигурация

Для настройки выберите *Problems* в качестве типа:




Вы можете различными способами ограничить количество проблем, отображаемых в виджете, — по статусу проблемы, имени проблемы, серьезности, группе узлов сети, узлу сети, тегу события, статусу подтверждения и т. д.

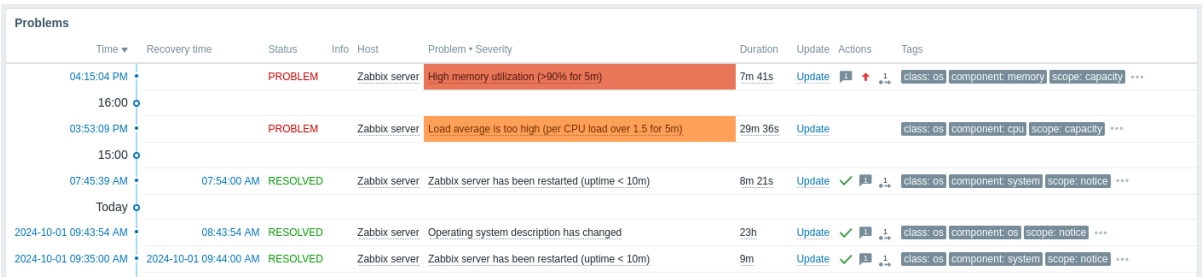
Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Show* | Фильтрация по статусу проблемы:<br>**Recent problems** - отображаются нерешенные и недавно решенные проблемы (по умолчанию);<br>**Problems** - отображаются нерешенные проблемы;<br>**History** - отображается история всех событий. |

| | |
|---|---|
| *Host groups* | Выберите группы узлов сети, проблемы которых нужно отображать в виджете.<br>Кроме того, можно выбрать совместимый виджет в качестве источника данных для групп узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет предлагаться выпадающий список подходящих групп.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети.<br>Проблемы из этих групп узлов сети будут отображаться в виджете; если группы узлов сети не указаны, будут отображаться проблемы из всех групп узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Exclude host groups* | Выберите группы узлов сети, проблемы которых нужно скрыть в виджете.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет предлагаться выпадающий список подходящих групп.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети.<br>Проблемы из этих групп узлов сети не будут отображаться. Например, узлы сети 001, 002, 003 могут входить в Group A, а узлы сети 002, 003 — также в Group B. Если одновременно выбрать *show* Group A и *exclude* Group B, в виджете будут отображаться только проблемы с узла сети 001.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Hosts* | Выберите узлы сети, проблемы которых нужно отображать в виджете.<br>Кроме того, можно выбрать совместимый виджет или панель в качестве источника данных для узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет предлагаться выпадающий список подходящих узлов сети.<br>Если узлы сети не указаны, будут отображаться проблемы всех узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Problem* | Можно ограничить количество отображаемых проблем по их имени.<br>Если ввести здесь строку, будут отображаться только те проблемы, в имени которых содержится введенная строка.<br>Макросы не разворачиваются. |
| *Severity* | Отметьте серьезности проблем, чтобы отфильтровать проблемы, отображаемые в виджете.<br>Если ни одна серьезность не отмечена, будут отображаться все проблемы. |
| *Problem tags* | Укажите теги проблем, чтобы ограничить количество проблем, отображаемых в виджете. Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны несколько операторов:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны выполняться все условия, условия с одинаковым именем тега будут сгруппированы по условию *Or*;<br>**Or** - достаточно выполнения одного условия.<br><br>При фильтрации указанные здесь теги будут отображаться первыми вместе с проблемой, если это не переопределено списком *Tag display priority* (см. ниже). |
| *Show tags* | Выберите количество отображаемых тегов:<br>**None** - столбец *Tags* не отображается;<br>**1** - столбец *Tags* содержит один тег;<br>**2** - столбец *Tags* содержит два тега;<br>**3** - столбец *Tags* содержит три тега.<br>Чтобы увидеть все теги проблемы, наведите указатель мыши на значок с тремя точками. |

| | |
|---|---|
| *Tag name* | Выберите режим отображения имен тегов:<br>**Full** - имена и значения тегов отображаются полностью;<br>**Shortened** - имена тегов сокращаются до 3 символов, но значения тегов отображаются полностью;<br>**None** - отображаются только значения тегов; имен нет. |
| *Tag display priority* | Введите приоритет отображения тегов для проблемы в виде списка тегов, разделенного запятыми.<br>Следует использовать только имена тегов, без значений.<br>Пример: `Services,Applications,Application`<br>Теги из этого списка всегда будут отображаться первыми, переопределяя естественный алфавитный порядок. |
| *Show operational data* | Выберите режим отображения оперативных данных:<br>**None** - оперативные данные не отображаются;<br>**Separately** - оперативные данные отображаются в отдельном столбце;<br>**With problem name** - оперативные данные добавляются к имени проблемы в скобках. |
| *Show symptoms* | Отметьте флажок, чтобы отображать проблемы, классифицированные как симптомы, в отдельной строке. |
| *Show suppressed problems* | Отметьте флажок, чтобы отображать проблемы, которые в противном случае были бы подавлены (не показаны) из-за обслуживания узла сети или отдельного подавления проблемы. |
| *Acknowledgement status* | Фильтр для отображения всех проблем, только неподтвержденных проблем или только подтвержденных проблем. Отметьте дополнительный флажок, чтобы исключить проблемы, которые когда-либо были подтверждены вами. |
| *Sort entries by* | Сортировать записи по:<br>**Time** (по убыванию или возрастанию);<br>**Severity** (по убыванию или возрастанию);<br>**Problem name** (по убыванию или возрастанию);<br>**Host** (по убыванию или возрастанию).<br><br>Сортировка записей по **Host** (по убыванию или возрастанию) недоступна при настройке виджета на шаблонной панели. |
| *Show timeline* | Отметьте флажок, чтобы отображать визуальную временную шкалу. |
| *Highlight whole row* | Отметьте флажок, чтобы выделять всю строку для нерешенных проблем. Для выделения используется цвет серьезности проблемы.<br>*Highlight whole row* недоступен в темах с высокой контрастностью.<br>Этот параметр поддерживается начиная с Zabbix 7.4.3. |
| *Show lines* | Укажите количество строк проблем для отображения. |

Использование виджета

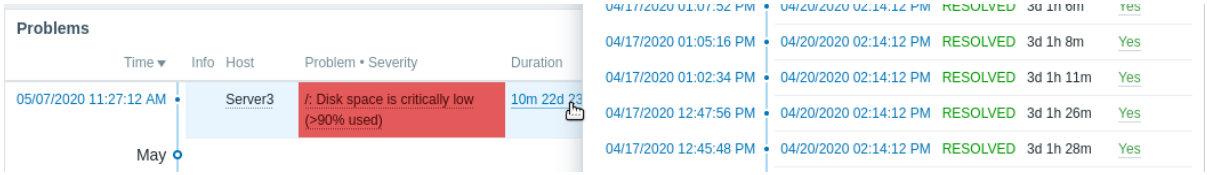



Виджет проблем предоставляет быстрый доступ к дополнительной информации:

- Щелкните по дате проблемы (в столбце Time) или по дате восстановления (в столбце Recovery Time), чтобы просмотреть сведения о событии.
- Если поле Info не пустое, наведите указатель мыши на отображаемый значок, чтобы просмотреть дополнительные сведения.
- Щелкните по имени узла сети, чтобы открыть меню узла сети.
- Щелкните по имени проблемы, чтобы открыть меню события.
- Наведите указатель мыши на длительность проблемы или щелкните по ней, чтобы просмотреть всплывающее окно события проблемы.
- Щелчок по Update открывает окно Обновление проблемы.
- Наведите указатель мыши на серый значок стрелки в столбце Actions или щелкните по нему, чтобы просмотреть список выполненных действий.

Всплывающее окно события проблемы

Всплывающее окно события проблемы содержит список событий проблемы для этого триггера и, если задано, описание триггера и кликабельный URL.




Чтобы открыть всплывающее окно события проблемы:

- Наведите указатель мыши на длительность проблемы в столбце *Duration* виджета *Problems*. Всплывающее окно исчезает, как только вы убираете указатель мыши с длительности.
- Щелкните по длительности в столбце *Duration* виджета *Problems*. Всплывающее окно исчезает только если вы щелкнете по длительности еще раз.

24 Проблемы по уровню серьезности

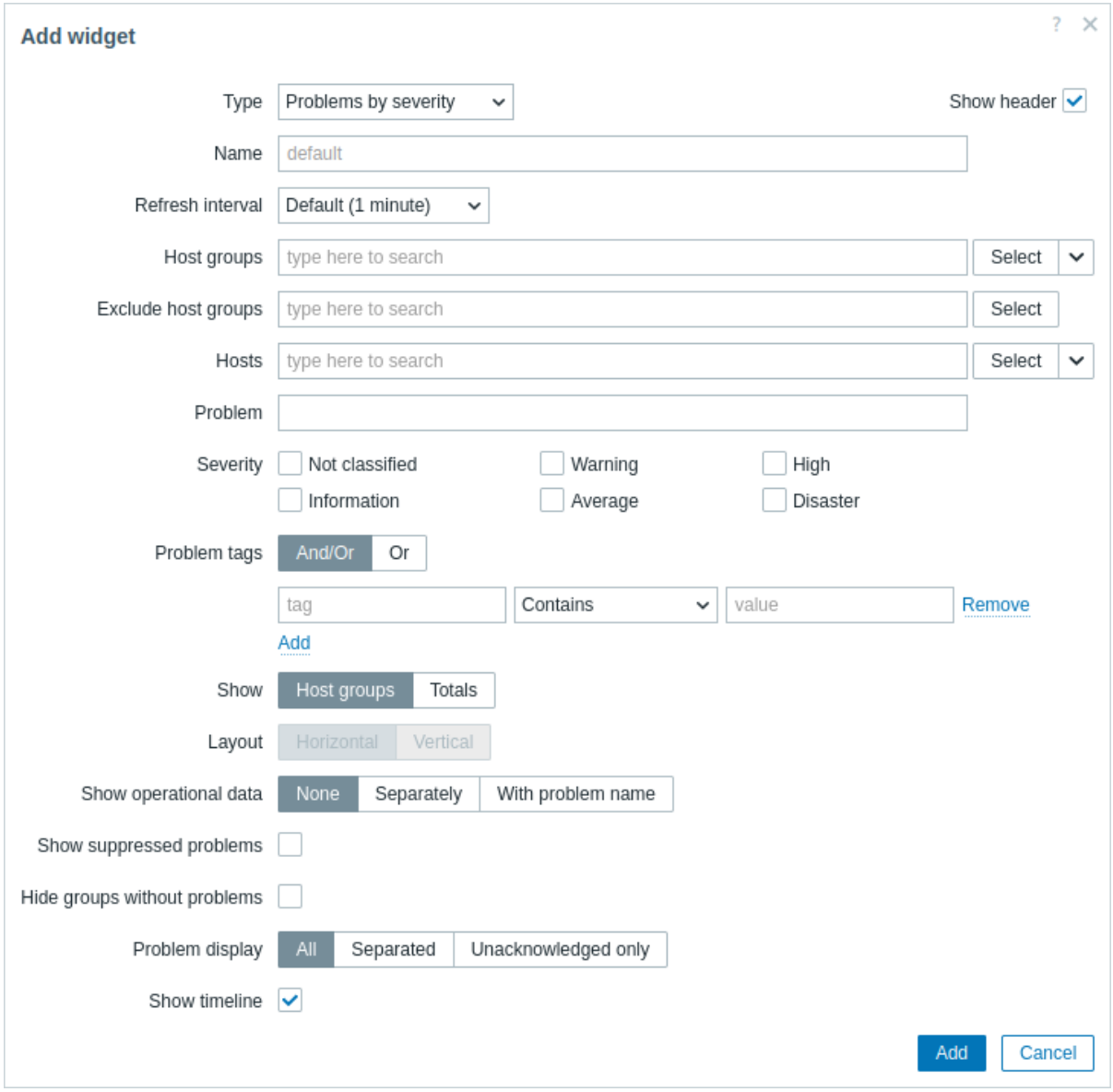
Обзор

В этом виджете можно отобразить количество проблем по их важности. Вы можете ограничить, какие узлы сети и триггеры отображаются в виджете, и определить, как отображается количество проблем.

Количество проблем отображается только для проблем-причин.

Настройка

Для настройки выберите *Проблемы по важности (Problems by severity)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети для отображения в виджете.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети.<br>Данные от этих групп узлов сети будут отображаться в виджете; если не выбрана ни одна группа, то будут отображаться все группы узлов сети.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Исключить группы узлов сети (Exclude host groups)* | Выберите группы узлов сети, которые нужно скрыть от виджета.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети.<br>Данные узлов сети из этих групп не будут отображаться в виджете. Например, узлы сети 001, 002, 003 могут быть в группе «Group A», а узлы сети 002, 003 — также и в группе «Group B». Если мы одновременно выбираем *показать* группу «Group A» и *исключить* группу «Group B», то на панели будут отображены данные только с узла сети 001.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |

| | |
|---|---|
| *Узлы сети (Hosts)* | Выберите узлы сети для отображения в виджете.<br>В качестве альтернативы выберите для узлов сети в качестве источника данных совместимый виджет или панель.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени узла сети будет автоматически предложен список совпадающих узлов сети в выпадающем меню.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Проблема (Problem)* | Количество отображаемых узлов сети с проблемами можно ограничить по имени проблемы.<br>Если ввести здесь строку, будут отображаться только те узлы сети, чьи имена проблем содержат введённую строку.<br>Макросы не раскрываются. |
| *Важность (Severity)* | Укажите важности проблем, чтобы отфильтровать проблемы для отображения в виджете.<br>Если никакие важности не отмечены, будут отображаться все проблемы. |
| *Теги проблем (Problem tags)* | Укажите теги проблем для ограничения количества проблем, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно указать несколько условий. Поиск соответствия имён тегов всегда чувствителен к регистру.<br><br>По каждому условию доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием *Или (Or)*;<br>**Или (Or)** — достаточно соблюдения одного условия. |
| *Показать (Show)* | Выберите вариант отображения:<br>**Группы узлов сети (Host groups)** — отображать проблемы по группам узлов сети;<br>**Общее количество (Totals)** — отображать общее количество проблем для всех выбранных групп узлов сети в блоках, подкрашенных в соответствии с важностью проблем.<br>Этот параметр недоступен при настройке виджета на панели шаблона, и будет отображаться только общее количество проблем. |
| *Размещение (Layout)* | Выберите вариант расположения:<br>**Горизонтальный (Horizontal)** — подкрашенные блоки с общим количеством будут отображаться горизонтально;<br>**Вертикальный (Vertical)** — подкрашенные блоки с общим количеством будут отображаться вертикально.<br>Это поле доступно, только если в опции *Показать (Show)* выбрано «Общее количество (Totals)». |
| *Оперативные данные (Show operational data)* | Отображение оперативных данных (смотрите описание *Оперативных данных* в *Мониторинг (Monitoring)* → *Проблемы (Problems)*):<br>**Нет (None)** — не показывать оперативные данные<br>**Раздельно (Separately)** — показывать оперативные данные в отдельном строке<br>**С именем проблемы (With problem name)** — оперативные данные будут указаны вместе с именем проблемы. |
| *Подавленные проблемы (Show suppressed problems)* | Установите флажок для отображения проблем, которые в противном случае могут быть подавлены (не показаны) по причине режима обслуживания узла сети. |
| *Скрыть группы без проблем (Hide groups without problems)* | Отметьте опцию *Скрыть группы без проблем (Hide groups without problems)*, чтобы скрыть в виджете данные от групп узлов сети, в которых проблемы отсутствуют.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |

| | |
|---|---|
| *Отображение проблем (Problem display)* | Отображать количество проблем как:<br>**Все (All)** — будет отображаться общее количество проблем;<br>**Раздельно (Separated)** — количество неподтвержденных проблем будет отображаться отдельным числом от общего количества проблем;<br>**Только неподтверждённые (Unacknowledged only)** — будет отображаться только количество неподтверждённых проблем. |
| *Шкала времени (Show timeline)* | Установите флажок для отображения визуальной шкалы времени. |

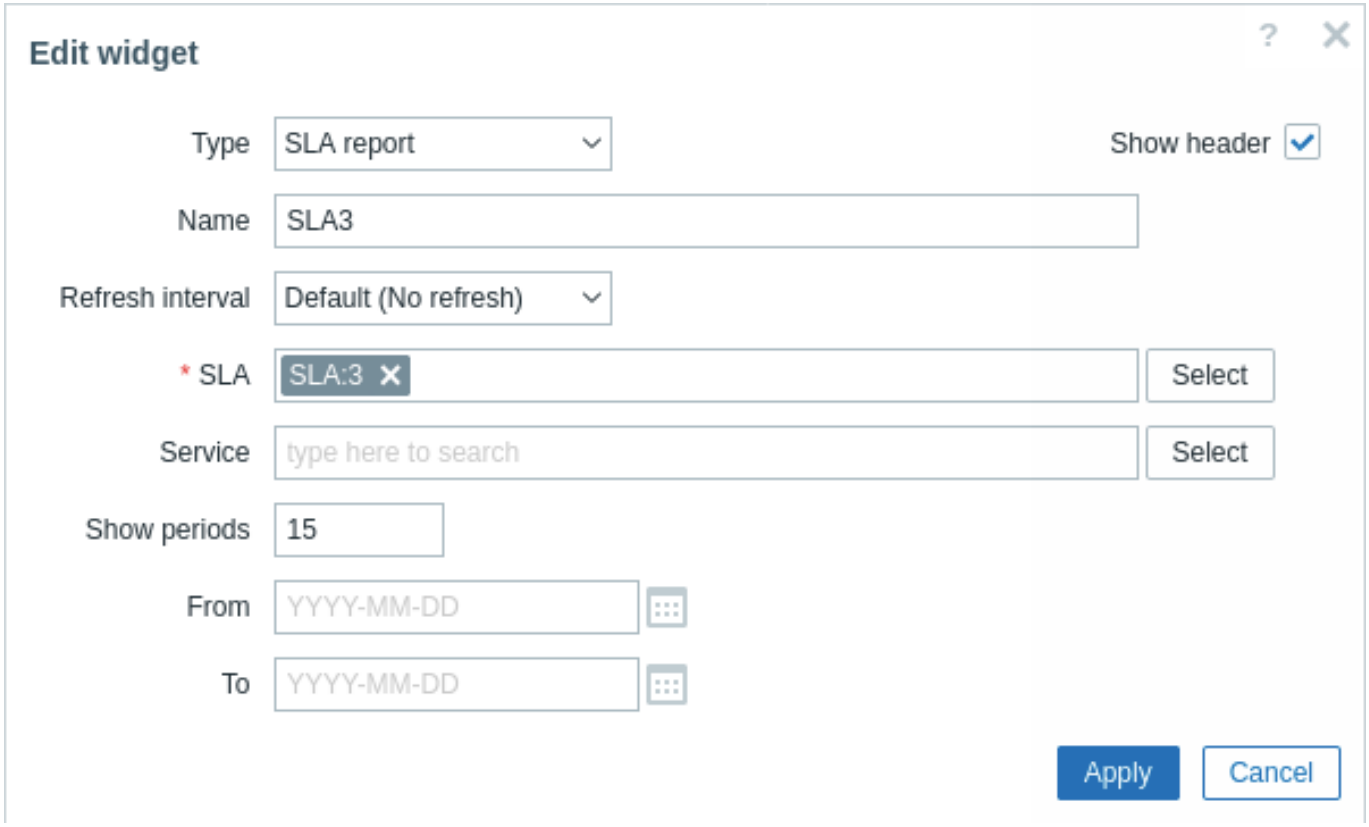
25 Отчет SLA

Обзор

Этот виджет полезен для отображения отчётов SLA. Функционально он аналогичен разделу *Услуги (Services)* → *Отчёты SLA (SLA report)*.

Настройка

Для настройки выберите *Отчёт SLA (SLA report)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *SLA* | Выберите SLA для отчёта. |
| *Услуга (Service)* | Выберите услугу для отчёта. |
| *Отображение периодов (Show periods)* | Укажите, сколько периодов будет отображаться в виджете (по умолчанию 20, максимум 100). |
| *От (From)* | Выберите начальную дату для отчёта.<br>Поддерживаются относительные даты: `now`, `now/d`, `now/w-1w` и т.д.; поддерживаются модификаторы дат: d, w, M, y. |
| *К (To)* | Выберите конечную дату для отчёта.<br>Поддерживаются относительные даты: `now`, `now/d`, `now/w-1w` и т.д.; поддерживаются модификаторы дат: d, w, M, y. |

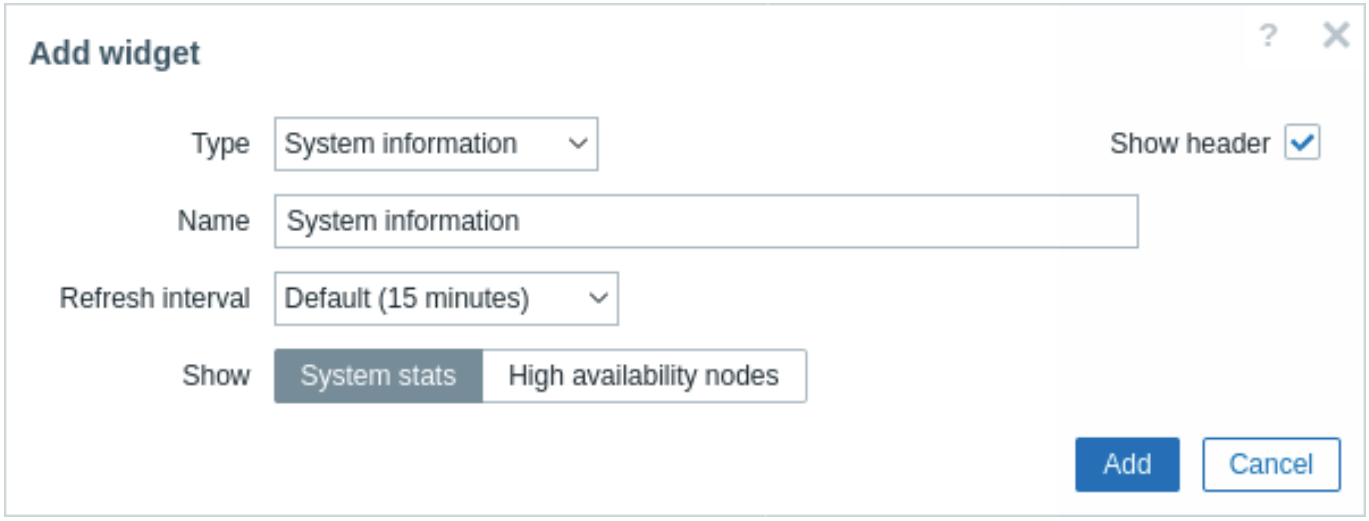
## 26 Системная информация

### Обзор

В этом виджете отображается та же информация, что и в *Отчеты (Reports)* → *Информация о системе (Systeminformation)*, однако, один виджет панели может отображать либо только статистику системы, либо ноды отказоустойчивого кластера (не то и другое одновременно).

### Настройка

Для настройки выберите *Информация о системе (System information)* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

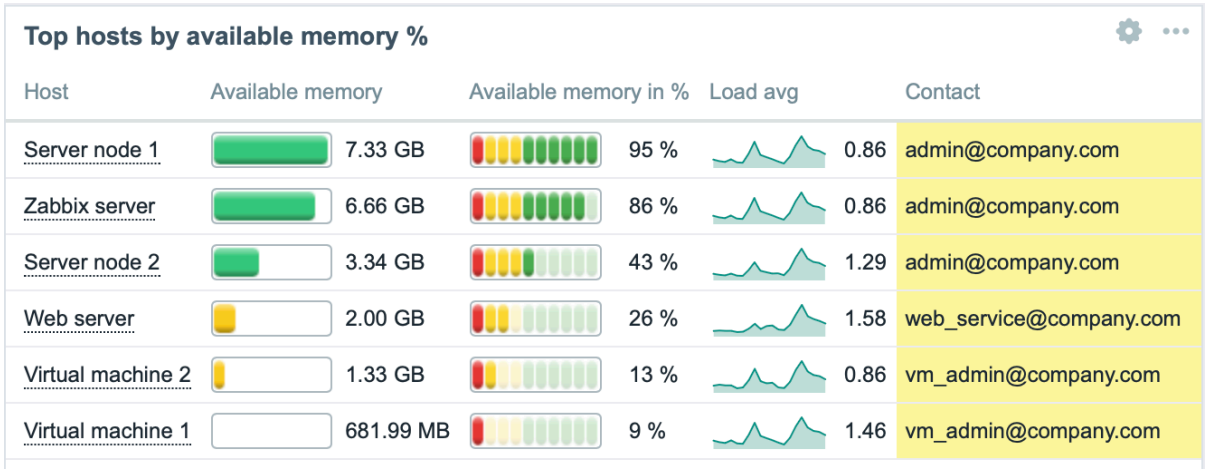
| | |
|---|---|
| *Показать (Show)* | Выбор, что отображать:<br>**Статистика системы (System stats)** — отобразить общую сводку о ключевых данных сервера Zabbix и системы;<br>**Ноды отказоустойчивого кластера (High availability nodes)** — отобразить состояние нод отказоустойчивого кластера (если активирован кластер высокой доступности). |
| *Детали проверки обновлений программного обеспечения (Show software update check details)* | Установите флажок для отображения подробностей проверки обновлений программного обеспечения Zabbix.<br>Этот параметр доступен, только если проверка обновлений программного обеспечения активирована в настройках Zabbix сервера, а в поле *Показать (Show)* выбрано «Статистика системы (System stats)». |

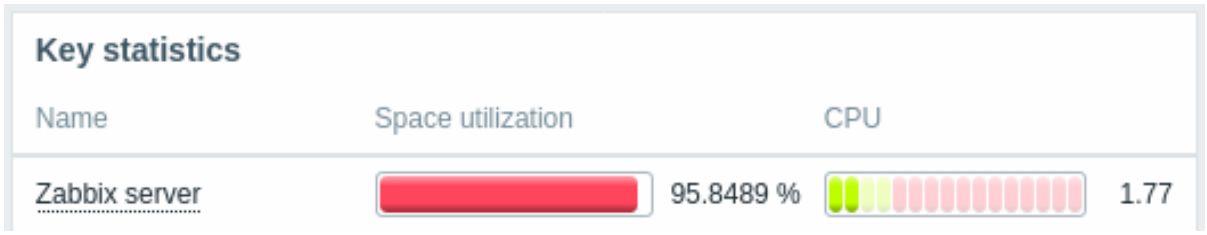
## 27 Лучшие узлы сети

### Обзор

Виджет Top hosts предоставляет возможность создавать пользовательские таблицы для отображения данных по нескольким узлам сети и может использоваться для создания отчетов в стиле *Top N* и отчетов с полосами прогресса, полезных для планирования емкости. Тип отображаемых данных настраивается и может включать загрузку CPU, доступность памяти или любые другие собранные значения в числовом, текстовом или двоичном формате, а также статические текстовые строки. Данные могут отображаться в различных форматах, таких как полосы, индикаторы или графики sparkline.

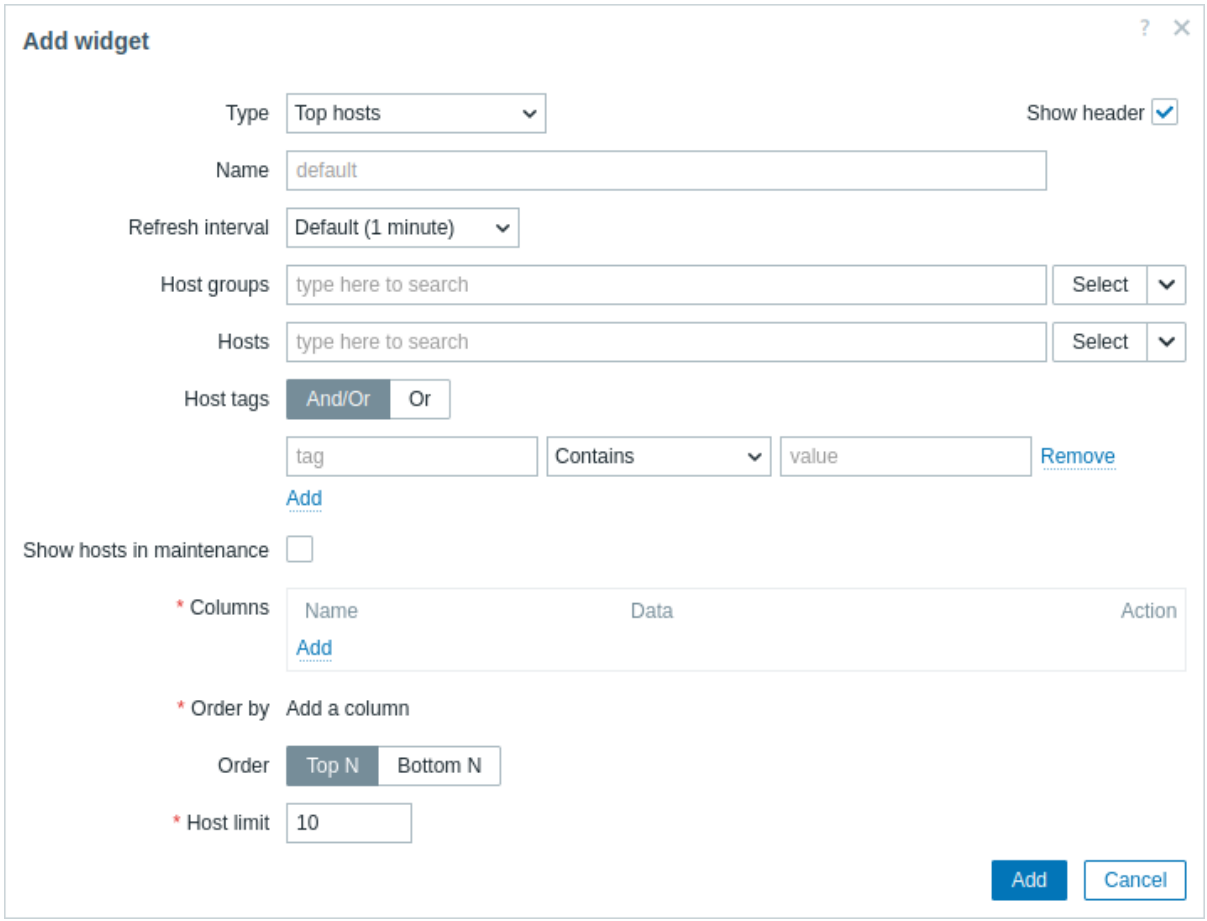
Максимальное количество узлов сети, которое можно отобразить, — 1000.




Настройка

Для настройки выберите *Топ узлов сети (Top hosts)* в качестве типа:




Помимо параметров, которые являются общими для всех виджетов, можно задать следующие специальные параметры:

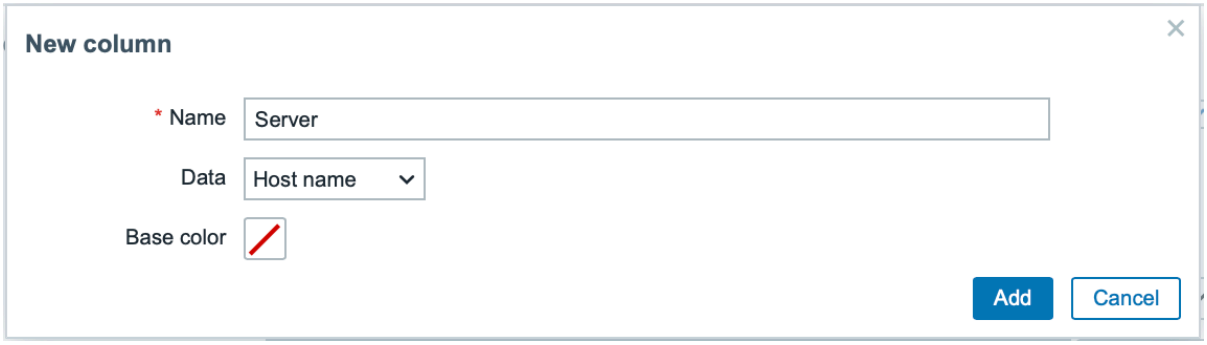
| | |
|---|---|
| *Группы узлов сети* | Выберите группы узлов сети для отображения в виджете.<br>В качестве альтернативы выберите совместимый виджет как источник данных для групп узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет предлагаться выпадающий список подходящих групп.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Узлы сети* | Выберите узлы сети для отображения в виджете.<br>В качестве альтернативы выберите совместимый виджет или панель как источник данных для узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет предлагаться выпадающий список подходящих узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Теги узлов сети* | Укажите теги, чтобы ограничить число узлов сети, отображаемых в виджете.<br>Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны несколько операторов:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия.<br><br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Показывать узлы сети в обслуживании* | Установите этот флажок, чтобы также отображались узлы сети в обслуживании (в этом случае рядом с именем узла сети будет показан значок обслуживания). По умолчанию не отмечен. |
| *Столбцы* | Добавьте столбцы данных для отображения.<br>Порядок столбцов определяет их отображение слева направо.<br>Столбцы можно менять местами, перетаскивая их вверх и вниз за маркер перед именем столбца. |
| *Сортировать по* | Укажите столбец из заданного списка *Столбцы*, который будет использоваться для сортировки *Top N* или *Bottom N*. |
| *Порядок* | Укажите порядок строк:<br>**Top N** - по убыванию в соответствии с агрегированным значением *Сортировать по*;<br>**Bottom N** - по возрастанию в соответствии с агрегированным значением *Сортировать по*. |
| *Лимит узлов сети* | Количество строк узлов сети для отображения (1-1000).<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |

Конфигурация столбцов

Можно добавить столбцы трех типов данных: имя узла сети, значение элемента данных или текст. Список доступных параметров столбца зависит от типа данных столбца и, для типа значения элемента данных, от формата значения.

Столбец имени узла сети

Столбец имени узла сети используется для отображения имени узла сети.

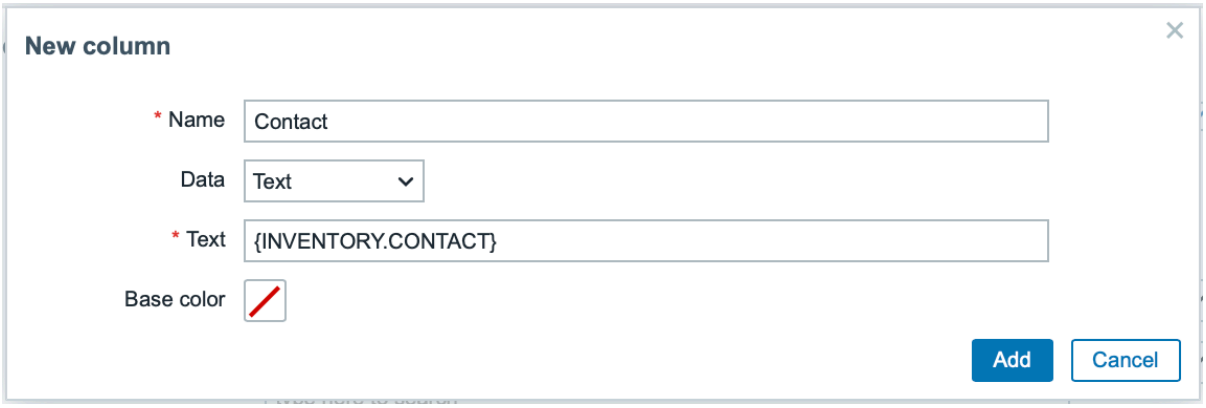



Поддерживаемые параметры:

| | |
|---|---|
| *Name* | Имя столбца. |
| *Data* | Тип данных, отображаемых в столбце, выберите **Host name**. |
| *Base color* | Цвет фона столбца. |

Текстовый столбец

Текстовый столбец используется для отображения любой указанной текстовой строки.

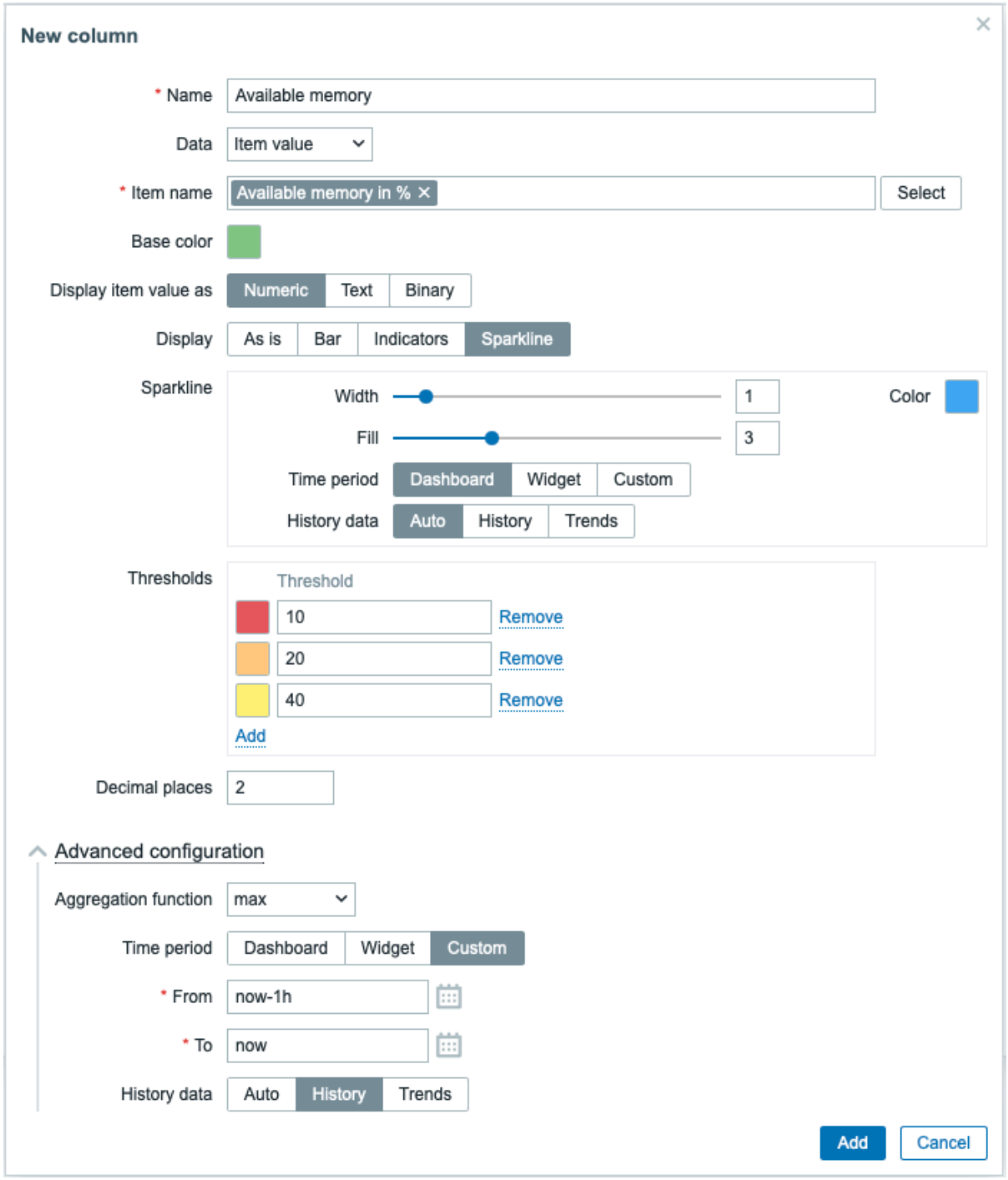



Поддерживаемые параметры:

| | |
|---|---|
| *Name* | Имя столбца. |
| *Data* | Тип данных для отображения в столбце, выберите **Text**. |
| *Base color* | Цвет фона столбца. |
| *Text* | Введите строку для отображения.<br>Может содержать макросы узла сети и инвентаризации. |

Столбец значения элемента данных

Столбец значения элемента данных используется для отображения значения указанного элемента данных; он поддерживает несколько форматов отображения значений и параметров.




Поддерживаемые параметры:

| | |
|---|---|
| *Name* | Имя столбца. |
| *Data* | Тип данных для отображения в столбце, выберите **Item value**. |
| *Item name* | Выберите элемент данных; его имя используется для сопоставления и отображения элементов данных с тем же именем на всех выбранных узлах сети. Если у узла сети есть несколько таких элементов данных, виджет отображает элемент данных, ключ которого первым идет в алфавитном порядке (например, *CPU utilization* с ключом `proc.cpu.util` вместо `system.cpu.util`).<br>При настройке виджета на шаблонной панели можно выбрать только элементы данных, настроенные в шаблоне. |
| *Base color* | Цвет фона столбца; цвет заливки, если данные *Item value* отображаются в виде полосы/индикаторов. Цвет по умолчанию может быть переопределен пользовательским цветом, если значение элемента данных превышает один из указанных порогов. |
| *Display item value as* | Формат отображения значения элемента данных: **Numeric**, **Text** или **Binary**.<br>Выбранный вариант определяет, какие дополнительные параметры будут доступны. См. список параметров для каждого формата ниже. |

| | |
|---|---|
| ***Advanced configuration*** | |
| *Aggregation function* | Укажите, какую функцию агрегации использовать в пределах выбранного *Time period*:<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение;<br>**count** - отображать количество значений;<br>**sum** - отображать сумму значений;<br>**first** - отображать первое значение;<br>**last** - отображать последнее значение.<br>Если выбрано **not used**, отображается самое последнее значение (без агрегации).<br><br>Агрегация позволяет отображать агрегированное значение за выбранный интервал (5 минут, час, день) вместо самого последнего значения.<br>Только числовые данные могут отображаться для *min*, *max*, *avg* и *sum*. Для *count* нечисловые данные будут преобразованы в числовые. |
| *Time period* | Укажите временной период, который будет использоваться для агрегации значений:<br>**Dashboard** - использовать временной период панели;<br>**Widget** - использовать временной период указанного виджета;<br>**Custom** - использовать пользовательский временной период.<br>Этот параметр не будет отображаться, если для *Aggregation function* выбрано "not used". |
| *Widget* | Выберите виджет.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Widget". |
| *From* | Выберите начало временного периода (значение по умолчанию `now-1h`). См. синтаксис относительного времени.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Custom". |
| *To* | Выберите конец временного периода (значение по умолчанию `now`). См. синтаксис относительного времени.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Custom". |
| *History data* | Брать данные из истории или трендов:<br>**Auto** - автоматический выбор;<br>**History** - брать данные истории;<br>**Trends** - брать данные трендов.<br>Этот параметр доступен только для числовых значений элементов данных. Для текстовых и бинарных значений данные всегда будут браться из истории. |

**Параметры числового значения элемента данных**:

| | |
|---|---|
| *Display* | Определите, как должно отображаться значение:<br>**As is** - как обычный текст;<br>**Bar** - как сплошная полоса с заливкой цветом;<br>**Indicators** - как сегментированная полоса с заливкой цветом;<br>**Sparkline** - мини-график. |
| *Thresholds* | Укажите пороговые значения, при которых должен изменяться цвет фона/заливки.<br>При сохранении список будет отсортирован по возрастанию.<br>Для sparkline пороги применяются только к последнему значению элемента данных. |
| *Decimal places* | Укажите, сколько знаков после запятой будет отображаться у значения. |
| ***Bar/Indicators*** | |
| *Min* | Минимальное значение. |
| *Max* | Максимальное значение. |
| ***Sparkline*** | |
| *Width* | Задайте толщину линии графика с помощью ползунка или вручную введите значение в диапазоне от 0 до 10. |
| *Color* | Выберите цвет линии и заливки. |
| *Fill* | Задайте уровень прозрачности цвета заливки с помощью ползунка или вручную введите значение в диапазоне от 0 до 10. |

| | |
|---|---|
| *Time period* | Укажите временной период, значения которого будут включены в график sparkline:<br>**Dashboard** - использовать временной период панели;<br>**Widget** - использовать временной период указанного виджета;<br>**Custom** - использовать пользовательский временной период.<br><br>Обратите внимание, что временной период влияет только на график sparkline. Значение, отображаемое рядом со sparkline, представляет собой последнее значение элемента данных, а не последнее значение в выбранном временном периоде. |
| *Widget* | Выберите виджет.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Widget". |
| *From* | Выберите начало временного периода (значение по умолчанию `now-1h`). См. синтаксис относительного времени.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Custom". |
| *To* | Выберите конец временного периода (значение по умолчанию `now`). См. синтаксис относительного времени.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Custom". |
| *History data* | Брать данные из истории или трендов:<br>**Auto** - автоматический выбор;<br>**History** - брать данные истории;<br>**Trends** - брать данные трендов. |

**Параметры текстового значения элемента данных**:

| | |
|---|---|
| *Highlights* | Укажите регулярные выражения, при совпадении с которыми должен изменяться цвет фона/заливки. |

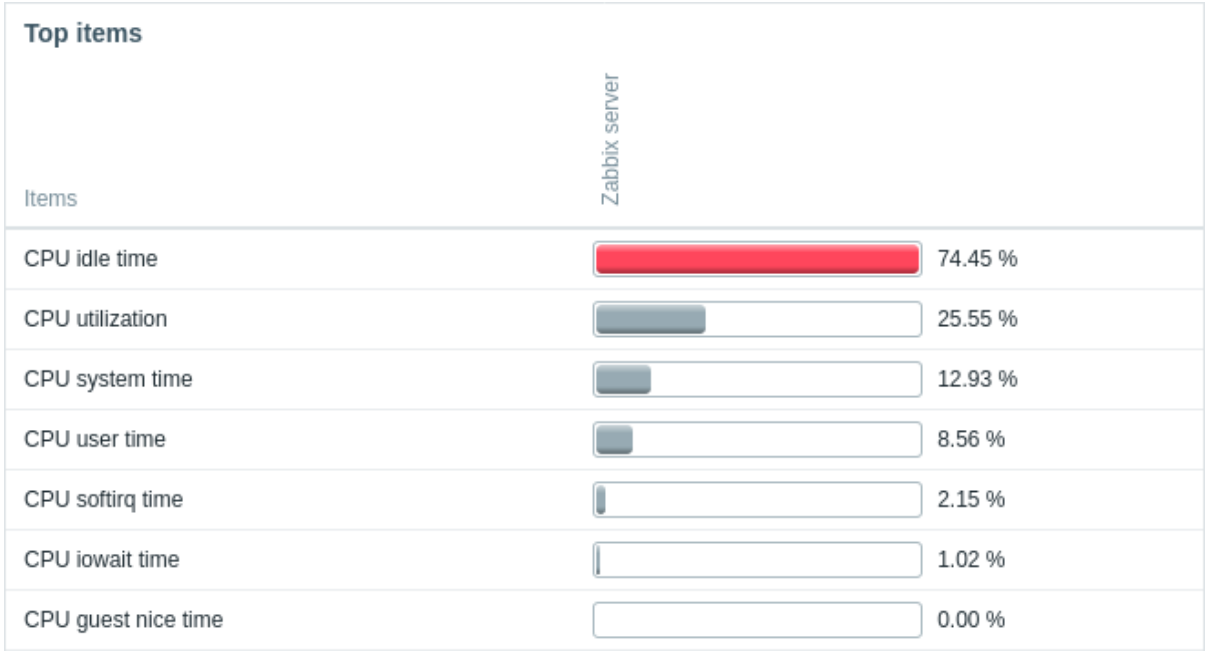
**Параметры бинарного значения элемента данных**:

| | |
|---|---|
| *Show thumbnail* | Укажите, нужно ли создавать и отображать миниатюру для изображения, содержащего бинарные данные, или отображать гиперссылку *Show*, ведущую к изображению в полном размере в столбце значения. |

28 Лучшие элементы данных

Обзор

В виджете *Top items* можно отображать наибольшие (или наименьшие) значения для группы элементов данных.



Можно выбирать элементы данных напрямую или по шаблону, настраивать порядок столбцов и выделение.

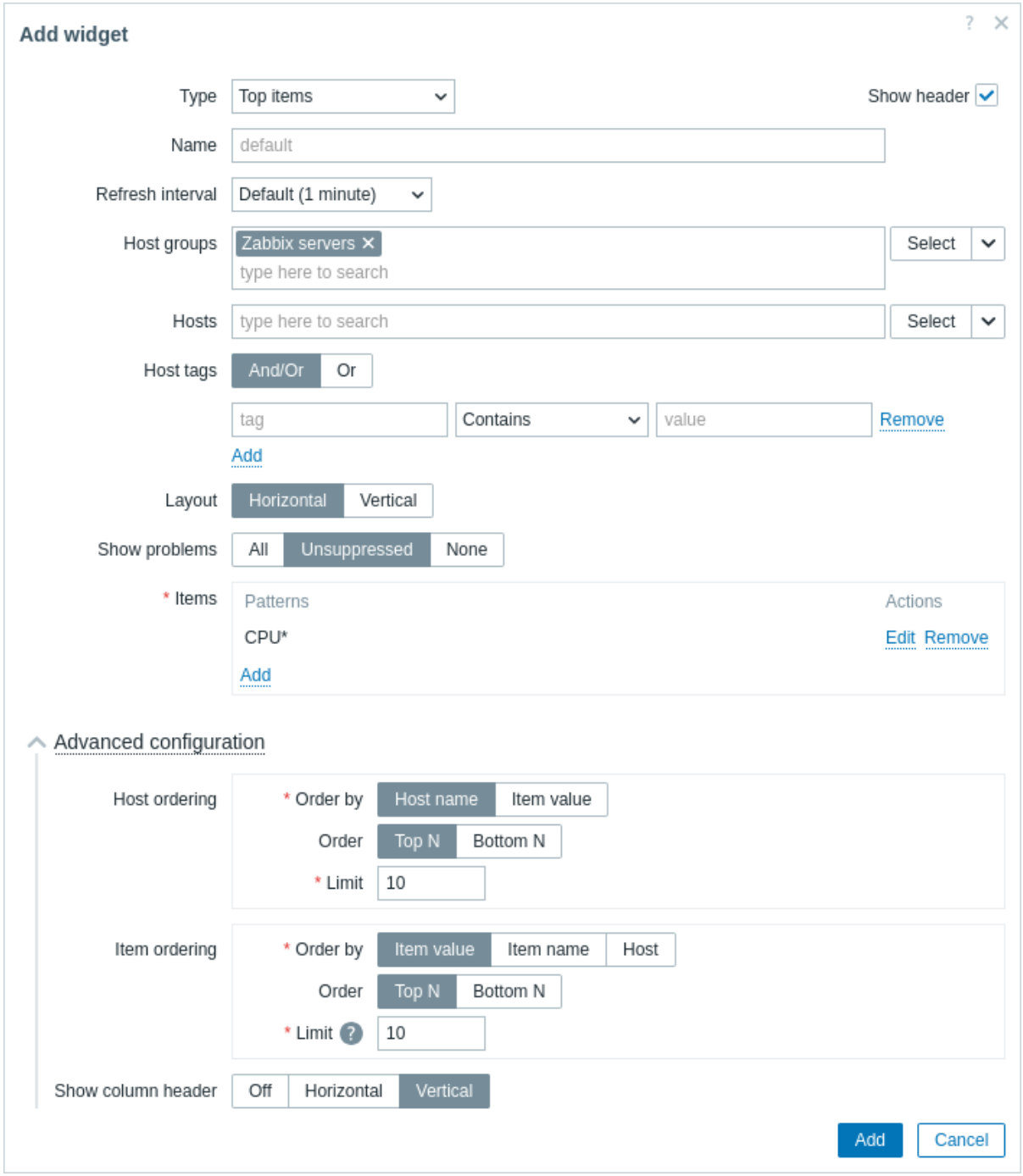
Цвет проблемных элементов данных основан на цвете серьезности проблемы, который можно изменить на экране обновления проблемы.

По умолчанию отображаются только значения, попадающие в последние 24 часа. Это ограничение было введено с целью улучшения времени первоначальной загрузки для больших страниц с последними данными. Этот предел настраивается в *Администрирование* → *Общее* → *GUI* с помощью параметра *Максимальный период отображения истории*.

При нажатии на значение элемента данных открывается меню элемента данных.

Конфигурация

Чтобы настроить, выберите *Top items* в качестве типа:



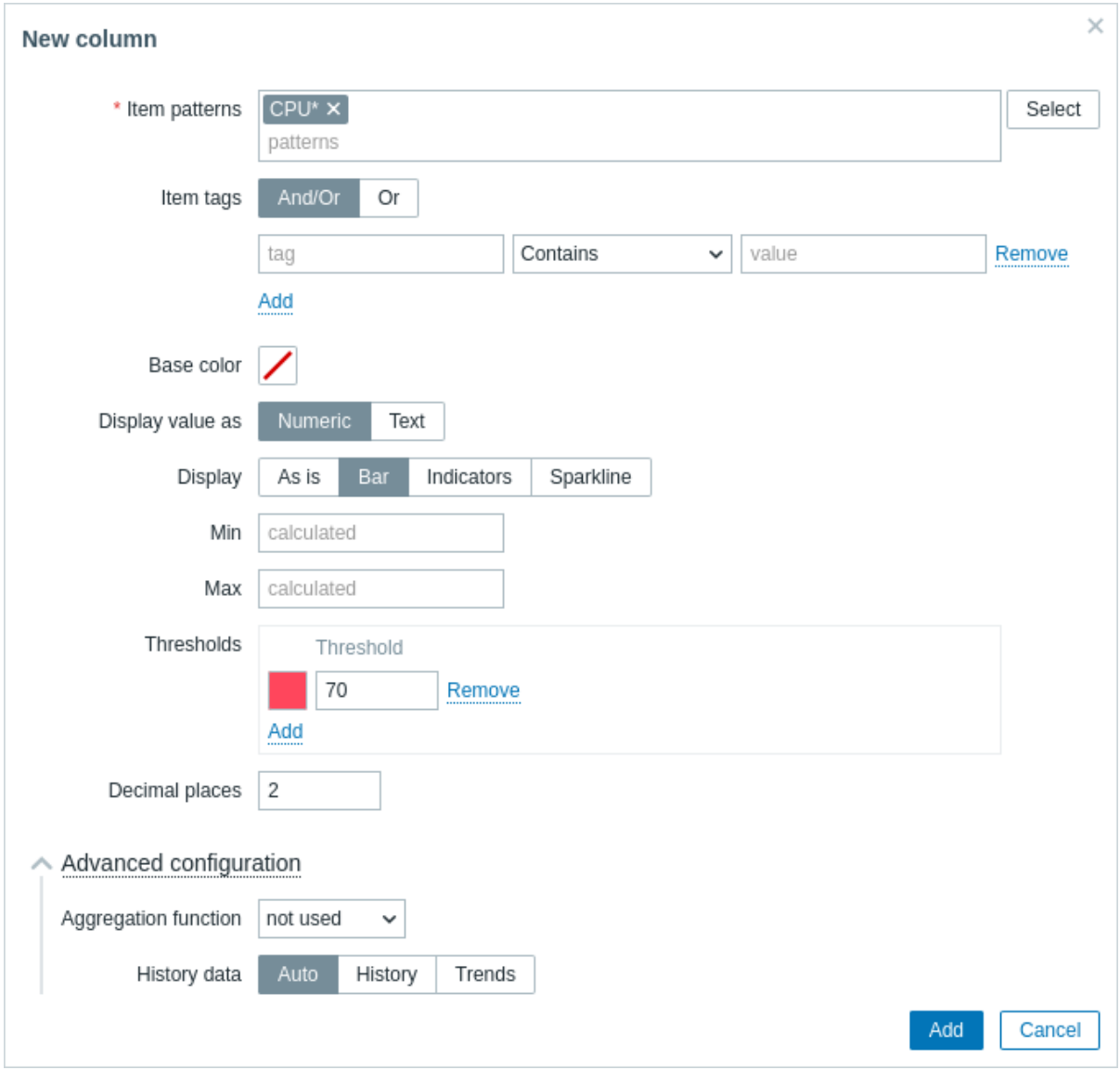

Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Группы узлов сети* | Выберите группы узлов сети.<br>Кроме того, можно выбрать совместимый виджет в качестве источника данных для групп узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет предлагаться выпадающий список подходящих групп.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Узлы сети* | Выберите узлы сети.<br>Кроме того, можно выбрать совместимый виджет или панель в качестве источника данных для узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет предлагаться выпадающий список подходящих узлов.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Теги узлов сети* | Укажите теги, чтобы ограничить количество данных узлов сети, отображаемых в виджете.<br>Можно включать и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны несколько операторов:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена и значения тегов (с учетом регистра);<br>**Contains** - включить указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена и значения тегов (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию *Or*;<br>**Or** - достаточно выполнения одного условия. |
| *Макет* | Выберите вариант макета:<br>**Horizontal** - имена узлов сети будут отображаться слева;<br>**Vertical** - имена узлов сети будут отображаться сверху. |
| *Показывать проблемы* | Выберите, как отображать проблемы:<br>**All** - отображать все проблемы;<br>**Unsuppressed** - отображать только проблемы, не подавленные из-за обслуживания узла сети;<br>**None** - не отображать проблемы вообще.<br>*Примечание*: если отображается состояние проблемы, оно переопределяет параметры стиля отображения из конфигурации столбца элемента данных. |
| *Элементы данных* | Добавьте шаблоны элементов данных или конкретные элементы данных для отображения (см. Конфигурация столбца). |
| ***Дополнительная конфигурация*** | |
| *Порядок узлов сети* | Выберите параметры сортировки для столбца/строки узлов сети. |
| *Сортировать по* | Сортировать узлы сети по:<br>**Host name** - узлы сети будут отсортированы по имени узла сети;<br>**Item value** - узлы сети будут отсортированы по значению выбранных элементов данных. |
| *Порядок* | Выберите, отображать ли наибольшие или наименьшие значения:<br>**TopN** - первые N значений;<br>**BottomN** - последние N значений.<br>Значение N выбирается в поле *Limit*. |
| *Ограничение* | Введите ограничение для отображаемых узлов сети (диапазон 1-1000; по умолчанию 10).<br>Это значение станет значением N в поле *Order*. |
| *Порядок элементов данных* | Выберите параметры сортировки для столбца/строки элементов данных. |

| | |
|---|---|
| *Сортировать по* | Сортировать элементы данных по:<br>**Item value** - элементы данных будут отсортированы по значению элемента данных;<br>**Item name** - элементы данных будут отсортированы по имени элемента данных;<br>**Host** - элементы данных будут отсортированы по выбранному шаблону узла сети. |
| *Порядок* | Выберите, отображать ли наибольшие или наименьшие значения:<br>**TopN** - первые N значений;<br>**BottomN** - последние N значений.<br>Значение N выбирается в поле *Limit*. |
| *Ограничение* | Введите ограничение для отображаемых элементов данных (диапазон 1-1000; по умолчанию 10).<br>Это значение станет значением N в поле *Order*. |
| *Показывать заголовок столбца* | Выберите параметры отображения заголовка столбца:<br>**Off** - не отображать заголовок столбца;<br>**Horizontal** - отображать текст в заголовке горизонтально;<br>**Vertical** - отображать текст в заголовке вертикально. |

Конфигурация столбцов

Чтобы настроить столбцы элементов данных, нажмите *Add* в параметре *Items*:




Общие параметры столбца:

| | |
|---|---|
| *Item patterns* | Укажите один или несколько шаблонов элементов данных, используя символ подстановки. Либо выберите элементы данных.<br>При настройке виджета на панели шаблона можно выбрать только элементы данных, настроенные в шаблоне. |
| *Item tags* | Укажите теги, чтобы ограничить количество данных элементов, отображаемых в виджете. Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию *Or*;<br>**Or** - достаточно выполнения одного условия. |
| *Base color* | Выберите цвет фона столбца или цвет заливки, если для *Display* установлено значение ”Bar” или ”Indicators”.<br>Обратите внимание, что базовый цвет можно переопределить цветами порога или подсветки. |
| *Display value as* | Формат отображения значения элемента данных: **Numeric** или **Text**.<br><br>Выбранный вариант определяет, какие дополнительные параметры будут доступны. См. список параметров для каждого формата ниже. |

Параметры столбца для числовых значений:

| | |
|---|---|
| *Display* | Определите, как должно отображаться значение:<br>**As is** - как обычный текст;<br>**Bar** - как сплошная цветная полоса;<br>**Indicators** - как сегментированная цветная полоса;<br>**Sparkline** - мини-график линии. |
| *Min* | Минимальное значение для отображения bar/indicators. |
| *Max* | Максимальное значение для отображения bar/indicators. |
| *Width* | Задайте толщину линии графика с помощью ползунка или вручную введите значение в диапазоне от 0 до 10.<br>Этот параметр используется только для отображения sparkline. |
| *Fill* | Задайте уровень прозрачности цвета заливки с помощью ползунка или вручную введите значение в диапазоне от 0 до 10.<br>Этот параметр используется только для отображения sparkline. |
| *Color* | Выберите цвет линии и заливки.<br>Этот параметр используется только для отображения sparkline. |
| См. *Advanced configuration* для описания полей, связанных с выбором временного периода и данных истории. | |
| *Thresholds* | Укажите пороговые значения, при которых должен изменяться цвет фона/заливки.<br>При сохранении список будет отсортирован по возрастанию.<br>Для sparkline пороги применяются только к последнему значению элемента данных. |
| *Decimal places*<br>***Advanced configuration*** | Укажите, сколько знаков после запятой будет отображаться вместе со значением. |

| | |
|---|---|
| *Aggregation function* | Укажите, какую функцию агрегации использовать в выбранном *Time period*:<br>**min** - отображать наименьшее значение;<br>**max** - отображать наибольшее значение;<br>**avg** - отображать среднее значение;<br>**count** - отображать количество значений;<br>**sum** - отображать сумму значений;<br>**first** - отображать первое значение;<br>**last** - отображать последнее значение.<br>Если выбрано **not used**, отображается самое последнее значение (без агрегации).<br><br>Агрегация позволяет отображать агрегированное значение за выбранный интервал (5 минут, час, день) вместо самого последнего значения.<br>Только числовые данные могут отображаться для *min*, *max*, *avg* и *sum*. Для *count* нечисловые данные будут преобразованы в числовые. |
| *Time period* | Укажите временной период, который будет использоваться для агрегации значений:<br>**Dashboard** - использовать временной период панели;<br>**Widget** - использовать временной период указанного виджета;<br>**Custom** - использовать пользовательский временной период.<br>Этот параметр не будет отображаться, если для *Aggregation function* выбрано "not used". |
| *Widget* | Выберите виджет.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Widget". |
| *From* | Выберите начало временного периода (значение по умолчанию `now-1h`). См. relative time syntax.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Custom". |
| *To* | Выберите конец временного периода (значение по умолчанию `now`). См. relative time syntax.<br>Этот параметр будет отображаться только если для *Time period* выбрано "Custom". |
| *History data* | Брать данные из history или trends:<br>**Auto** - автоматический выбор;<br>**History** - брать данные history;<br>**Trends** - брать данные trends.<br>Этот параметр доступен только для числовых значений элементов данных. Для текстовых значений данные всегда будут браться из history. |

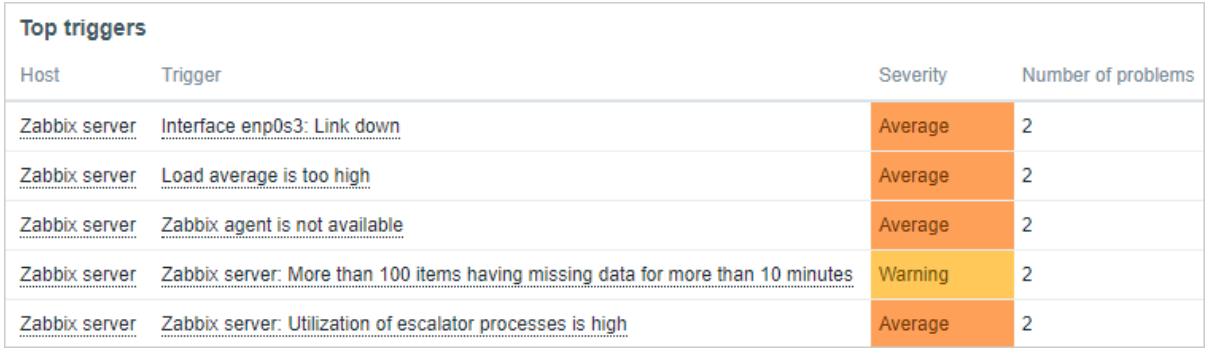
Параметры столбца для текстовых значений:

| | |
|---|---|
| *Highlights* | Укажите регулярные выражения, при совпадении с которыми должен изменяться цвет фона/заливки. |

29 Лучшие триггеры

Обзор

В виджете *Top triggers* можно увидеть триггеры с наибольшим количеством проблем.

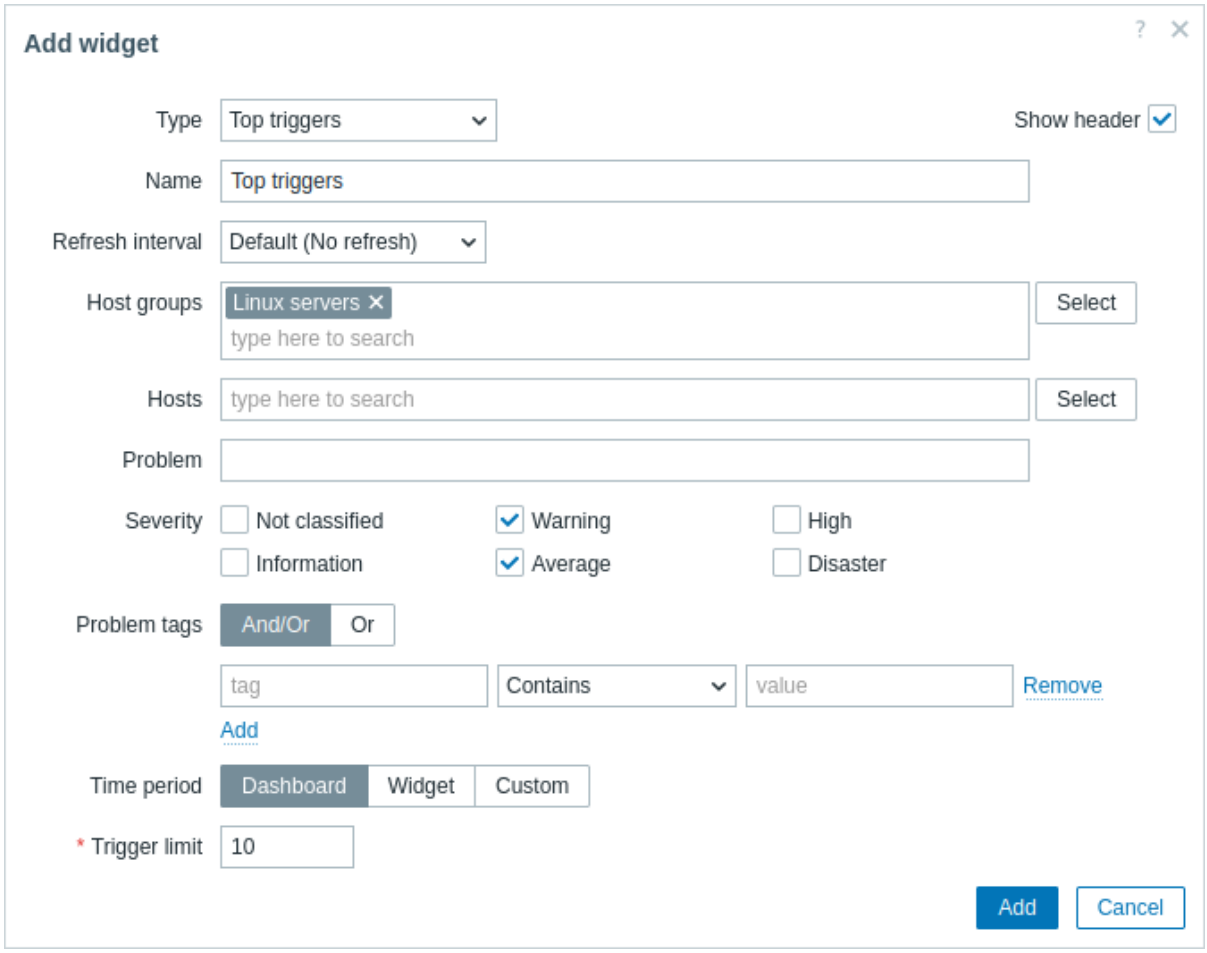



Максимальное количество триггеров, которое можно отобразить, — 1000. При просмотре виджета на панели можно выбрать период времени для отображения данных.

Информация о top triggers также доступна в разделе меню *Reports* → *Top 100 triggers*.

Конфигурация

Чтобы настроить, выберите *Top triggers* в качестве типа виджета:




Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Группы узлов сети* | Выберите группы узлов сети, проблемы которых нужно отображать в виджете.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет предлагаться выпадающий список подходящих групп.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети.<br>Проблемы из этих групп узлов сети будут отображаться в виджете; если группы узлов сети не указаны, будут отображаться проблемы из всех групп узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Узлы сети* | Выберите узлы сети, проблемы которых нужно отображать в виджете.<br>Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет предлагаться выпадающий список подходящих узлов сети.<br>Если узлы сети не указаны, будут отображаться проблемы всех узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Проблема* | Можно просматривать только триггеры для определенных проблем. Для этого введите строку, которая должна совпадать с названием проблемы.<br>Макросы не подставляются. |
| *Серьезность* | Отметьте уровни серьезности триггеров, чтобы отфильтровать триггеры, отображаемые в виджете.<br>Если ни один уровень серьезности не отмечен, будут отображаться все триггеры. |

| | |
|---|---|
| *Теги проблемы* | Укажите теги проблем, которые нужно отображать в виджете.<br>Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Существует** - включить указанные имена тегов;<br>**Равно** - включить указанные имена тегов и значения (с учетом регистра);<br>**Содержит** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Не существует** - исключить указанные имена тегов;<br>**Не равно** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Не содержит** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**И**/**Или** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию *Или*;<br>**Или** - достаточно выполнения одного условия. |
| *Период времени* | Выберите источник данных для периода времени:<br>**Панель** - использовать в качестве источника данных селектор *Период времени*;<br>**Виджет** - использовать в качестве источника данных совместимый виджет, указанный в параметре *Виджет*;<br>**Пользовательский** - использовать в качестве источника данных период времени, указанный в параметрах *С* и *По*; если задано, в правом верхнем углу виджета будет отображаться значок часов, который при наведении мыши показывает установленное время. |
| *Виджет* | Введите или выберите совместимый виджет (*Graph*, *Graph (classic)*, *Graph prototype*) в качестве источника данных для периода времени.<br>Этот параметр доступен, если для *Период времени* задано значение "Виджет". |
| *С* | Введите или выберите начало периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w` и т. д.).<br>Этот параметр доступен, если для *Период времени* задано значение "Пользовательский". |
| *По* | Введите или выберите конец периода времени.<br>Поддерживается синтаксис относительного времени (`now`, `now/d`, `now/w-1w` и т. д.).<br>Этот параметр доступен, если для *Период времени* задано значение "Пользовательский". |
| *Ограничение триггеров* | Задайте количество отображаемых триггеров. Допустимый диапазон значений: 1-1000. |

30 Обзор триггеров

Обзор

В виджете обзора триггеров можно отображать состояния триггеров для группы узлов сети.

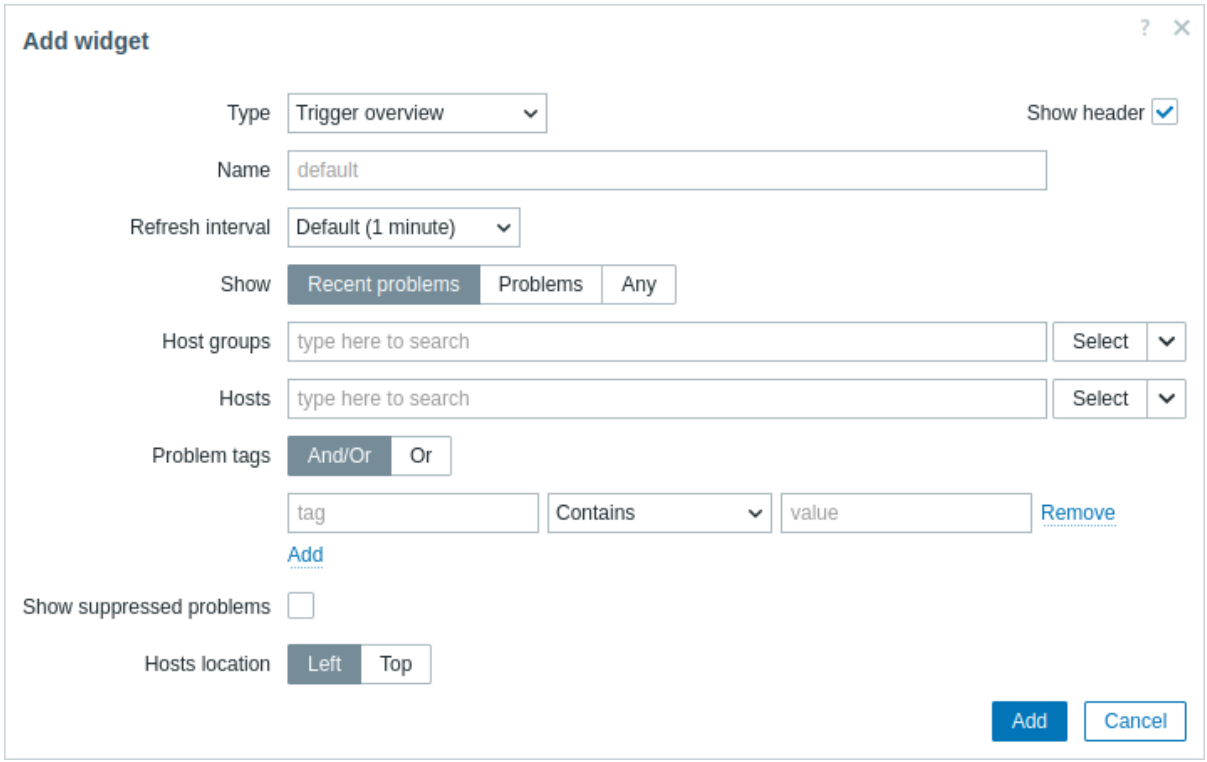
- Состояния триггеров отображаются в виде цветных блоков (цвет блоков для триггеров в состоянии PROBLEM зависит от цвета серьезности проблемы, который можно настроить на экране обновления проблемы). Обратите внимание, что недавние изменения состояния триггера (в течение последних 2 минут) будут отображаться в виде мигающих блоков.
- Серые стрелки вверх и вниз указывают на триггеры, у которых есть зависимости. При наведении курсора отображаются сведения о зависимости.
- Значок флажка указывает на подтвержденные проблемы. Для отображения этого значка все проблемы или решенные проблемы триггера должны быть подтверждены.

При нажатии на блок триггера открываются контекстно-зависимые ссылки на события проблемы триггера, экран подтверждения проблемы, конфигурацию триггера, URL триггера или простой график/список последних значений.

Обратите внимание, что по умолчанию отображается 50 записей (это можно настроить в *Administration* → *General* → *GUI*, используя параметр *Max number of columns and rows in overview tables*). Если записей больше, чем настроено для отображения, в нижней части таблицы выводится сообщение с просьбой указать более точные критерии фильтрации. Пагинация отсутствует. Обратите внимание, что этот предел применяется сначала, до любой последующей фильтрации данных, например по тегам.

Конфигурация

Чтобы настроить, выберите *Trigger overview* в качестве типа:




Помимо параметров, общих для всех виджетов, можно задать следующие специальные параметры:

| | |
|---|---|
| *Show* | Фильтрация триггеров по состоянию триггера:<br>**Recent problems** - *(по умолчанию)* показывать триггеры, которые недавно были или все еще находятся в состоянии PROBLEM (устраненные и неустраненные);<br>**Problems** - показывать триггеры, которые находятся в состоянии PROBLEM (неустраненные);<br>**Any** - показывать все триггеры. |
| *Host groups* | Выберите группы узлов сети.<br>Либо выберите совместимый виджет в качестве источника данных для групп узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени группы будет предлагаться выпадающий список подходящих групп.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |
| *Hosts* | Выберите узлы сети.<br>Либо выберите совместимый виджет или панель в качестве источника данных для узлов сети.<br>Это поле поддерживает автодополнение, поэтому при вводе имени узла сети будет предлагаться выпадающий список подходящих узлов сети.<br>Этот параметр недоступен при настройке виджета на шаблонной панели. |

| | |
|---|---|
| *Problem tags* | Укажите теги для фильтрации триггеров, отображаемых в виджете.<br>Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>**Примечание:** если параметр *Show* установлен в 'Any', будут отображаться все триггеры, даже если указаны теги. Однако, хотя недавние изменения состояния триггера (отображаемые как мигающие блоки) будут обновляться для всех триггеров, сведения о состоянии триггера (цвет серьезности проблемы и наличие подтверждения проблемы) будут обновляться только для триггеров, соответствующих указанным тегам.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включать указанные имена тегов;<br>**Equals** - включать указанные имена и значения тегов (с учетом регистра);<br>**Contains** - включать указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключать указанные имена тегов;<br>**Does not equal** - исключать указанные имена и значения тегов (с учетом регистра);<br>**Does not contain** - исключать указанные имена тегов, если значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию *Or*;<br>**Or** - достаточно выполнения одного условия. |
| *Show suppressed problems* | Установите флажок, чтобы отображать проблемы, которые в противном случае были бы подавлены (не показаны) из-за обслуживания узла сети. |
| *Layout* | Выберите вариант расположения:<br>**Horizontal** - имена узлов сети будут отображаться слева;<br>**Vertical** - имена узлов сети будут отображаться сверху. |

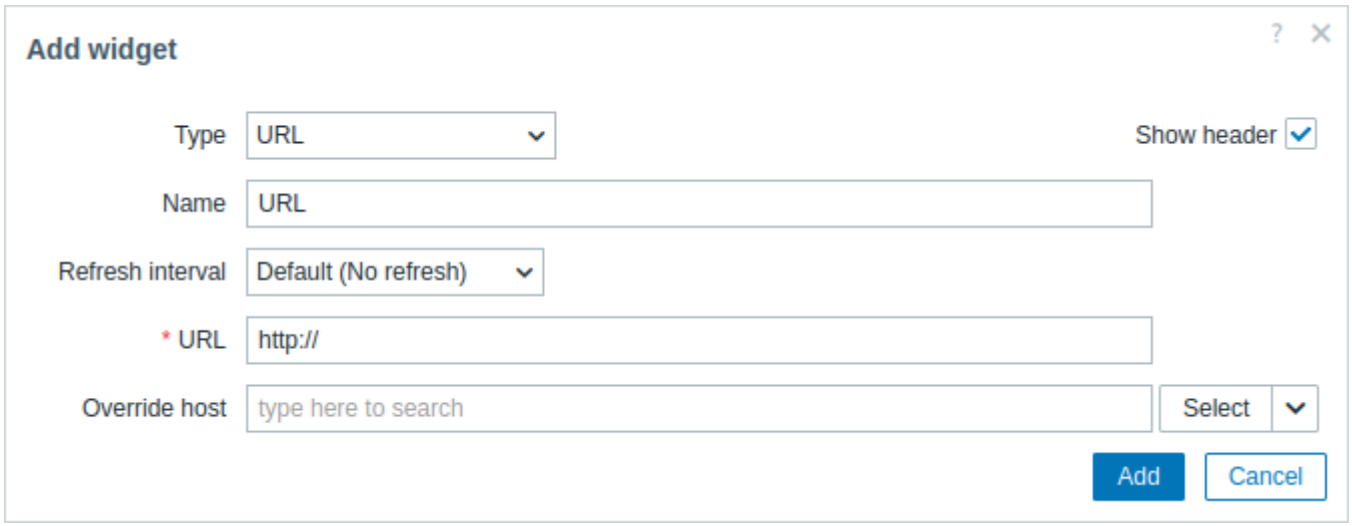
31 URL

Обзор

Этот виджет отображает содержимое, полученное по указанному URL-адресу.

Настройка

Для настройки выберите *URL* в качестве типа:




В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *URL* | Введите URL для отображения (до 2048 символов).<br>Внешние URL должны начинаться с `http://` или `https://`.<br>Внутренние URL поддерживают относительные пути (например, `zabbix.php?action=report.status`).<br>Поддерживаются макросы {HOST.*}. |
| *Переопределение узла сети (Override host)* | Выберите для узлов сети в качестве источника данных совместимый виджет или панель.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |

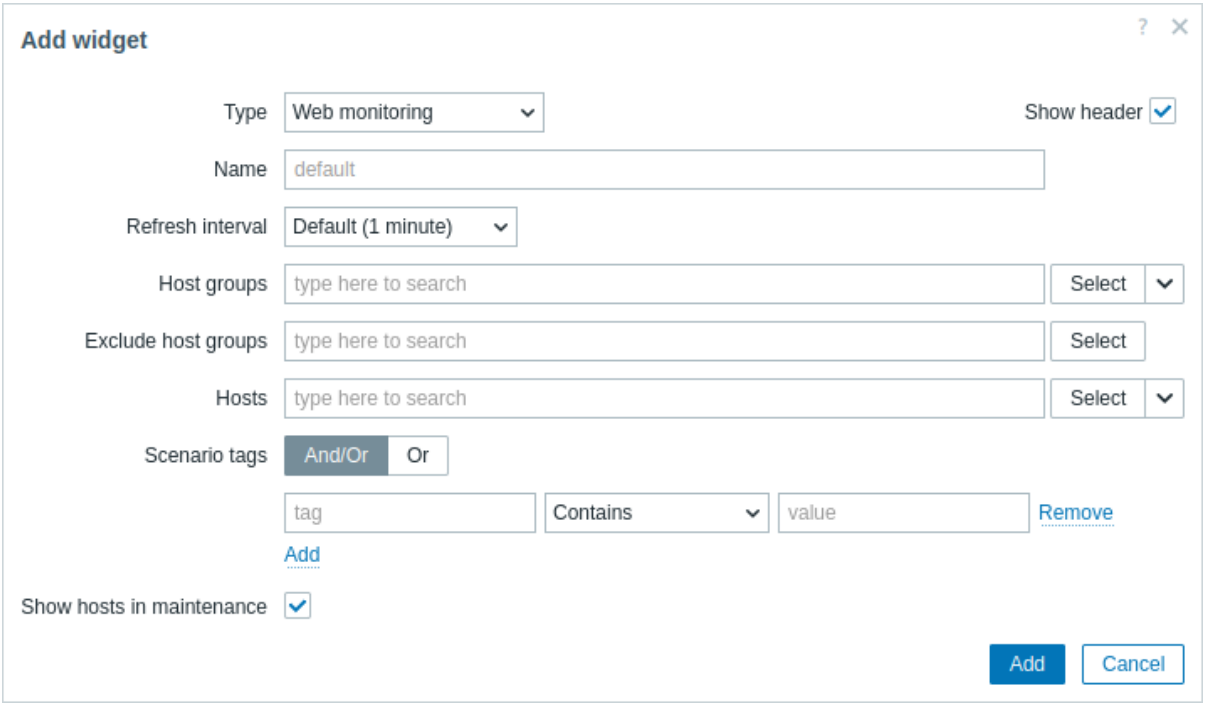
**Attention:**
Браузеры могут не загружать страницу HTTP, включённую в виджет, если доступ к веб-интерфейсу Zabbix осуществляется через HTTPS.



Обзор

Этот виджет отображает сводку состояния активных сценариев веб-мониторинга. Подробную информацию смотрите в разделе Виджет веб-мониторинга.

Настройка




**Note:**
В случае, когда у пользователя не хватает прав для доступа к определённым элементам виджетов, тогда в процессе настройки виджета имя такого элемента будет отображаться как *Недоступно (Inaccessible)*. Это приводит к тому, что вместо «реального» имени элемента появятся имена *Недоступный элемент данных, Недоступный узел сети, Недоступная группа, Недоступная карта сети и Недоступный график*.

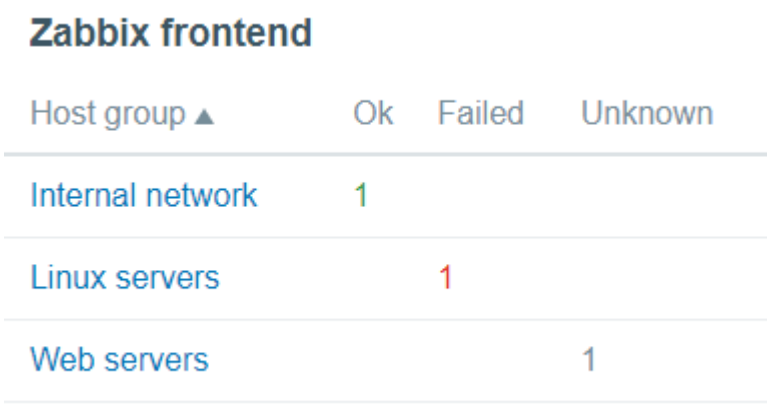
В дополнение к параметрам, которые являются общими для всех виджетов, вы можете установить следующие специфичные опции:

| | |
|---|---|
| *Группы узлов сети (Host groups)* | Выберите группы узлов сети для отображения в виджете.<br>В качестве альтернативы выберите для групп узлов сети в качестве источника данных совместимый виджет.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети.<br>В виджете будут отображаться данные узлов сети из этих групп; если не выбрана ни одна группа, то будут отображаться узлы сети из всех групп.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Исключить группы узлов сети (Exclude host groups)* | Выберите группы узлов сети, которые надо скрыть от виджета.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени группы будет автоматически предложен список совпадающих групп в выпадающем меню.<br>При выборе родительской группы узлов сети неявно выбираются все вложенные группы узлов сети.<br>Данные узлов сети из этих групп не будут отображаться в виджете. Например, узлы сети 001, 002, 003 могут быть в группе «Group A», а узлы сети 002, 003 — также и в группе «Group B». Если мы одновременно выбираем *показать* группу «Group A» и *исключить* группу «Group B», то на панели будут отображены данные только с узла сети 001.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Узлы сети (Hosts)* | Выберите узлы сети для отображения в виджете.<br>В качестве альтернативы выберите для узлов сети в качестве источника данных совместимый виджет или панель.<br>Это поле с функцией автодополнения; таким образом, после начала ввода имени узла сети будет автоматически предложен список совпадающих узлов сети в выпадающем меню.<br>Если не выбраны никакие узлы сети, то будут отображаться все узлы сети.<br>Этот параметр недоступен при настройке виджета на панели шаблона. |
| *Теги сценария (Scenario tags)* | Укажите теги для ограничения количества веб-сценариев, отображаемых в виджете.<br>Можно как включать, так и исключать определённые теги и значения тегов. Можно указать несколько условий. Поиск соответствия имён тегов всегда чувствителен к регистру.<br><br>По каждому условию доступно несколько операторов:<br>**Существует (Exists)** — включить указанные имена тегов;<br>**Равно (Equals)** — включить указанные имена тегов и их значения (с учётом регистра);<br>**Содержит (Contains)** — включить указанные имена тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра);<br>**Не существует (Does not exist)** — исключить указанные имена тегов;<br>**Не равно (Does not equal)** — исключить указанные имена тегов и их значения (с учётом регистра);<br>**Не содержит (Does not contain)** — исключить имена указанных тегов, в которых значения тегов содержат введённую строку (совпадение подстроки, без учёта регистра).<br><br>Для условий имеется два типа вычислений:<br>**И/Или (And/Or)** — все условия должны соблюдаться, условия с одинаковыми именами тегов будут сгруппированы с условием *Или (Or)*;<br>**Или (Or)** — достаточно соблюдения одного условия. |
| *Узлы сети в обслуживании (Show hosts in maintenance)* | Включить в статистику узлы сети в обслуживании.<br>Этот параметр называется *Показывать данные в обслуживании (Show data in maintenance)* при настройке виджета на панели шаблона. |

Виджет мониторинга веб-сайтов

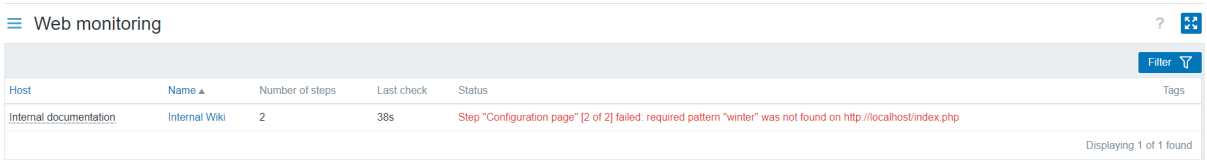
После завершения настройки вы можете просмотреть виджет с отображаемыми в нем данными. Для этого перейдите в *Дашборды* и нажмите на название дашборда, на котором вы создали виджет.

В этом примере вы видите виджет с названием ”Zabbix frontend”, отображающий состояние мониторинга веб-сайтов для трех групп узлов сети: ”Internal network”, ”Linux servers” и ”Web servers.”

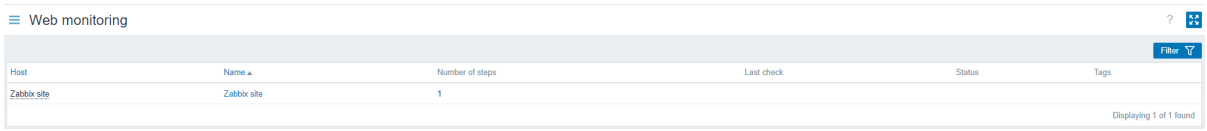



Виджет мониторинга веб-сайтов отображает следующую информацию:

- название виджета; ниже расположены четыре столбца:
  - *Host group* - отображает список групп узлов сети, содержащих узлы сети, для которых настроены веб-сценарии;
  - *Ok* - отображает количество веб-сценариев (зеленым цветом), когда выполняются два условия:
    * Zabbix собрал последние данные для веб-сценария(ев);
    * все шаги, настроенные в веб-сценарии, имеют статус "Ok" *Status*.
  - *Failed* - отображает количество веб-сценариев (красным цветом), в которых есть неудачные шаги:
    * нажмите на имя узла сети, и откроется новое окно; столбец *Status* предоставляет подробную информацию (красным цветом) о шаге, на котором Zabbix не смог собрать данные; а также
    * дает подсказку по параметру, который необходимо исправить в форме настройки.



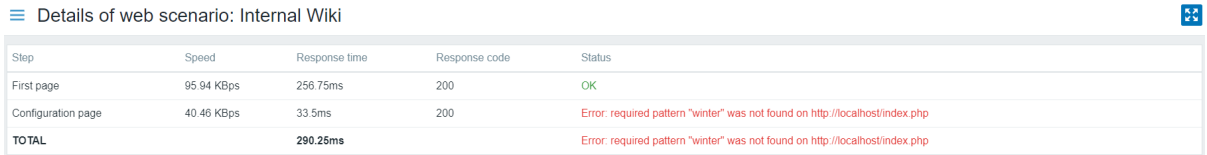

- *Unknown* - отображает количество веб-сценариев (серым цветом), для которых Zabbix не собрал данные и не имеет информации о неудачных шагах.




Просмотр статуса и данных

Кликабельные ссылки в виджете позволяют легко переходить к нужным разделам и быстро получать полную информацию по каждому веб-сценарию. Таким образом, чтобы просмотреть:

- Статус веб-сценария, нажмите на имя группы узлов сети.
- более подробную статистику, нажмите на имя сценария. В этом примере это "Zabbix frontend".
- подробности в случае статуса *Failed*, нажмите на имя группы узлов сети; в открывшемся окне нажмите на имя веб-сценария в столбце *Name*; откроется более подробная информация о настроенных шагах, для которых Zabbix не смог собрать данные.




Теперь вы можете вернуться к форме настройки веб-сценария и исправить параметры.

Чтобы просмотреть подробности в случае статуса *Unknown*, можно повторить те же шаги, что и для *Failed*.

**Attention:**
При первом запуске мониторинга веб-сценарий всегда отображается в состоянии *Unknown*, которое сразу после первой проверки переключается в состояние *Failed* или *Ok*. В случае, когда узел сети контролируется через прокси, изменение статуса происходит в соответствии с частотой сбора данных, настроенной на прокси.

## 2 Мониторинг

Обзор

Меню Monitoring предназначено для отображения данных. Какая бы информация ни была настроена в Zabbix для сбора, визуализации и последующей обработки, она будет отображаться в различных разделах меню Monitoring.

Кнопки режима просмотра

Следующие кнопки, расположенные в правом верхнем углу, являются общими для каждого раздела:

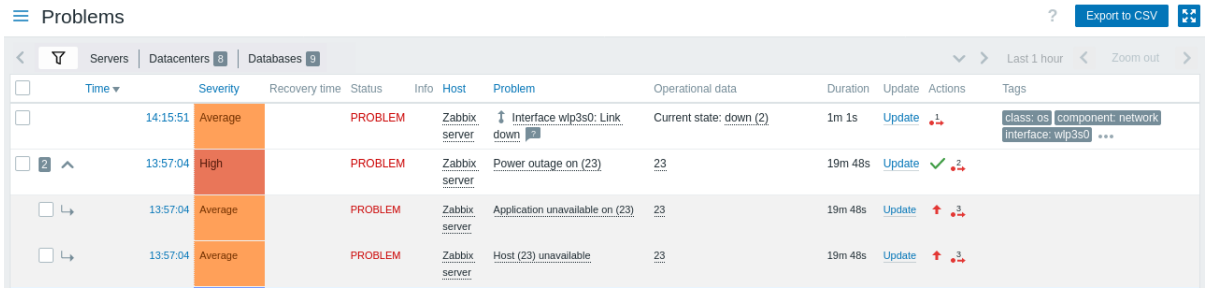
| | |
|---|---|
| | Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.<br>Чтобы выйти из режима киоска, перемещайте курсор мыши, пока не появится кнопка выхода , и нажмите ее. Вы будете возвращены в обычный режим. |

## 1 Проблемы

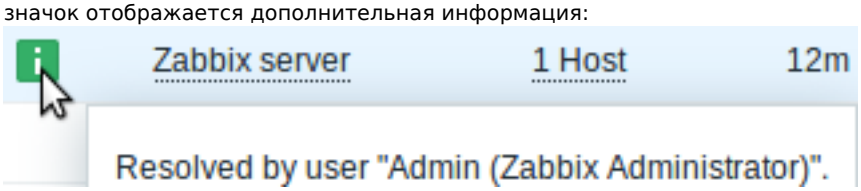
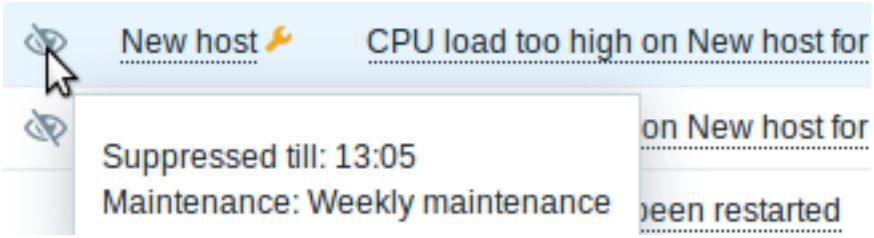
Обзор

В *Мониторинг* > *Проблемы* вы можете видеть, какие проблемы у вас сейчас есть. Проблемы — это триггеры, которые находятся в состоянии "Problem".

По умолчанию все новые проблемы классифицируются как проблемы-причины. Некоторые проблемы можно вручную переклассифицировать как проблемы-симптомы для проблемы-причины. Подробнее см. события-причины и симптомы.




| Column | Description |
|---|---|
| *Checkbox* | Отображаются флажки для выбора проблем.<br>Значки рядом с флажками имеют следующее значение:<br> - количество событий-симптомов для проблемы-причины;<br> - развернуть, чтобы показать события-симптомы;<br> - свернуть, чтобы скрыть события-симптомы;<br> - это событие-симптом. |
| *Time* | Отображается время начала проблемы. |
| *Severity* | Отображается серьезность проблемы.<br>Серьезность проблемы изначально основана на серьезности соответствующего триггера проблемы, однако после возникновения события ее можно изменить с помощью экрана *Обновить проблему* screen. Цвет серьезности проблемы используется как фон ячейки в течение времени существования проблемы. |
| *Recovery time* | Отображается время устранения проблемы. |
| *Status* | Отображается статус проблемы:<br>**Problem** - нерешенная проблема<br>**Resolved** - недавно решенная проблема. Вы можете скрыть недавно решенные проблемы с помощью фильтра.<br>Новые и недавно решенные проблемы мигают в течение 2 минут. Решенные проблемы отображаются в общей сложности 5 минут. Оба этих значения настраиваются в *Администрирование* > *Общее* > *Параметры отображения триггеров*. |

| Column | Description |
|---|---|
| *Info* | Отображается зеленый значок информации, если проблема закрыта глобальной корреляцией или вручную при обновлении проблемы. При наведении указателя мыши на значок отображается дополнительная информация:<br>  <br>Если отображается подавленная проблема (см. параметр *Показывать подавленные проблемы* в фильтре), показывается следующий значок. При наведении указателя мыши на значок отображается дополнительная информация:<br>  |
| *Host* | Отображается узел сети проблемы.<br>Щелчок по имени узла сети открывает меню узла сети. |
| *Problem* | Отображается имя проблемы.<br>Имя проблемы основано на имени соответствующего триггера проблемы.<br>Макросы в имени триггера разрешаются в момент возникновения проблемы, и их значения больше не обновляются.<br>*Примечание*: к имени проблемы можно добавить операционные данные, показывающие некоторые последние значения элементов данных.<br>Щелчок по имени проблемы открывает меню события.<br>При наведении курсора на значок после имени проблемы отображается описание триггера (для тех проблем, у которых оно есть). |
| *Operational data* | Отображаются операционные данные, содержащие последние значения элементов данных.<br>Операционные данные могут быть комбинацией текста и макросов значений элементов данных, если они настроены на уровне триггера. Если на уровне триггера операционные данные не настроены, отображаются последние значения всех элементов данных из выражения.<br>Этот столбец отображается только если в фильтре для параметра *Show operational data* выбрано значение *Separately*. |
| *Duration* | Отображается длительность проблемы.<br>См. также: Отрицательная длительность проблемы |
| *Update* | Щелкните ссылку *Update*, чтобы перейти на экран обновления проблемы, где можно выполнить различные действия с проблемой, включая добавление комментария и подтверждение проблемы. |

| Column | Description |
|---|---|
| *Actions* | История действий, связанных с проблемой, отображается с помощью символических значков:<br> - проблема подтверждена. Этот значок всегда отображается первым.<br> - были добавлены комментарии. Также отображается количество комментариев.<br> - серьезность проблемы была повышена (например, Information > Warning).<br> - серьезность проблемы была понижена (например, Warning > Information).<br> - серьезность проблемы была изменена, но затем возвращена к исходному уровню (например, Warning > Information > Warning).<br> - проблема была подавлена.<br> - подавление проблемы было снято.<br> - были выполнены действия. Также отображается количество действий.<br> - были выполнены действия, по крайней мере одно из них выполняется. Также отображается количество действий.<br> - были выполнены действия, по крайней мере одно из них завершилось неудачно. Также отображается количество действий.<br>При наведении указателя мыши на значки отображаются всплывающие окна с подробностями о действии. См. просмотр подробностей, чтобы узнать больше о значках, используемых во всплывающем окне для выполненных действий. |
| *Tags* | Отображаются теги (если есть).<br>Кроме того, могут отображаться теги из внешней системы обработки заявок (см. параметр *Process tags* при настройке вебхуков). |

Отрицательная длительность проблемы

На самом деле в некоторых распространенных ситуациях возможна отрицательная длительность проблемы, то есть когда время устранения проблемы раньше времени ее создания, например:

- Если какой-либо узел сети контролируется через прокси и происходит сетевая ошибка, из-за которой в течение некоторого времени не поступают данные от прокси, триггер nodata(/host/key) будет сработан сервером. Когда соединение будет восстановлено, сервер получит данные элемента данных от прокси со временем из прошлого. Затем проблема nodata(/host/key) будет решена, и у нее будет отрицательная длительность проблемы;
- Когда данные элемента данных, которые устраняют событие проблемы, отправляются Zabbix sender и содержат метку времени, более раннюю, чем время создания проблемы, также будет отображаться отрицательная длительность проблемы.

**Note:**
Отрицательная длительность проблемы никак не влияет на расчет SLA или отчет о доступности для конкретного триггера; она не уменьшает и не увеличивает время проблемы.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Массовое обновление* - обновить выбранные проблемы, перейдя на экран обновления проблемы

Чтобы использовать этот вариант, установите флажки напротив соответствующих проблем, затем нажмите кнопку *Массовое обновление*.

Кнопки

Кнопка справа предлагает следующую опцию:

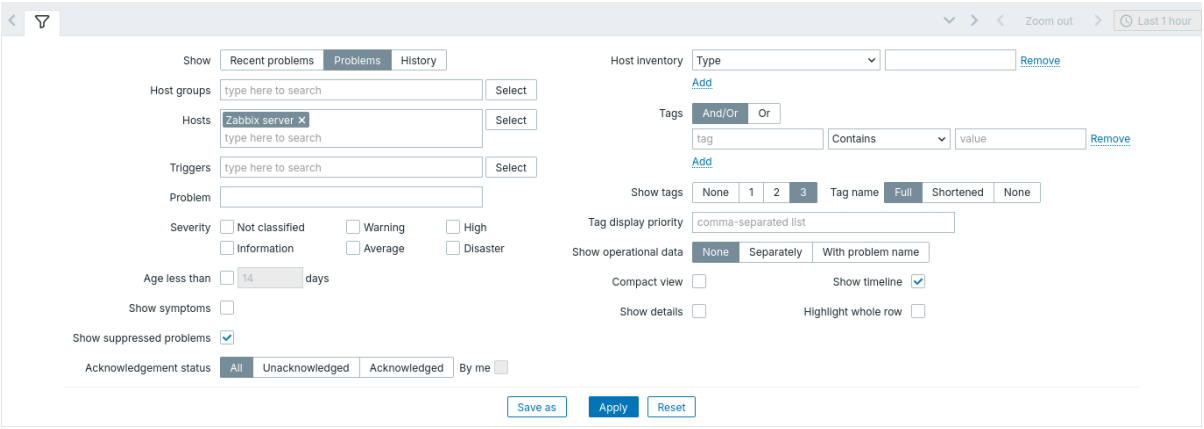
| | |
|---|---|
| Export to CSV | Экспорт содержимого со всех страниц в CSV-файл. |

Кнопки режима просмотра, общие для всех разделов, описаны на странице Monitoring.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только интересующие вас проблемы. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Фильтр расположен над таблицей. Избранные настройки фильтра можно сохранить как вкладки, а затем быстро открывать их, нажимая вкладки над фильтром.



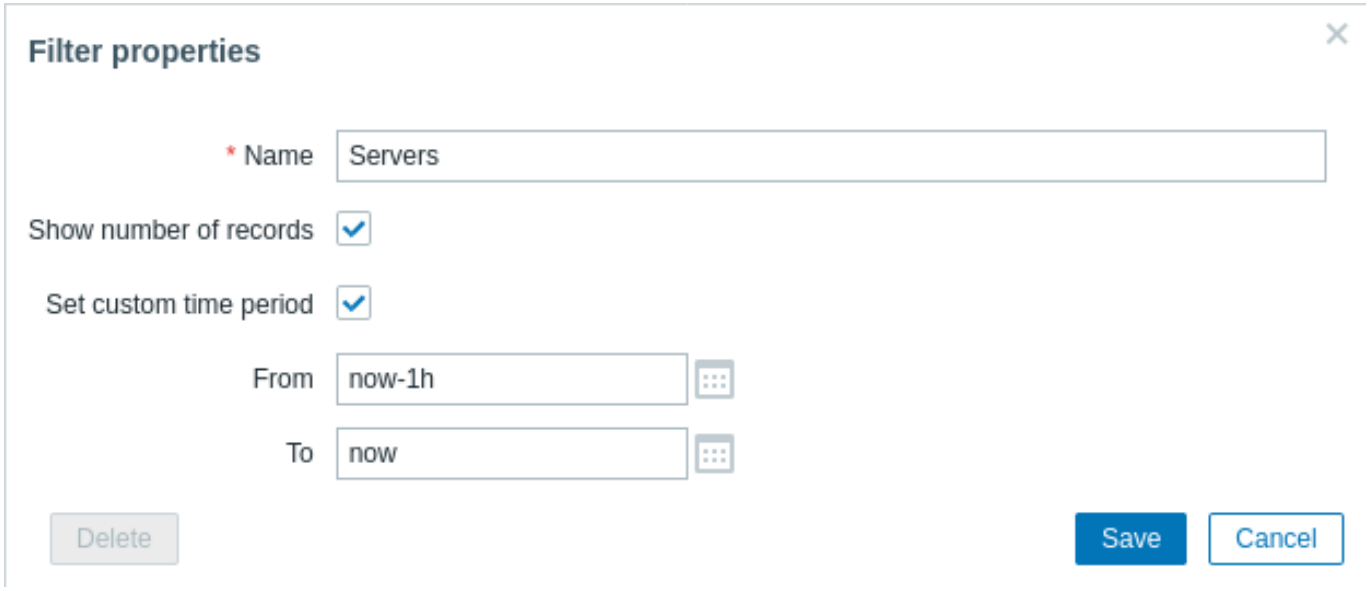

| Parameter | Description |
|---|---|
| *Show* | Фильтрация по статусу проблемы:<br>**Recent problems** - отображаются нерешенные и недавно решенные проблемы (по умолчанию)<br>**Problems** - отображаются нерешенные проблемы<br>**History** - отображается история всех событий |
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети. |
| *Hosts* | Фильтрация по одному или нескольким узлам сети. |
| *Triggers* | Фильтрация по одному или нескольким триггерам. |
| *Problem* | Фильтрация по названию проблемы. |
| *Severity* | Фильтрация по уровню серьезности триггера (проблемы). |
| *Age less than* | Фильтрация по возрасту проблемы. |
| *Show symptoms* | Установите флажок, чтобы отображать в отдельной строке проблемы, классифицированные как симптомы. |
| *Show suppressed problems* | Установите флажок, чтобы отображать проблемы, которые в противном случае были бы подавлены (не показаны) из-за обслуживания узла сети или отдельного подавления проблем. |
| *Acknowledgement status* | Фильтрация для отображения всех проблем, только неподтвержденных проблем или только подтвержденных проблем. Установите дополнительный флажок, чтобы отфильтровать проблемы, которые когда-либо были подтверждены вами. |
| *Host inventory* | Фильтрация по типу и значению инвентарных данных. |

| Parameter | Description |
|---|---|
| *Tags* | Фильтрация по имени и значению тега события. Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, значения которых содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, значения которых содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия<br>При фильтрации указанные здесь теги будут отображаться первыми вместе с проблемой, если это не переопределено списком *Tag display priority* (см. ниже). |
| *Show tags* | Выберите количество отображаемых тегов:<br>**None** - столбец *Tags* в *Monitoring* > *Problems* не отображается<br>**1** - столбец *Tags* содержит один тег<br>**2** - столбец *Tags* содержит два тега<br>**3** - столбец *Tags* содержит три тега<br>Чтобы увидеть все теги проблемы, наведите указатель мыши на значок с тремя точками. |
| *Tag name* | Выберите режим отображения имен тегов:<br>**Full** - имена и значения тегов отображаются полностью<br>**Shortened** - имена тегов сокращаются до 3 символов; значения тегов отображаются полностью<br>**None** - отображаются только значения тегов; имен нет |
| *Tag display priority* | Введите приоритет отображения тегов для проблемы в виде списка тегов, разделенных запятыми (например: `customer,scope,component`). Следует использовать только имена тегов, без значений. Теги из этого списка всегда будут отображаться первыми, переопределяя естественный алфавитный порядок. |
| *Show operational data* | Выберите режим отображения оперативных данных:<br>**None** - оперативные данные не отображаются<br>**Separately** - оперативные данные отображаются в отдельном столбце<br>**With problem name** - добавить оперативные данные к названию проблемы, заключая их в скобки |
| *Compact view* | Установите флажок, чтобы включить компактный вид. |
| *Show details* | Установите флажок, чтобы отображать базовые выражения триггеров проблем.<br>Недоступно, если установлен флажок *Compact view*. |
| *Show timeline* | Установите флажок, чтобы отображать визуальную временную шкалу и группировку.<br>Недоступно, если установлен флажок *Compact view*. |
| *Highlight whole row* | Установите флажок, чтобы выделять всю строку для нерешенных проблем. Для выделения используется цвет серьезности проблемы.<br>*Highlight whole row* недоступно в темах с высокой контрастностью.<br><br>Для версий Zabbix ранее 7.4.3 эта опция поддерживается только когда в стандартных синей и темной темах установлен флажок *Compact view*. |

Вкладки для избранных фильтров

Часто используемые наборы параметров фильтра можно сохранять во вкладках.

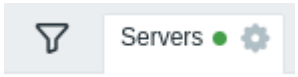
Чтобы сохранить новый набор параметров фильтра, откройте основную вкладку, настройте параметры фильтра, а затем нажмите кнопку *Save as*. В новом всплывающем окне задайте *Filter properties*.




| Parameter | Description |
|---|---|
| *Name* | Имя фильтра, которое будет отображаться в списке вкладок. |
| *Show number of records* | Установите этот флажок, если хотите, чтобы рядом с названием вкладки отображалось количество проблем. |
| *Override time period selector* | Установите этот флажок, чтобы задать для этого набора фильтров определенный период времени по умолчанию. Если этот параметр включен, изменить период времени для этой вкладки можно будет только путем обновления настроек фильтра. Для вкладок без пользовательского периода времени диапазон времени можно изменить, нажав кнопку выбора времени в правом верхнем углу (название кнопки зависит от выбранного интервала: Эта неделя, Последние 30 минут, Вчера и т. д.).<br>Этот параметр доступен только для фильтров в *Monitoring* > *Problems*. |
| *From/To* | Начало и конец периода времени в абсолютном (`Y-m-d H:i:s`) или относительном синтаксисе времени (`now-1d`). Доступно, если установлен флажок *Set custom time period*. |

После сохранения фильтр создается как именованная вкладка фильтра и сразу же активируется.

Чтобы изменить свойства фильтра существующего фильтра, нажмите значок шестеренки рядом с названием активной вкладки.




Примечания:

- Чтобы скрыть область фильтра, нажмите на название текущей вкладки. Чтобы снова открыть область фильтра, еще раз нажмите на название активной вкладки.
- Поддерживается навигация с клавиатуры: используйте стрелки для переключения между вкладками, нажмите *Enter*, чтобы открыть.
- Кнопки слева/справа над фильтром можно использовать для переключения между сохраненными фильтрами. Кроме того, кнопка со стрелкой вниз открывает выпадающее меню со всеми сохраненными фильтрами, и вы можете выбрать нужный.
- Вкладки фильтров можно менять местами перетаскиванием.
- Если настройки сохраненного фильтра были изменены (но не сохранены), после названия фильтра отображается зеленая точка. Чтобы обновить фильтр в соответствии с новыми настройками, нажмите кнопку *Update*, которая отображается вместо кнопки *Save as*.
- Текущие настройки фильтра сохраняются в профиле пользователя. Когда пользователь снова открывает страницу, настройки фильтра остаются прежними.

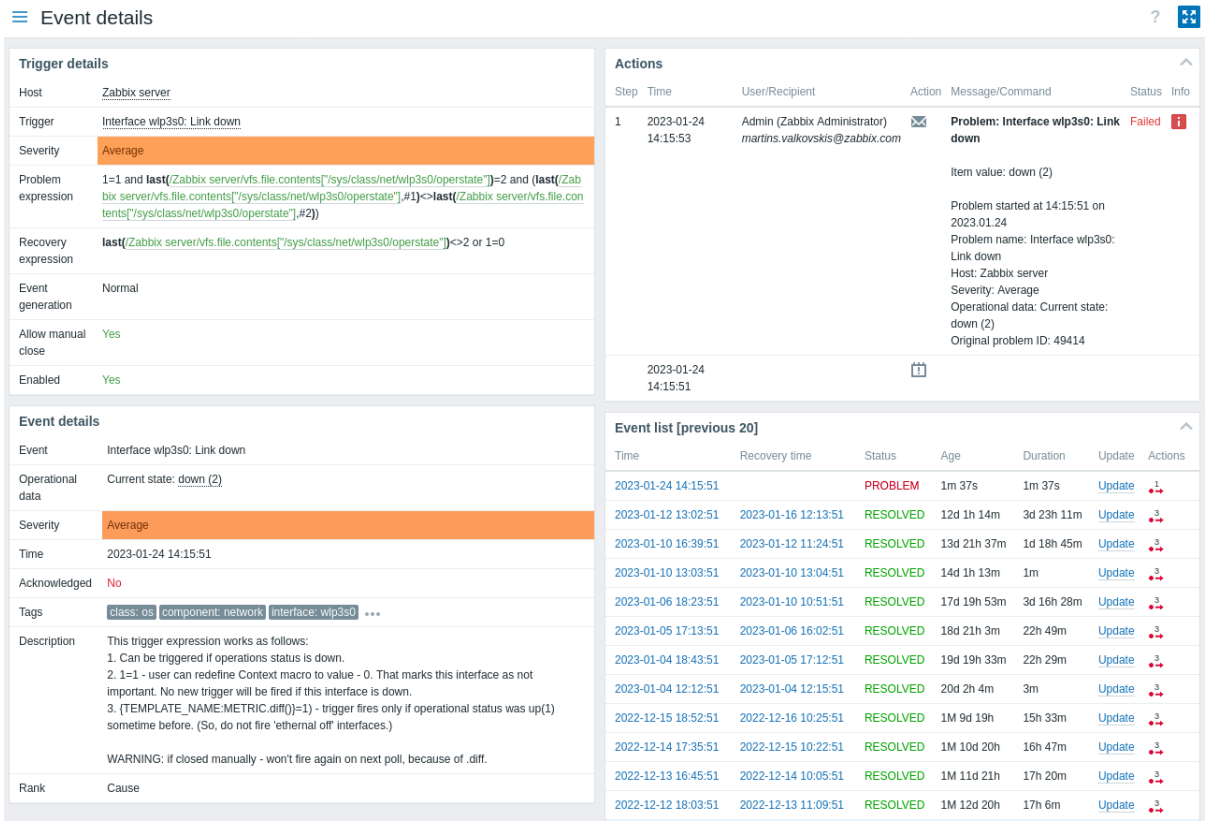
**Note:**
Чтобы поделиться фильтрами, скопируйте и отправьте другим URL активного фильтра. После открытия этого URL другие пользователи смогут сохранить этот набор параметров как постоянный фильтр в своей учетной записи Zabbix.<br> См. также: Параметры страницы.

Кнопки фильтра

| | |
|---|---|
| Apply | Применить выбранные критерии фильтра (без сохранения). |
| Reset | Сбросить текущие настройки фильтра и вернуться к сохраненным параметрам текущей вкладки. В главной вкладке при нажатии этой кнопки все поля фильтра вернутся к состоянию по умолчанию. |
| Save as | Сохранить текущие параметры фильтра в новой вкладке. Доступно только в главной вкладке. |
| Update | Изменить параметры вкладки на заданные в текущий момент. Недоступно в главной вкладке. |

Просмотр подробностей

Время начала проблемы и восстановления в *Monitoring* > *Problems* являются ссылками. При нажатии на них открываются дополнительные сведения о событии.




Обратите внимание, что уровень серьезности проблемы может отличаться для триггера и события проблемы — если он был изменен для события проблемы с помощью экрана *Update problem*.

В списке действий для обозначения типа активности используются следующие значки:

- - событие проблемы сгенерировано.
- - сообщение было отправлено.
- - событие проблемы подтверждено.
- - событие проблемы не подтверждено.
- - был добавлен комментарий.
- - уровень серьезности проблемы был повышен (например, Information > Warning).
- - уровень серьезности проблемы был понижен (например, Warning > Information).

- - уровень серьезности проблемы был изменен, но затем возвращен к исходному уровню (например, Warning > Information > Warning).
- - была выполнена удаленная команда.
- - событие проблемы восстановилось.
- - проблема была закрыта вручную.
- - проблема была подавлена.
- - подавление проблемы было снято.
- - проблема была преобразована в проблему-симптом.
- - проблема была преобразована в проблему-причину.

1 Причины и проблемы с симптомами

Обзор

По умолчанию все новые проблемы классифицируются как проблемы причины. Некоторые проблемы можно вручную переклассифицировать как проблемы-симптомы проблемы причины.

Например, отключение питания может быть фактической первопричиной того, что какой-то узел сети недоступен или какая-то служба не работает. В этом случае проблемы "узел сети недоступен" и "служба не работает" должны быть классифицированы как проблемы-симптомы "отключения питания" - проблемы причины.

Иерархия причина-симптом поддерживает только два уровня. Проблеме, которая уже является симптомом, нельзя назначать "подчиненные" проблемы-симптомы; любые проблемы, назначенные как симптомы проблеме-симптому, станут симптомами той же проблемы причины.

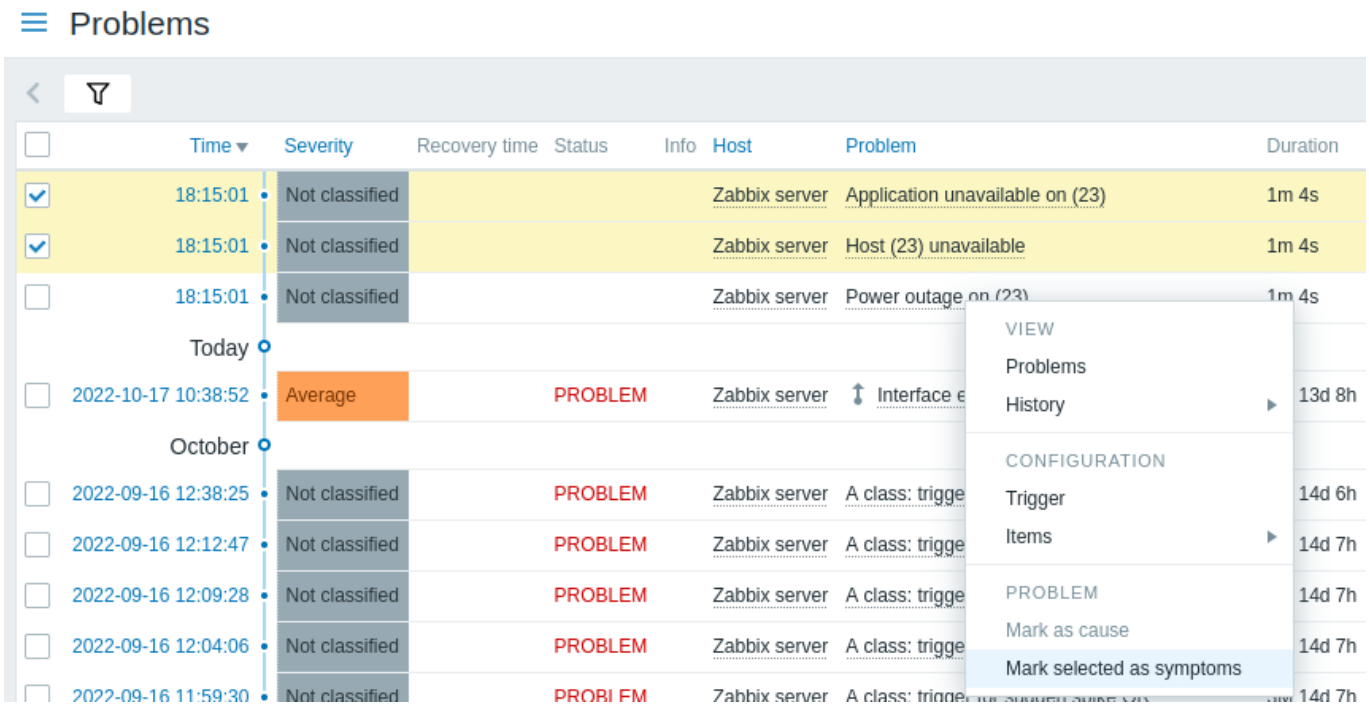
В общих итогах проблем в картах, виджетах панели, таких как *Проблемы по серьезности* или *Проблемные узлы сети*, и т. д. учитываются только проблемы причины. Однако ранжирование проблем не влияет на службы.

Проблема-симптом может быть связана только с одной проблемой причины. Проблемы-симптомы не разрешаются автоматически, если проблема причины разрешена или закрыта.

Конфигурация

Чтобы переклассифицировать проблему как симптомную проблему, сначала выберите ее в списке проблем. Можно выбрать одну или несколько проблем.

Затем перейдите к проблеме-источнику и в ее контекстном меню нажмите пункт *Mark selected as symptoms*.



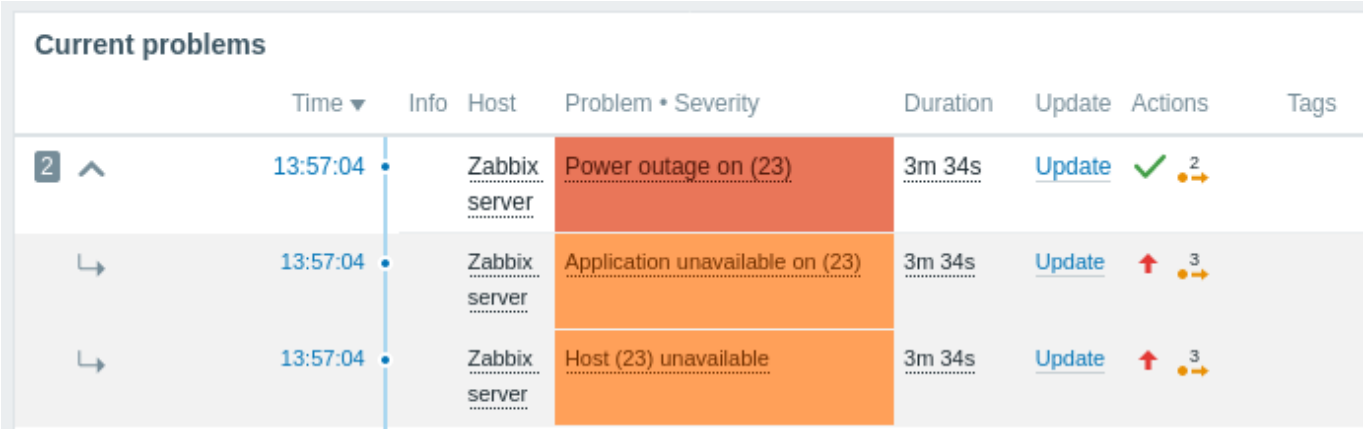
После этого выбранные проблемы будут обновлены сервером до симптомных проблем проблемы-источника.

Пока статус проблемы обновляется, он отображается одним из двух способов:

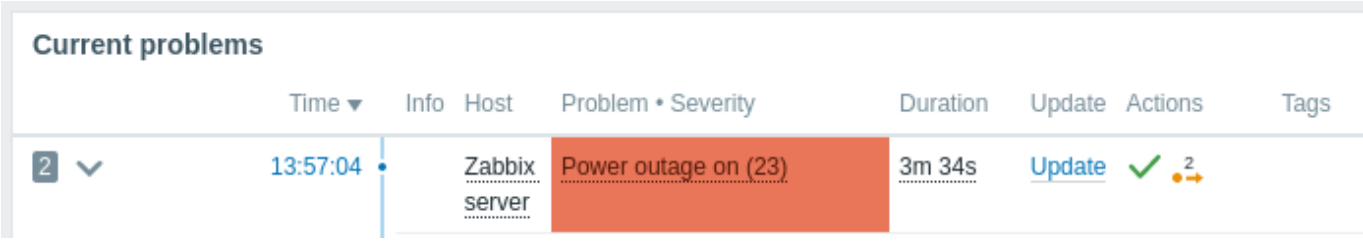
- В столбце Status отображается мигающий статус ”UPDATING”;
- В столбце Info отображается мигающий значок или (это действует, если в фильтре выбраны только *Problems*, и поэтому столбец Status не отображается).

Отображение

Проблемы-симптомы отображаются ниже проблемы-причины и соответствующим образом помечаются в *Monitoring -> Problems* (а также в виджете панели *Problems*) — значком, меньшим шрифтом и другим фоном.




В свернутом виде видна только проблема-причина; наличие проблем-симптомов обозначается числом в начале строки и значком для разворачивания представления.




Также можно дополнительно отображать проблемы-симптомы обычным шрифтом и на отдельной строке. Для этого выберите *Show symptoms* в настройках фильтра или конфигурации виджета.

Возврат к проблеме-источнику

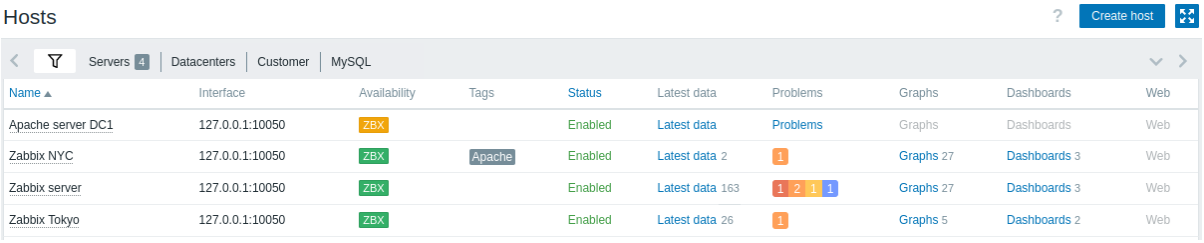
Проблему-симптом можно вернуть обратно к проблеме-источнику. Для этого либо:

- выберите пункт *Пометить как причину* в контекстном меню проблемы-симптома;
- отметьте пункт *Преобразовать в причину* на экране обновления проблемы и нажмите *Обновить* (этот вариант также работает, если выбрано несколько проблем).

**2 узла сети**

Обзор

Раздел *Monitoring* → *Hosts* отображает полный список мониторируемых узлов сети с подробной информацией об интерфейсе узла сети, доступности, тегах, текущих проблемах, статусе (включен/выключен), а также ссылками для удобной навигации к последним данным узла сети, истории проблем, графикам, дашбордам и веб-сценариям.



| Column | Description |
|---|---|
| *Name* | Отображаемое имя узла сети. Щелчок по имени открывает меню узла сети.<br>Оранжевый значок гаечного ключа после имени означает, что этот узел сети находится в обслуживании.<br>Щелкните заголовок столбца, чтобы отсортировать узлы сети по имени по возрастанию или убыванию. |
| *Interface* | Отображается основной интерфейс узла сети. |
| *Availability* | Отображается доступность узла сети по настроенным интерфейсам.<br><br>Значки доступности отражают текущее состояние интерфейса узла сети на сервере Zabbix. Поэтому, если вы отключите узел сети в веб-интерфейсе, его доступность обновится после того, как сервер Zabbix синхронизирует изменения конфигурации. Аналогично, если вы включите узел сети, его доступность обновится после того, как сервер Zabbix синхронизирует изменения конфигурации и опросит узел сети.<br><br>Значки доступности отображают только те типы интерфейсов (Agent, SNMP, IPMI, JMX), которые настроены.<br>При наведении указателя на значок отображается всплывающее окно со списком всех интерфейсов того же типа с подробностями, статусом и ошибками. Для интерфейса Agent во всплывающем окне отображаются интерфейсы (пассивные) и активные проверки. Если у узла сети настроены только активные проверки, значок интерфейса Agent отображается даже в том случае, если у узла сети не настроен интерфейс Agent.<br>Для узлов сети без интерфейсов столбец пуст.<br><br>Состояние отдельного интерфейса узла сети определяется по соединению между включенным элементом данных, использующим этот интерфейс, и узлом сети. Состояние может быть следующим:<br>**Available** - соединение с узлом сети успешно;<br>**Not available** - соединение с узлом сети не удалось (тайм-аут, проблемы с межсетевым экраном и т. д.);<br>**Unknown** - попытка соединения с узлом сети не выполнялась или результат неизвестен.<br>Дополнительные сведения о том, как сервер Zabbix определяет состояние интерфейса, см. в разделах Unknown interface status и Unreachable/unavailable host interface settings.<br><br>Состояние всех интерфейсов одного типа (Agent, SNMP, IPMI, JMX) определяется по тем интерфейсам, которые используются хотя бы одним включенным элементом данных.<br>Состояние обозначается цветом значка:<br>**Green** - все интерфейсы доступны;<br>**Yellow** - как минимум один интерфейс недоступен, и как минимум один доступен или неизвестен;<br>**Red** - все интерфейсы недоступны;<br>**Gray** - как минимум один интерфейс неизвестен, но ни один не является недоступным.<br><br>**Доступность активных проверок.** Если на узле сети включена хотя бы одна активная проверка, активные проверки также влияют на общую доступность интерфейса Agent, как описано выше. Для определения доступности активных проверок heartbeat-сообщения отправляются в потоке активных проверок агента. Частота heartbeat-сообщений задается параметром `HeartbeatFrequency` в конфигурации Zabbix agent или agent 2 (по умолчанию 60 секунд, диапазон 0-3600). Активные проверки считаются недоступными, если heartbeat активной проверки старше чем 2 x HeartbeatFrequency секунд.<br>**Примечание:** агенты Zabbix версии ниже 6.2.x не отправляют heartbeat активных проверок, поэтому доступность их узлов сети остается неизвестной. |
| *Tags* | Теги узла сети и всех связанных шаблонов, при этом макросы не подставлены. |
| *Status* | Статус узла сети - *Enabled* или *Disabled*.<br>Щелкните заголовок столбца, чтобы отсортировать узлы сети по статусу по возрастанию или убыванию. |
| *Latest data* | Щелчок по ссылке откроет страницу *Monitoring* **-** *Latest data* со всеми последними данными, собранными с узла сети.<br>Серым цветом отображается количество элементов данных с последними данными. |

| Column | Description |
|---|---|
| *Problems* | Количество открытых проблем узла сети, отсортированных по серьезности. Цвет квадрата указывает на серьезность проблемы. Число на квадрате означает количество проблем для соответствующей серьезности.<br>Щелчок по значку откроет страницу *Monitoring - Problems* для текущего узла сети.<br>Если у узла сети нет проблем, отображается ссылка на раздел *Problems* для этого узла сети в виде текста.<br>Используйте фильтр, чтобы выбрать, включать ли подавленные проблемы (по умолчанию не включаются). |
| *Graphs* | Щелчок по ссылке отобразит графики, настроенные для узла сети. Количество графиков отображается серым цветом.<br>Если у узла сети нет графиков, ссылка отключена (серый текст), и число не отображается. |
| *Dashboards* | Щелчок по ссылке отобразит дашборды, настроенные для узла сети. Количество дашбордов отображается серым цветом.<br>Если у узла сети нет дашбордов, ссылка отключена (серый текст), и число не отображается. |
| *Web* | Щелчок по ссылке отобразит веб-сценарии, настроенные для узла сети. Количество веб-сценариев отображается серым цветом.<br>Если у узла сети нет веб-сценариев, ссылка отключена (серый текст), и число не отображается. |

Кнопки

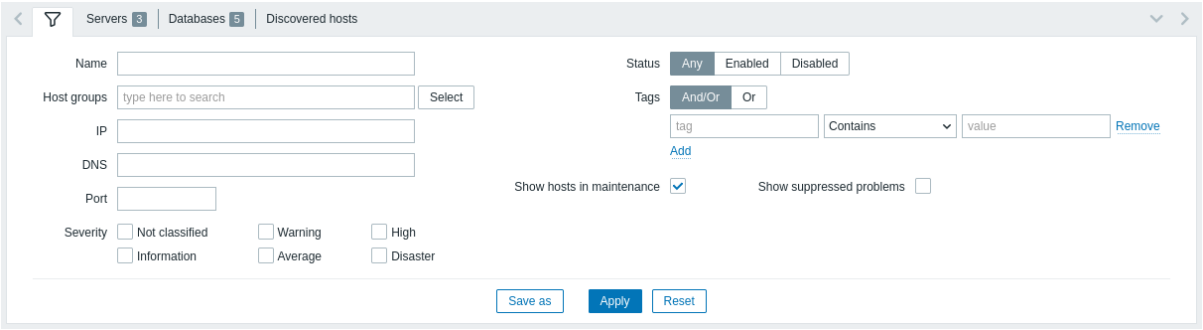
*Create host* позволяет создать новый узел сети. Эта кнопка доступна только пользователям с ролями Admin и Super Admin.

Кнопки режима просмотра, общие для всех разделов, описаны на странице Monitoring.

Использование фильтра

С помощью фильтра можно отображать только те узлы сети, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Фильтр расположен над таблицей. Можно фильтровать узлы сети по имени, группе узлов сети, IP или DNS, порту интерфейса, тегам, серьезности проблемы, статусу (включен/выключен/любой); также можно выбрать, отображать ли подавленные проблемы и узлы сети, которые в данный момент находятся в обслуживании.




| Parameter | Description |
|---|---|
| *Name* | Фильтрация по видимому имени узла сети. |
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети. |
| *IP* | Фильтрация по IP-адресу. |
| *DNS* | Фильтрация по DNS-имени. |
| *Port* | Фильтрация по номеру порта. |
| *Severity* | Фильтрация по серьезности проблемы. По умолчанию отображаются проблемы всех уровней серьезности. Проблемы отображаются, если они не подавлены. |
| *Status* | Фильтрация по статусу узла сети. |

| Parameter | Description |
|---|---|
| *Tags* | Фильтрация по имени и значению тега узла сети. Узлы сети можно фильтровать по тегам уровня узла сети, а также по тегам всех связанных шаблонов, включая вложенные шаблоны.<br>Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны несколько операторов:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, значения которых содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, значения которых содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия |
| *Show hosts in maintenance* | Установите флажок, чтобы отображать узлы сети, находящиеся в обслуживании (отображается по умолчанию). |
| *Show suppressed problems* | Установите флажок, чтобы отображать проблемы, которые в противном случае были бы подавлены (не показаны) из-за обслуживания узла сети или отдельного подавления проблемы. |

Сохранение фильтра

Избранные настройки фильтра можно сохранить как вкладки, а затем быстро открывать их, нажимая соответствующую вкладку над фильтром.

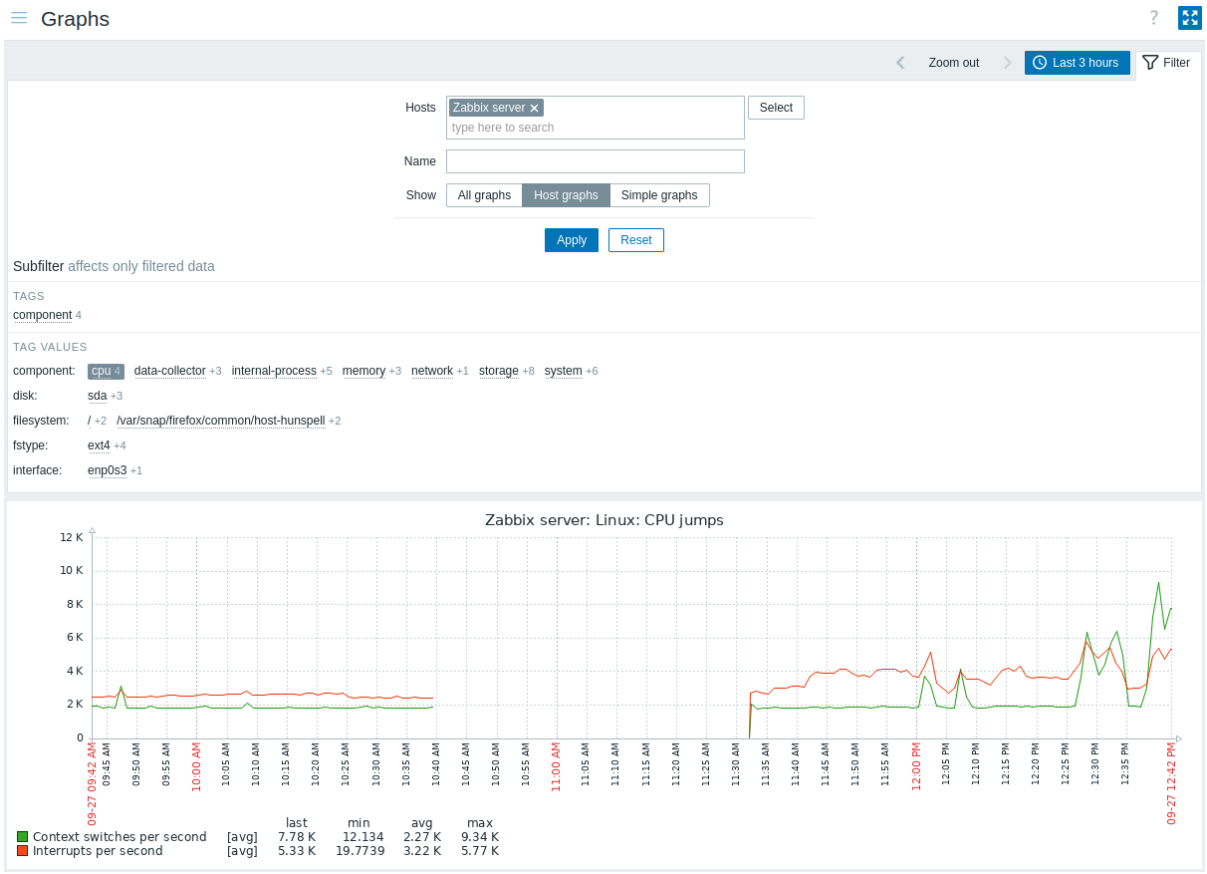
Подробнее о сохранении фильтров.

1 Графики

Обзор

К графикам узла сети можно перейти из *Мониторинг* → *Узлы сети*, нажав Графики для соответствующего узла сети.

Можно отображать любой пользовательский график, настроенный для узла сети, а также любой простой график.




Графики сортируются по:

- имени графика (пользовательские графики)
- имени элемента данных (простые графики)

Графики для отключенных узлов сети также доступны.

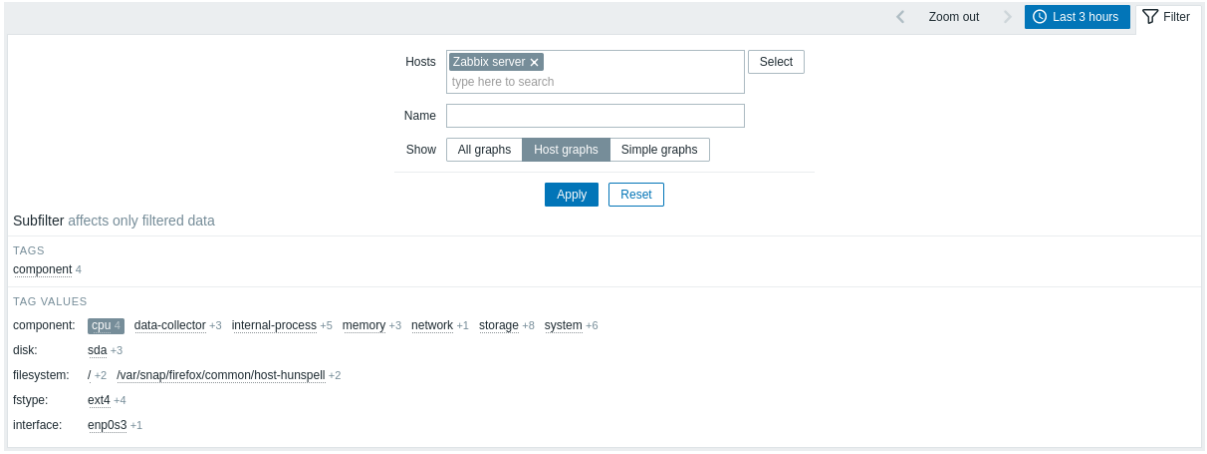
Выбор периода времени

Обратите внимание на выбор периода времени над графиком. Он позволяет выбирать часто используемые периоды одним щелчком мыши.

См. также: *Выбор периода времени*

Использование фильтра

Чтобы просмотреть определенный график, выберите его в фильтре. Фильтр позволяет указать узел сети, имя графика и параметр *Показать* (все/графики узла сети/простые графики).




Если в фильтре не выбран узел сети, графики не отображаются.

Использование подфильтра

Подфильтр позволяет дополнительно изменять фильтрацию, заданную в основном фильтре.

Он содержит кликабельные ссылки для быстрого доступа к связанным графикам. Графики связаны общим объектом - именем или значением тега. Когда нажимают на имя/значение тега, оно выделяется серым фоном, а графики немедленно фильтруются (не нужно нажимать *Применить* в основном фильтре). При нажатии на другое имя/значение тега оно добавляется к отфильтрованным результатам. При повторном нажатии на имя/значение тега фильтрация снимается.

Подфильтры формируются на основе отфильтрованных данных, количество которых ограничено 1000 записей. Если вы хотите видеть больше записей в подфильтре, необходимо увеличить значение параметра *Limit for search and filter results* в разделе *Administration -> General -> GUI*.

В отличие от основного фильтра, подфильтр обновляется вместе с каждым запросом на обновление таблицы, чтобы всегда получать актуальную информацию о доступных вариантах фильтрации и их числовых значениях счетчиков.

Количество отображаемых объектов ограничено 100 по горизонтали. Если их больше, в конце отображается значок с тремя точками; он не является кликабельным. Вертикальные списки (например, теги со значениями) ограничены 20 элементами. Если их больше, отображается значок с тремя точками; он не является кликабельным.

Число рядом с каждым кликабельным объектом показывает количество графиков, которое он имеет в результатах основного фильтра.

После выбора одного объекта числа у других доступных объектов отображаются со знаком плюс, показывая, сколько графиков можно добавить к текущему выбору.

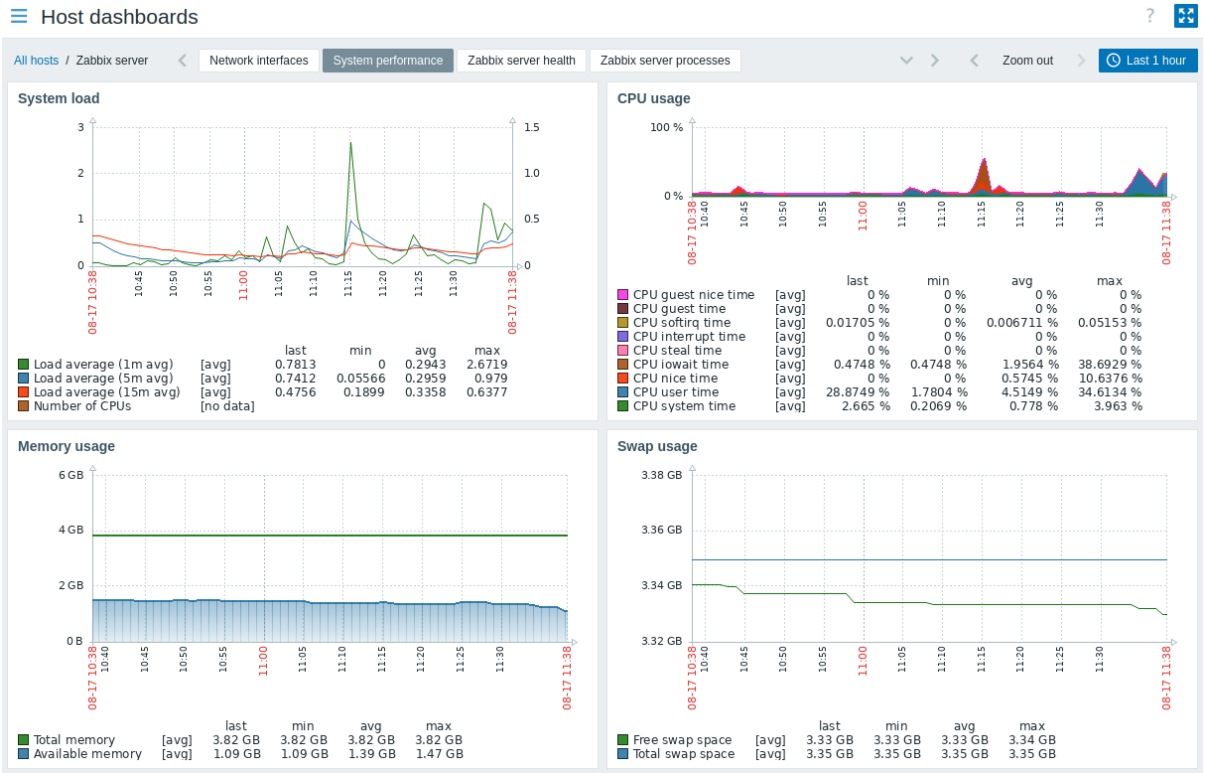
Кнопки

Кнопки режима просмотра, общие для всех разделов, описаны на странице Мониторинг.

2 Панели узла сети

Обзор

Панели узла сети выглядят похоже на глобальные панели; однако у панелей узла сети отсутствует владелец, и они отображают данные только для выбранного узла сети.




При просмотре панелей узла сети можно переключаться между настроенными панелями, нажимая:

- вкладки панелей;
- кнопки со стрелками под заголовком;
- кнопку со стрелкой под заголовком, которая отобразит полный список доступных панелей узла сети.

Чтобы перейти в раздел *Monitoring* → *Hosts*, нажмите ссылку навигации *All hosts* под заголовком в левом верхнем углу.

Конфигурация

Панели узла сети настраиваются на уровне шаблона. После того как шаблон привязан к узлу сети, для этого узла сети создаются панели узла сети. Обратите внимание, что панели узла сети *нельзя* настраивать в разделе *Панели*, который предназначен для глобальных панелей.

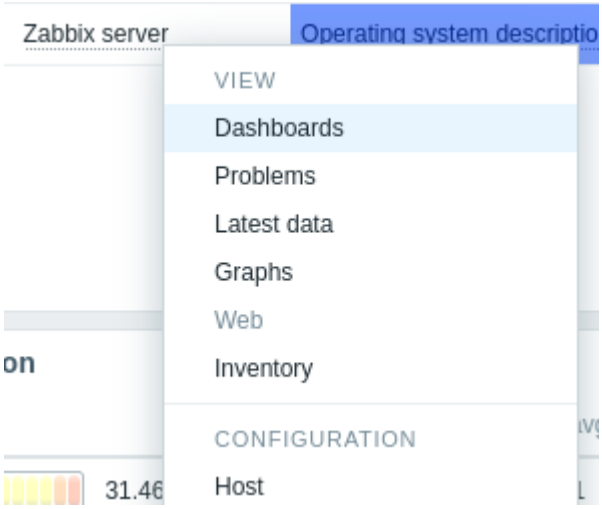
Виджеты панелей узла сети также можно настраивать только на уровне шаблона, за исключением изменения интервала обновления. Кроме того, виджеты панелей узла сети можно копировать только в другие панели узла сети в пределах одного и того же шаблона. Обратите внимание, что виджеты из глобальных панелей *нельзя* копировать в панели узла сети.

**Note:**
До Zabbix 5.2 панели узла сети назывались экранами узла сети. При импорте более старого шаблона, содержащего экраны, импорт экранов будет проигнорирован.

Доступ

К панелям узла сети можно получить доступ:

- после поиска имени узла сети в глобальном поиске (нажмите ссылку *Панели*, указанную в результатах поиска);
- после нажатия на имя узла сети в *Инвентаризация* → *Узлы сети* (нажмите ссылку *Панели*, указанную в обзоре узла сети);
- из меню узла сети, нажав *Панели*.




Обратите внимание, что к панелям узла сети *нельзя* получить прямой доступ в разделе *Панели*, который предназначен для глобальных панелей.

3 Веб-сценарии

Обзор

Информацию о web scenario узла сети можно получить в разделе *Monitoring* → *Hosts*, нажав Web для соответствующего узла сети.

Нажатие на имя узла сети открывает host menu. Данные отключенных узлов сети также доступны. Имя отключенного узла сети отображается красным цветом.

Максимальное количество сценариев, отображаемых на странице, зависит от пользовательской настройки профиля *Rows per page* setting.

По умолчанию отображаются только значения, попадающие в последние 24 часа. Это ограничение введено для улучшения времени первоначальной загрузки больших страниц с последними данными. Вы можете увеличить этот период, изменив значение параметра *Max history display period* в разделе меню *Administration* → *General* → *GUI*.

Имя сценария является ссылкой на более подробную статистику по нему:

Использование фильтра

На странице отображается список всех веб-сценариев выбранного узла сети. Чтобы просмотреть веб-сценарии для другого узла сети или группы узлов сети без возврата на страницу *Monitoring* → *Hosts*, выберите этот узел сети или группу в фильтре. Вы также можете фильтровать сценарии по тегам.

Кнопки

Кнопки режима просмотра, общие для всех разделов, описаны на странице Мониторинг.

**3 Последние данные**

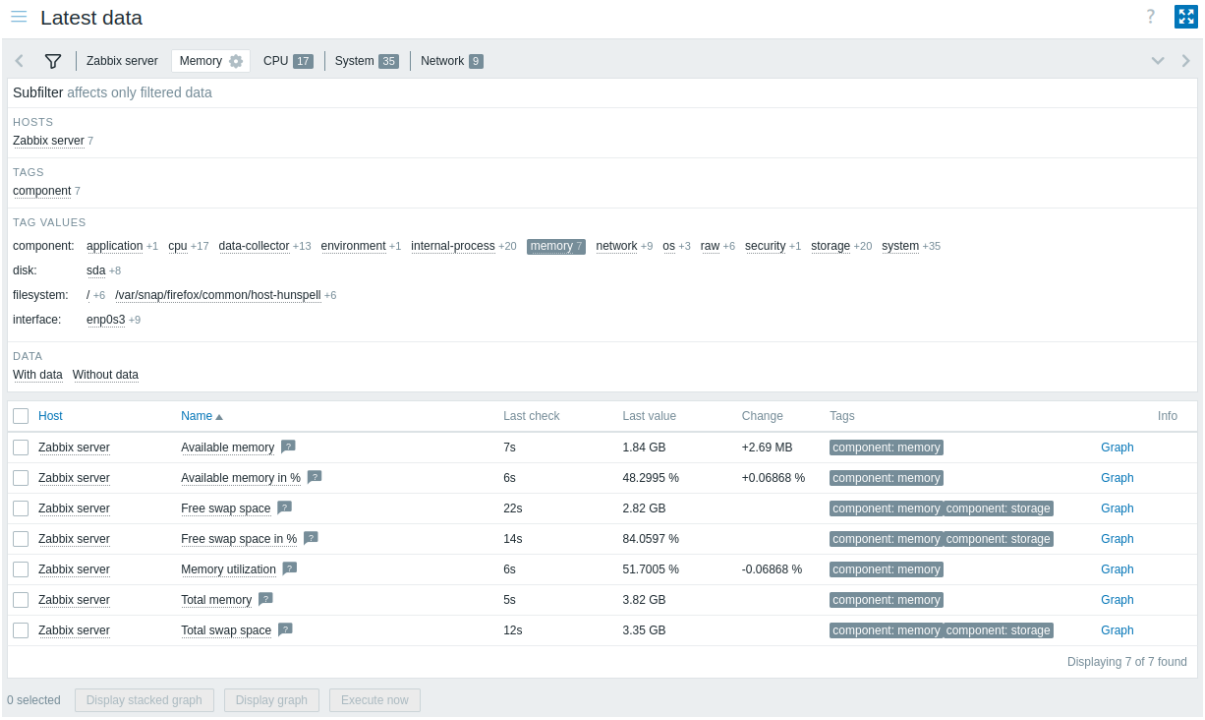
Обзор

Раздел *Monitoring* → *Latest data* отображает последние значения, собранные элементами данных.

Этот раздел содержит следующие элементы:

- Фильтр
- Подфильтр
- Список элементов данных

**Note:**
Подфильтр и список элементов данных отображаются только если задан фильтр и есть результаты для отображения.




| Column | Description |
|---|---|
| *Host* | Имя узла сети, которому принадлежит элемент данных.<br>Щелчок по имени открывает контекстное меню узла сети.<br>Если узел сети находится на обслуживании, после его имени отображается оранжевый значок гаечного ключа .<br>Если узел сети отключен, его имя отображается красным цветом. Обратите внимание, что данные отключенных узлов сети, включая графики и списки значений элементов данных, доступны в разделе *Latest data*. |
| *Name* | Имя элемента данных.<br>Щелчок по имени открывает меню элемента данных.<br>Рядом с именем элемента данных для всех элементов, у которых есть описание, отображается значок вопросительного знака . Наведите указатель на значок, чтобы отобразить всплывающую подсказку с описанием элемента данных. |
| *Last check* | Время, прошедшее с момента последней проверки элемента данных. |
| *Last value* | Последнее значение элемента данных.<br>Значения отображаются с применением преобразования единиц измерения и сопоставления значений.<br>Наведение указателя на значение отображает необработанные данные (обрезанные до 8192 символов).<br>Двоичные значения отображаются как заполнитель (*binary value*) вместо фактического значения.<br>По умолчанию отображаются только значения, полученные за последние 24 часа. Это ограничение улучшает время первоначальной загрузки для больших страниц последних данных; чтобы увеличить его, измените значение параметра *Max history display period* в *Administration → General → GUI*. |
| *Change* | Разница между предыдущим значением и самым последним значением.<br>Для элементов данных с частотой обновления 1 день и более величина изменения никогда не отображается (при настройках по умолчанию). В этом случае последнее значение вообще не будет отображаться, если оно было получено более 24 часов назад. |

| Column | Description |
|---|---|
| *Tags* | Теги, связанные с элементом данных.<br>Теги в списке элементов данных являются кликабельными. Щелчок по тегу включает его в подфильтре, в результате чего список элементов данных отображает только элементы, содержащие этот тег (и любые другие теги, ранее выбранные в подфильтре). Обратите внимание, что после такой фильтрации элементы данных в списке больше не являются кликабельными. Дальнейшее изменение по тегам (например, удаление тегов или указание других фильтров) необходимо выполнять в подфильтре. |
| *Graph/History* | Ссылка на простой график/историю значений элемента данных. |
| *Info* | Дополнительная информация об элементе данных.<br>Если у элемента данных есть ошибки (например, он стал неподдерживаемым), отображается значок информации. Наведите указатель на значок, чтобы увидеть подробности. |

Кнопки

Кнопки режима просмотра, общие для всех разделов, описаны на странице Мониторинг.

Массовые действия

Кнопки под списком предлагают массовые действия с одним или несколькими выбранными элементами:

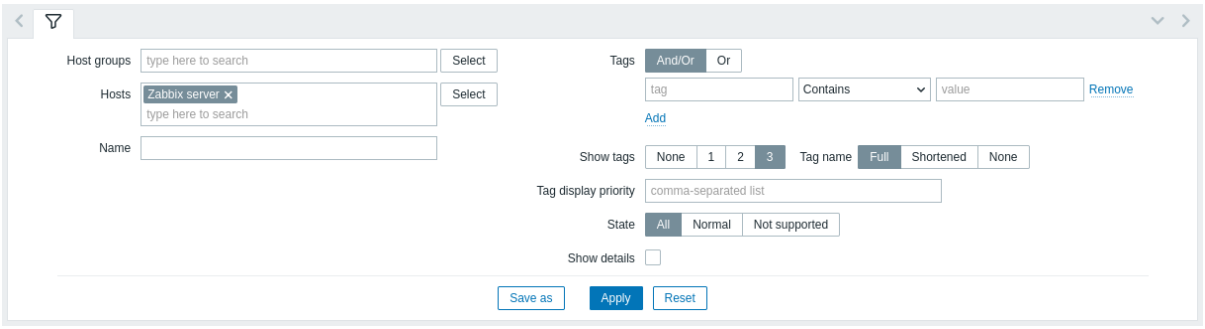
- *Показать составной график* - показать составной ad-hoc graph.
- *Показать график* - показать простой ad-hoc graph.
- *Выполнить сейчас* - немедленно выполнить проверку новых значений элементов данных. Поддерживается только для **пассивных** проверок (см. подробнее). Этот параметр доступен только для узлов сети с доступом на чтение и запись. Доступ к этой опции для узлов сети с правами только на чтение зависит от параметра роли пользователя под названием *Invoke "Execute now" on read-only hosts*.

Чтобы использовать эти параметры, установите флажки напротив соответствующих элементов данных, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те элементы данных, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Значок фильтра расположен над списком элементов данных и субфильтром. Нажмите его, чтобы развернуть фильтр.




Фильтр позволяет сузить список элементов данных по группе узлов сети, узлу сети, имени элемента данных, тегу, состоянию и другим параметрам. Указание родительской группы узлов сети в фильтре неявно выбирает все вложенные группы узлов сети. См. *Monitoring* → *Problems* для получения подробной информации о фильтрации по тегам.

Параметр фильтра *Show details* позволяет расширить отображаемую информацию об элементах данных. Отметьте его, чтобы отображать такие сведения, как интервал обновления элемента данных, настройки истории и трендов, тип элемента данных и ошибки элемента данных (fine/unsupported).

Сохранение фильтра

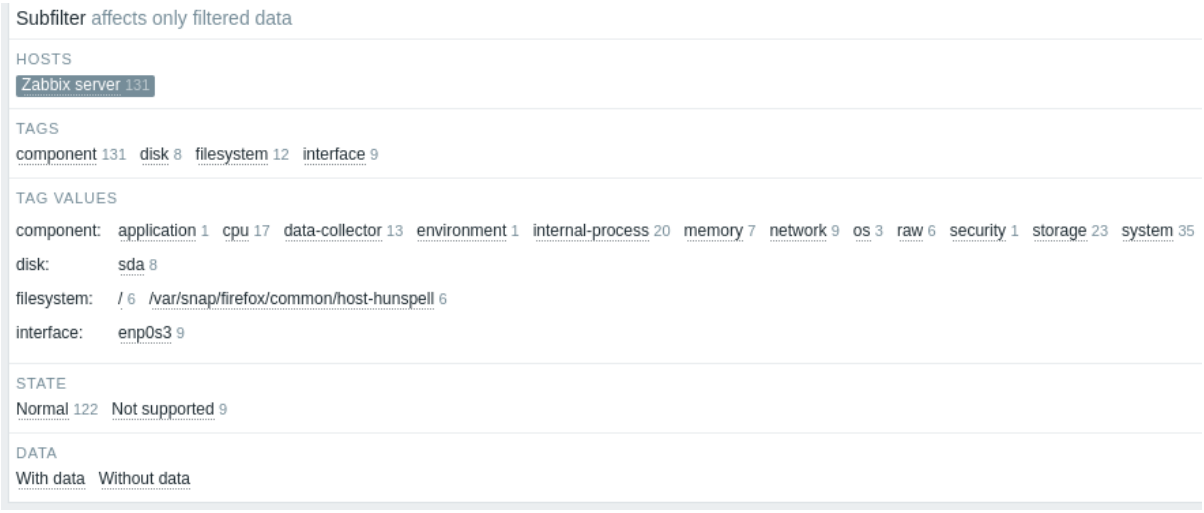
Настройки избранного фильтра можно сохранить в виде вкладок, а затем быстро открывать, щелкнув соответствующую вкладку над фильтром.

Подробнее см. в разделе сохранение фильтров.

Использование подфильтра

Подфильтр позволяет дополнительно изменять фильтрацию, заданную в основном фильтре.

Он содержит кликабельные ссылки для быстрого доступа к связанным элементам данных. Элементы данных связаны по общему объекту - узел сети, имя тега или значение, состояние элемента данных или статус данных. При нажатии на объект он выделяется серым фоном, а элементы данных немедленно фильтруются (нет необходимости нажимать *Применить* в основном фильтре). При нажатии на другой объект он добавляется к отфильтрованным результатам. При повторном нажатии на объект фильтрация снимается.




Подфильтры формируются на основе отфильтрованных данных, количество которых ограничено 1000 записей. Если есть 20 узлов сети, у каждого из которых по 100 элементов данных (то есть всего 2000 записей), в подфильтре будут видны только половина узлов сети. Если вы хотите видеть больше записей в подфильтре, необходимо увеличить значение параметра *Limit for search and filter results* в разделе *Administration -> General -> GUI*.

В отличие от основного фильтра, подфильтр обновляется при каждом запросе обновления таблицы, чтобы всегда иметь актуальную информацию о доступных вариантах фильтрации и их счетчиках.

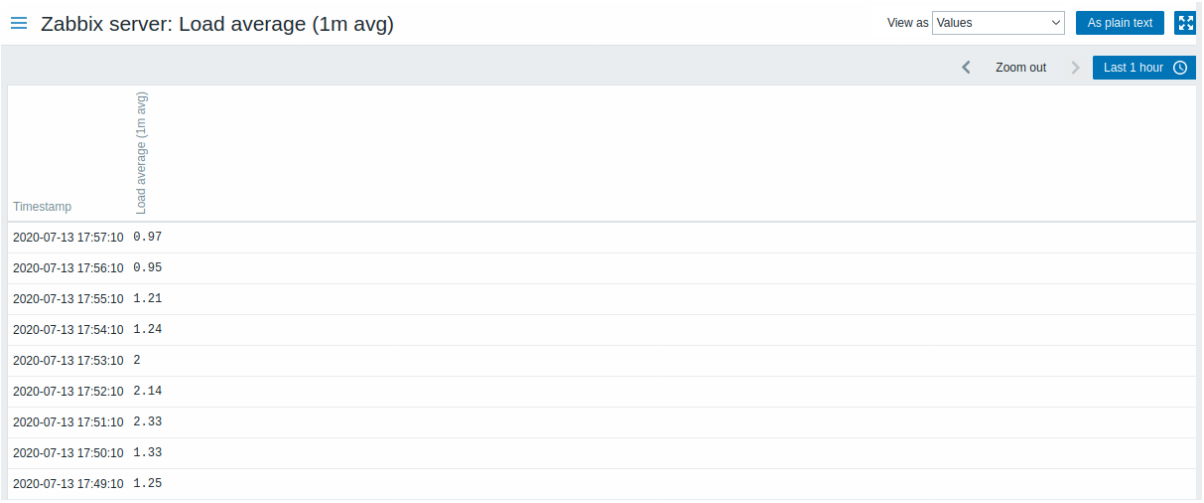
Для каждой группы объектов (узлы сети, теги, значения тегов и т. д.) отображается до 10 строк объектов. Если объектов больше, этот список можно развернуть, чтобы отобразить максимум 1000 записей (значение SUBFILTER_VALUES_PER_GROUP в frontend definitions), нажав на значок с тремя точками ••• в конце списка. Для *Tag values* список можно развернуть, чтобы отобразить максимум 200 имен тегов с соответствующими значениями. Обратите внимание, что после полного разворачивания список нельзя свернуть.

Число рядом с каждым кликабельным объектом показывает количество элементов данных, сгруппированных в нем (на основе результатов основного фильтра). Когда объект выбран, числа у других доступных объектов отображаются со знаком плюс, указывающим, сколько элементов данных может быть добавлено к текущему выбору. Объекты без элементов данных не отображаются, если только они не были выбраны в подфильтре ранее.

Графики и история

Столбец *Graph/History* в списке элементов данных предлагает следующие ссылки:

- **History** - для всех текстовых элементов данных, ведущая к спискам (*Values/500 latest values*), отображающим историю предыдущих значений элемента данных.
- **Graph** - для всех числовых элементов данных, ведущая к простому графику. Обратите внимание, что при отображении графика в раскрывающемся списке в правом верхнем углу также доступна возможность переключиться на *Values/500 latest values*.



Значения, отображаемые в этом списке, являются необработанными, то есть постобработка не применяется.

**Note:**
Общее количество отображаемых значений определяется параметром *Limit for search and filter results*, заданным в *Administration* → *General* → *GUI*.

## 4 Карты

Обзор

В разделе *Monitoring* → *Maps* вы можете настраивать, управлять и просматривать сетевые карты.

Когда вы открываете этот раздел, вы увидите либо последнюю карту, к которой вы обращались, либо список всех карт, к которым у вас есть доступ.

Все карты могут быть либо общедоступными, либо приватными. Общедоступные карты доступны всем пользователям, тогда как приватные карты доступны только их владельцу и пользователям, с которыми карта была расшарена.

Список карт

Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя карты. Щелкните имя, чтобы просмотреть карту. |
| *Width* | Отображается ширина карты. |
| *Height* | Отображается высота карты. |
| *Actions* | Доступны два действия:<br>**Properties** - задать общие свойства карты<br>**Edit** - открыть сетку для добавления элементов карты |

Чтобы настроить новую карту, нажмите кнопку *Create map* в правом верхнем углу. Чтобы импортировать карту из файла YAML, XML или JSON, нажмите кнопку *Import* в правом верхнем углу. Пользователь, который импортирует карту, будет назначен ее владельцем.

Две кнопки под списком предоставляют дополнительные возможности массового редактирования:

- *Export* - экспортировать карты в файл YAML, XML или JSON
- *Delete* - удалить карты

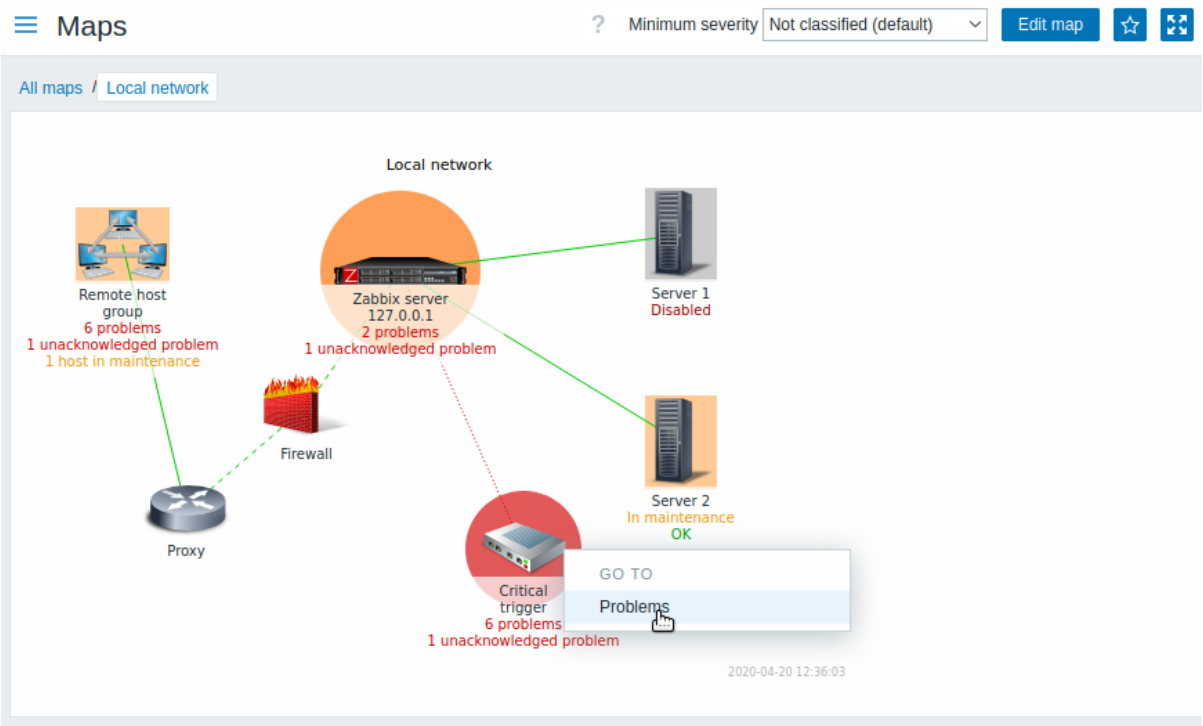
Чтобы использовать эти параметры, отметьте флажки напротив нужных карт, затем нажмите требуемую кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те карты, которые вам интересны. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Просмотр карт

Чтобы просмотреть карту, щелкните ее имя в списке всех карт.




Вы можете использовать выпадающий список в строке заголовка карты, чтобы выбрать минимальный уровень серьезности триггеров проблем для отображения. Серьезность, помеченная как *default*, — это уровень, заданный в конфигурации карты. Если карта содержит под-карту, при переходе к под-карте будет сохранен уровень серьезности карты более высокого уровня (за исключением случая, если это *Not classified*; в этом случае он не будет передан в под-карту).

Подсветка значков

Если элемент карты находится в состоянии проблемы, он выделяется круглым кружком. Цвет заливки кружка соответствует цвету серьезности проблемы. Для элемента будут отображаться только проблемы с уровнем серьезности не ниже выбранного. Если все проблемы подтверждены, вокруг кружка отображается толстая зеленая рамка.

Кроме того:

- узел сети в обслуживании выделяется оранжевым заполненным квадратом. Обратите внимание, что подсветка обслуживания имеет приоритет над подсветкой по серьезности проблемы, если элемент карты является узлом сети.
- отключенный (не контролируемый) узел сети выделяется серым заполненным квадратом.

Подсветка отображается, если установлен флажок *Подсветка значков* в настройке карты.

Маркеры недавних изменений

Красные треугольники, направленные внутрь и расположенные вокруг элемента, указывают на недавнее изменение состояния триггера — то есть изменение, произошедшее в течение последних 30 минут. Эти треугольники отображаются, если установлен флажок *Mark elements on trigger status change* в настройке карты.

Ссылки

Щелчок по элементу карты открывает меню с доступными ссылками. Щелчок по имени узла сети открывает меню узла сети.

Кнопки

Кнопки справа предлагают следующие возможности:

| | |
|---|---|
| Edit map | Перейти к редактированию содержимого карты. |
| | Добавить карту в виджет избранного в разделе Dashboards. |
| | Карта находится в виджете избранного в разделе Dashboards. Нажмите, чтобы удалить карту из виджета избранного. |

Кнопки режима просмотра, общие для всех разделов, описаны на странице Monitoring.

Краткое описание для чтения на картах

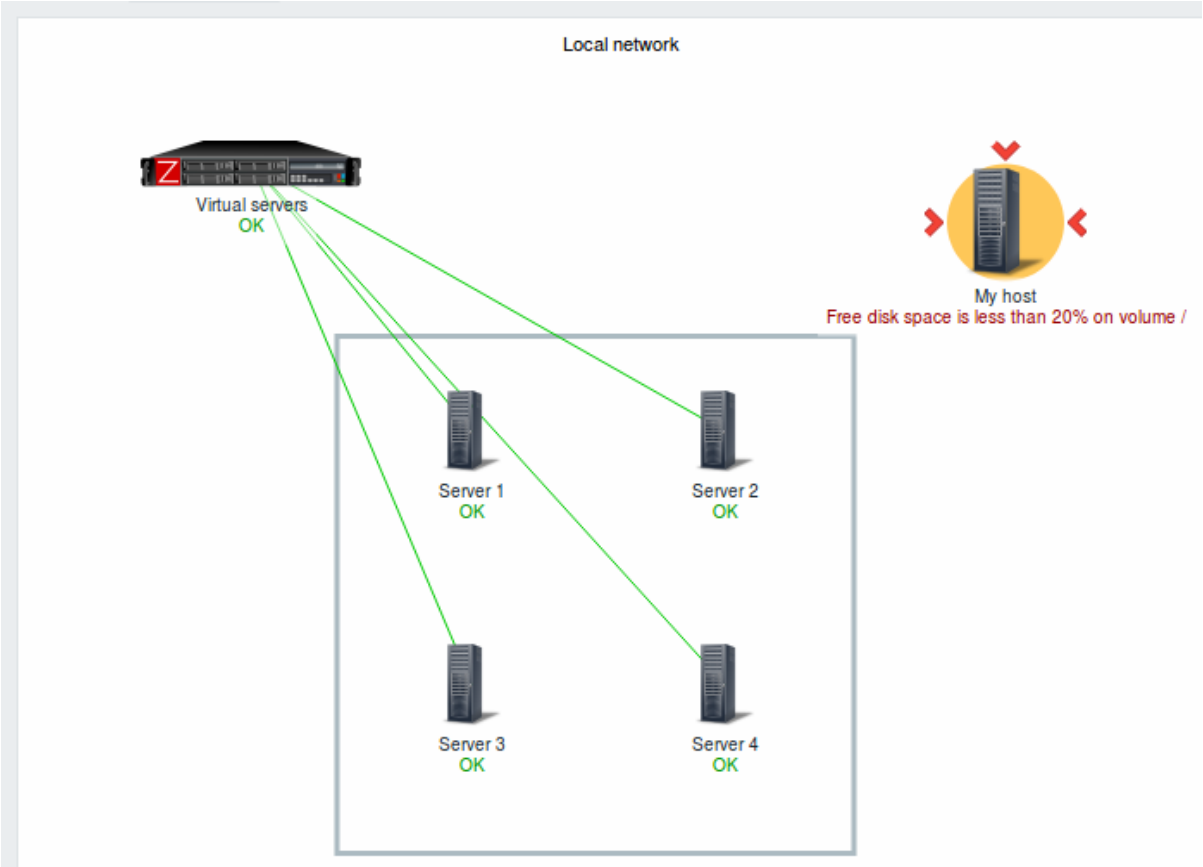
Доступно скрытое свойство "aria-label", которое позволяет считывать информацию о карте с помощью программы чтения с экрана. Доступны как общее описание карты, так и описание отдельных элементов, в следующем формате:

- для описания карты: `<Map name>, <* of * items in problem state>, <* problems in total>.`
- для описания одного элемента с одной проблемой: `<Element type>, Status <Element status>, <Element name>, <Problem description>.`
- для описания одного элемента с несколькими проблемами: `<Element type>, Status <Element status>, <Element name>, <* problems>.`
- для описания одного элемента без проблем: `<Element type>, Status <Element status>, <Element name>.`

Например, такое описание доступно:

```
'Local network, 1 of 6 elements in problem state, 1 problem in total. Host, Status problem, My host, Free
```

для следующей карты:




Ссылка на карту сети

На карты сети можно ссылаться с помощью GET-параметров `sysmapid` и `mapname`. Например,

```
http://zabbix/zabbix/zabbix.php?action=map.view&mapname=Local%20network
```

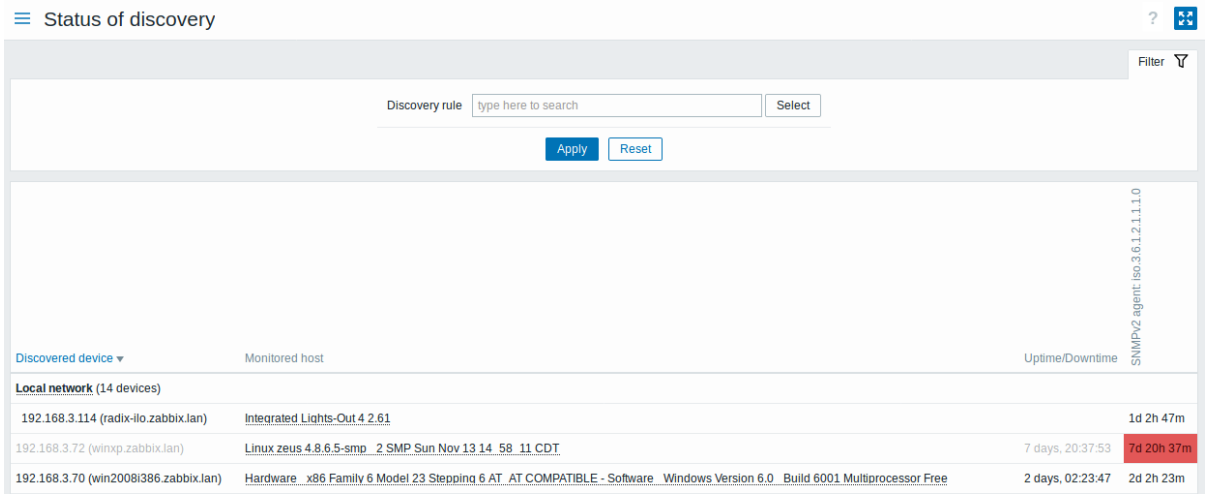
откроет карту с таким именем (Local network).

Если указаны и `sysmapid` (ID карты), и `mapname` (имя карты), то `mapname` имеет более высокий приоритет.

**5 Обнаружение**

Обзор

В разделе *Мониторинг* → *Обнаружение* отображаются результаты сетевого обнаружения. Обнаруженные устройства сортируются по правилу обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Обнаруженное устройство* | Обнаруженные устройства перечисляются и группируются по правилу обнаружения.<br>Щелчок по правилу обнаружения открывает меню правила, содержащее ссылку на форму настройки правила обнаружения. |
| *Контролируемый узел сети* | Если устройство уже контролируется, имя узла сети будет указано в этом столбце.<br>Щелчок по имени узла сети открывает меню узла сети. |
| *Время работы/Время простоя* | В этом столбце отображается длительность состояния обнаружения устройства или его потери после предыдущего обнаружения. |
| *Проверка обнаружения* | Отображается состояние отдельной службы (проверки обнаружения) для каждого обнаруженного устройства. Красная ячейка показывает, что служба недоступна. В ячейке также указывается время работы или простоя службы.<br>Этот столбец отображается только если служба была найдена хотя бы на одном обнаруженном устройстве. |

Кнопки

Кнопки режима просмотра, общие для всех разделов, описаны на странице Мониторинг.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те правила обнаружения, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Если в фильтре ничего не выбрано, отображаются все включенные правила обнаружения. Чтобы выбрать конкретное правило обнаружения для отображения, начните вводить его имя в фильтре. Будут показаны все совпадающие включенные правила обнаружения, доступные для выбора. Можно выбрать более одного правила обнаружения.

## 3 Сервисы

Обзор

Меню ”Услуги” предназначено для функций мониторинга услуг Zabbix.

## 1 Сервисы

Обзор

В этом разделе вы можете увидеть общий статус всех сервисов, которые были настроены в Zabbix на основе вашей инфраструктуры.

Сервис может представлять собой иерархию, состоящую из нескольких уровней других сервисов, называемых ”дочерними” сервисами, которые влияют на общий статус сервиса (см. также обзор функциональности мониторинга сервисов).

Основные категории статуса сервиса — *OK* или *Проблема*, где статус *Проблема* выражается соответствующим названием и цветом серьезности проблемы.

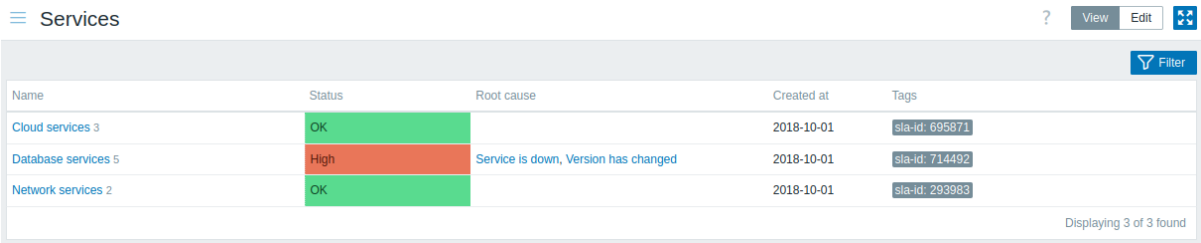
В режиме просмотра можно отслеживать сервисы, их статус и другие сведения, а также настраивать иерархию сервисов в этом разделе (добавлять/редактировать сервисы, дочерние сервисы), переключившись в режим редактирования.

Чтобы переключиться из режима просмотра в режим редактирования (и обратно), нажмите соответствующую кнопку в правом верхнем углу:

- View - просмотр сервисов
- Edit - добавление/редактирование сервисов и дочерних сервисов

Обратите внимание, что доступ к редактированию зависит от настроек роли пользователя.

Просмотр служб




Отображается список существующих служб.

Отображаемые данные:

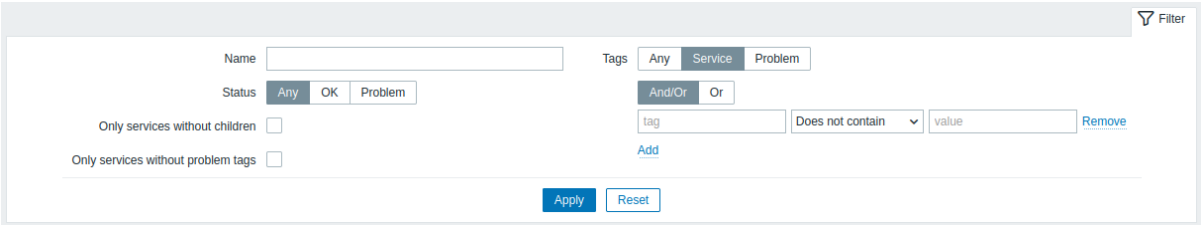
| Parameter | Description |
|---|---|
| *Name* | Имя службы.<br>Имя службы является ссылкой на сведения о службе.<br>Число после имени показывает, сколько дочерних служб имеет служба. |
| *Status* | Статус службы:<br>**OK** - проблем нет<br>**<problem color and severity>** - указывает на проблему и ее серьезность. При наличии нескольких проблем отображаются цвет и серьезность наиболее критичной проблемы. |
| *Root cause* | Перечислены основные проблемы, которые прямо или косвенно влияют на статус службы.<br>Те же проблемы перечислены так, как они возвращаются макросом {SERVICE.ROOTCAUSE} macro.<br>Щелкните имя проблемы, чтобы увидеть ее подробности в разделе *Monitoring* → *Problems*.<br>Проблемы, которые не влияют на статус службы, в список не включаются. |
| *Created at* | Отображается время, когда служба была создана. |
| *Tags* | Отображаются теги службы. Теги используются для идентификации службы в действиях службы и SLA. |

Кнопки

Кнопки режима просмотра, общие для всех разделов, описаны на странице Monitoring.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только интересующие вас сервисы.



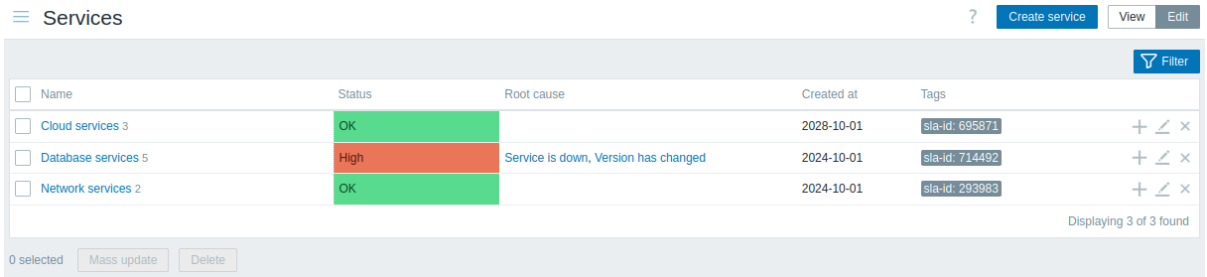

| Parameter | Description |
|---|---|
| *Name* | Фильтрация по имени сервиса. |
| *Status* | Фильтрация по статусу сервиса. |
| *Only services without children* | Установите флажок, чтобы отображать только сервисы без дочерних сервисов. Этот параметр доступен только в режиме редактирования сервисов. |

| Parameter | Description |
|---|---|
| *Only services without problem tags*<br>*Tags* | Установите флажок, чтобы отображать только сервисы без тегов проблем. Этот параметр доступен только в режиме редактирования сервисов.<br>Фильтрация по имени и значению тега сервиса или по имени и значению тега проблемы сервиса (в режиме редактирования сервисов).<br>Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия. |

Редактирование услуг

Нажмите кнопку *Edit*, чтобы перейти в режим редактирования. В режиме редактирования в списке перед элементами появляются флажки, а также следующие дополнительные параметры:

- - добавить дочернюю услугу к этой услуге
- - редактировать эту услугу
- - удалить эту услугу

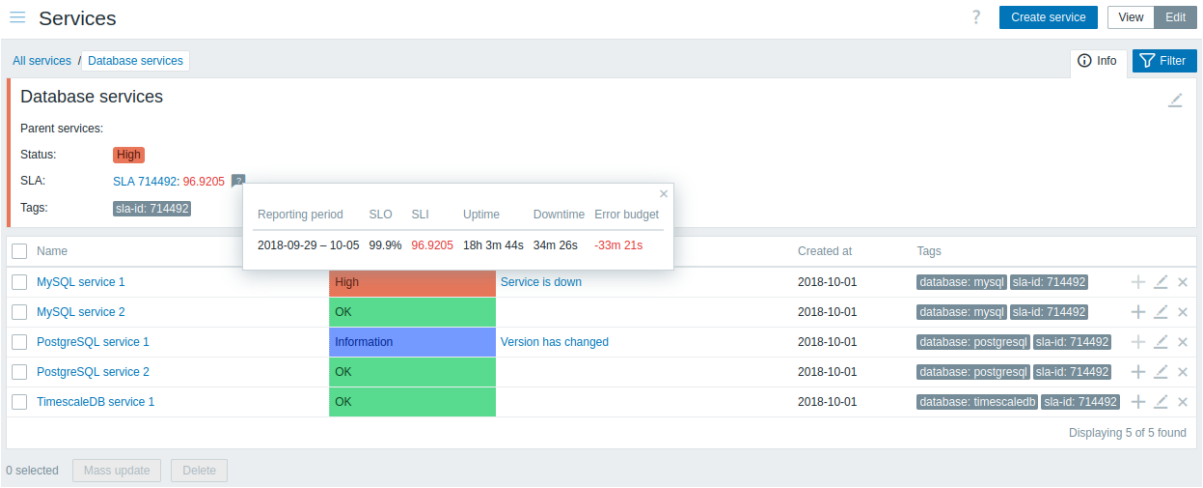



Чтобы настроить новую услугу, нажмите кнопку *Create service* в правом верхнем углу.

Сведения о сервисе

Чтобы открыть сведения о сервисе, нажмите на имя сервиса. Чтобы вернуться к списку всех сервисов, нажмите *All services*.

Сведения о сервисе включают информационный блок и список дочерних сервисов.




Чтобы открыть информационный блок, нажмите вкладку *Info*. Информационный блок содержит следующие поля:

- Имена родительских сервисов (если есть)
- Текущий статус этого сервиса
- Текущие SLA этого сервиса в формате `SLA name:service level indicator`. 'SLA name' также является ссылкой на отчет SLA для этого сервиса. Если навести указатель мыши на информационный блок рядом с индикатором уровня сервиса (SLI), отобразится всплывающий список с подробностями SLI. Индикатор уровня сервиса показывает текущий уровень сервиса в процентах.
- Теги сервиса

Информационный блок также содержит ссылку на конфигурацию сервиса.

Чтобы использовать фильтр для дочерних сервисов, нажмите вкладку *Filter*.

В режиме редактирования список дочерних сервисов дополняется дополнительными параметрами редактирования:

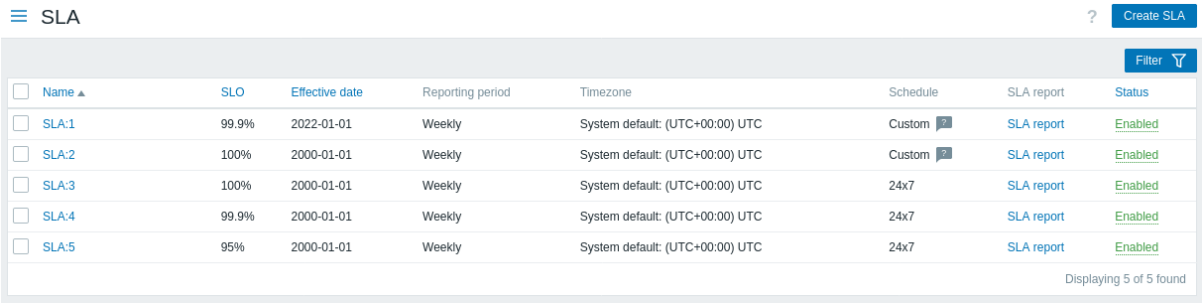
- - добавить дочерний сервис к этому сервису
- - редактировать этот сервис
- - удалить этот сервис

**2 SLA**

Обзор

Этот раздел веб-интерфейса позволяет просматривать и настраивать соглашения об уровне обслуживания (SLA).

SLA




Отображается список настроенных SLA. *Примечание*: будут показаны только SLA, связанные с услугами, доступными пользователю (в режиме только для чтения, если для роли пользователя не включено *Manage SLA*).

Отображаемые данные:

| Parameter | Description |
|---|---|
| *Name* | Отображается имя SLA.<br>Имя является ссылкой на конфигурацию SLA. |
| *SLO* | Отображается целевой уровень обслуживания (SLO). |
| *Effective date* | Отображается дата начала расчета SLA. |

| Parameter | Description |
|---|---|
| *Reporting period* | Отображается период, используемый в отчете SLA, - *daily*, *weekly*, *monthly*, *quarterly* или *annually*. |
| *Time zone* | Отображается часовой пояс SLA. |
| *Schedule* | Отображается расписание SLA - 24x7 или custom. |
| *SLA report* | Щелкните ссылку, чтобы просмотреть отчет SLA для этого SLA. |
| *Status* | Отображается статус SLA - enabled или disabled. |

### 3 Отчет SLA

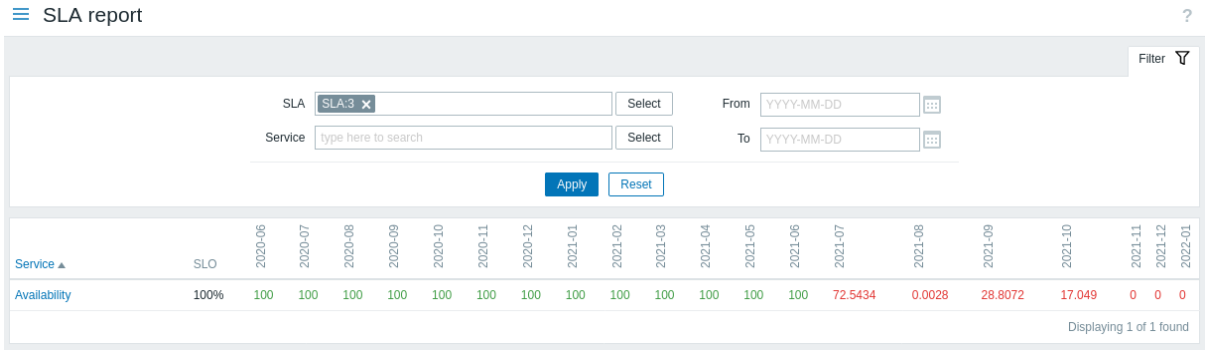
Обзор

Этот раздел позволяет просматривать отчеты SLA, основанные на критериях, выбранных в фильтре.

Отчеты SLA также можно отображать в виде виджета панели мониторинга.

Отчет

Фильтр позволяет выбрать отчет по имени SLA, а также по имени сервиса. Также можно ограничить отображаемый период.

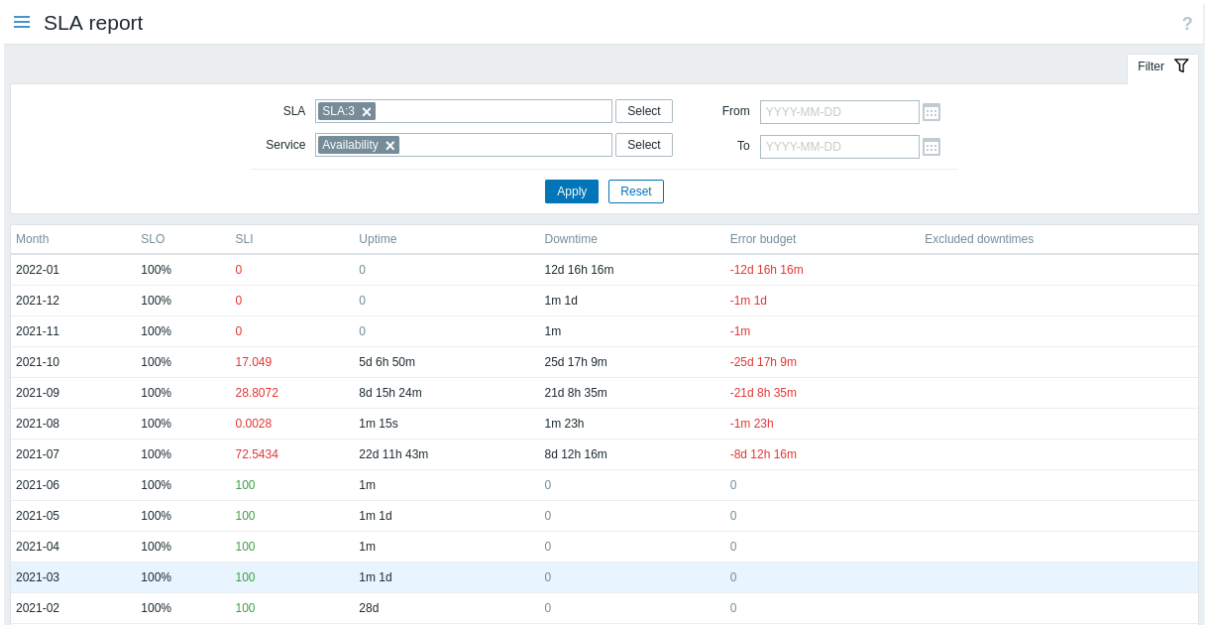



Каждый столбец (период) отображает SLI за этот период. SLI, которые нарушают заданный SLO, выделяются красным цветом.

В отчете отображаются 20 периодов. Можно отобразить максимум 100 периодов, если указаны обе даты: *From* и *To*.

Подробности отчета

Если вы нажмете на имя сервиса в отчете, вы сможете открыть другой отчет, который отображает более подробное представление.




Обратите внимание, что отрицательная длительность проблемы не влияет на расчет SLA или формирование отчетов.

## 4 Инвентарь

Обзор

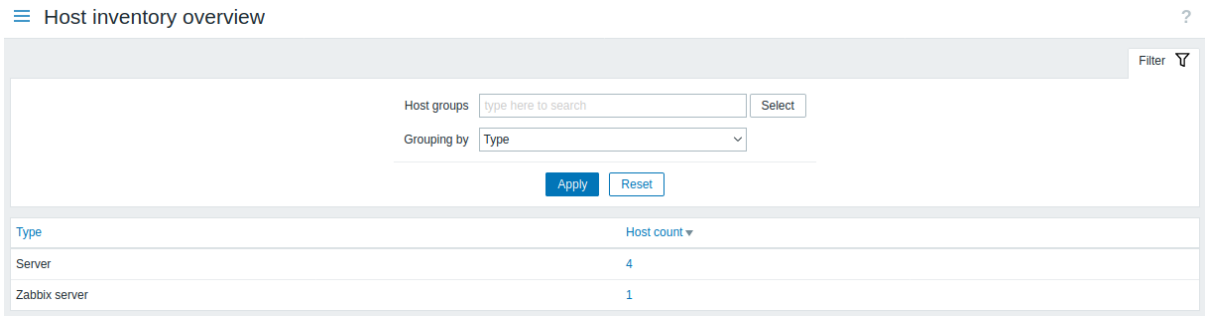
Меню «Инвентаризация» содержит разделы, предоставляющие обзор данных инвентаризации узлов сети по выбранному параметру, а также возможность просматривать подробные сведения об инвентаризации узлов сети.

### 1 Обзор

Обзор

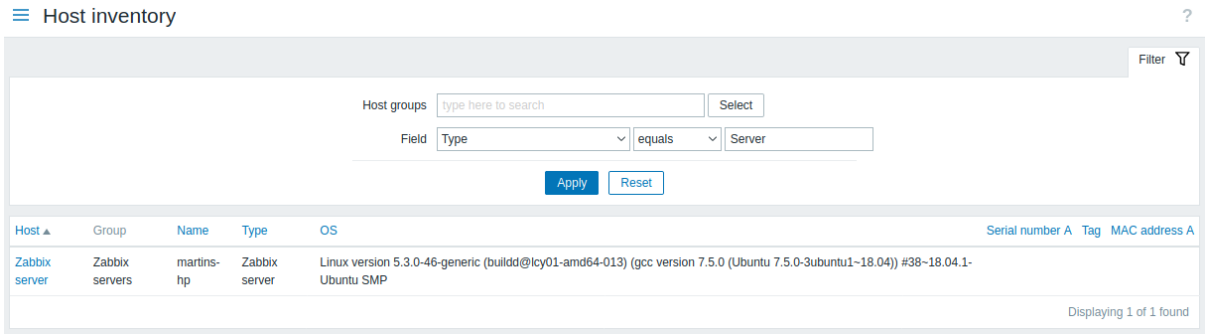
Раздел *Инвентарь* → *Обзор* предоставляет способы получить обзор данных инвентаря узла сети.

Чтобы отобразить обзор, выберите группы узлов сети (или не выбирайте ни одной) и поле инвентаря, по которому нужно отображать данные. Будет показано количество узлов сети, соответствующих каждому значению выбранного поля.




Полнота обзора зависит от того, насколько полно для узлов сети заполнена информация инвентаря.

Числа в столбце *Host count* являются ссылками; они ведут к отфильтрованному списку этих узлов сети в таблице *Host Inventories*.

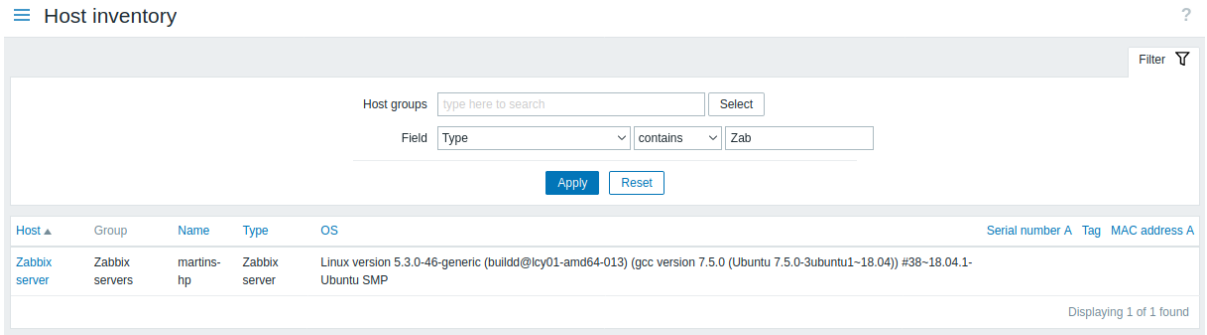



### 2 Узлы сети

Обзор

В разделе *Inventory* → *Hosts* отображаются данные инвентаризации узлов сети.

Вы можете фильтровать узлы сети по группам узлов сети и по любому полю инвентаризации, чтобы отображать только те узлы сети, которые вас интересуют.

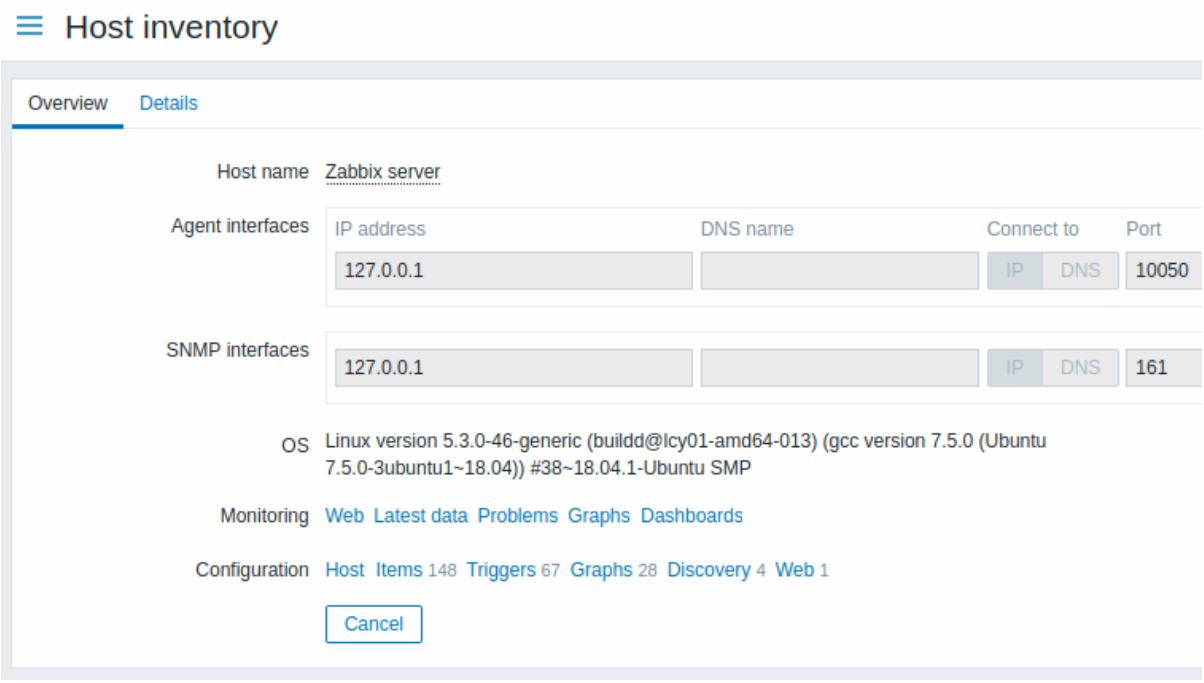



Чтобы отобразить всю инвентаризацию узлов сети, нажмите кнопку *Reset*.

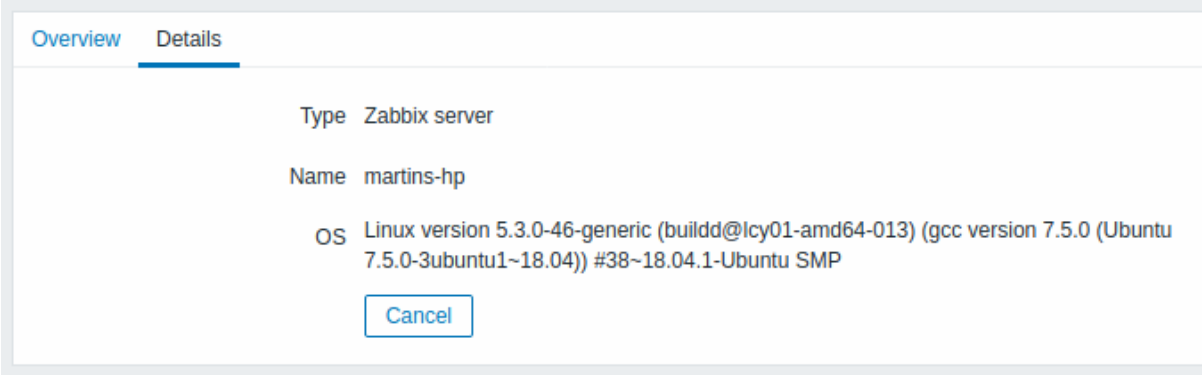
Хотя в таблице отображаются только некоторые ключевые поля инвентаризации, вы также можете просмотреть всю доступную информацию инвентаризации для этого узла сети. Для этого нажмите имя узла сети в первом столбце.

Сведения об инвентаризации

Вкладка **Обзор** содержит общую информацию об узле сети, последние данные мониторинга и параметры конфигурации узла сети:




Вкладка **Подробности** содержит все доступные сведения об инвентаризации для узла сети:




Полнота данных инвентаризации зависит от того, насколько полно ведется информация об инвентаризации для узла сети. Если информация не ведется, вкладка *Подробности* отключена.

**5 Отчеты**

Обзор

В меню «Отчеты» есть несколько разделов, содержащих множество предопределенных и настраиваемых пользователем отчетов, ориентированных на отображение таких параметров, как системная информация, триггеры и собранные данные.

**1 Информация о системе**

Обзор

В *Отчеты → Сведения о системе* отображается сводка ключевых данных о сервере Zabbix и системе. Системные данные собираются с помощью внутренних элементов данных.

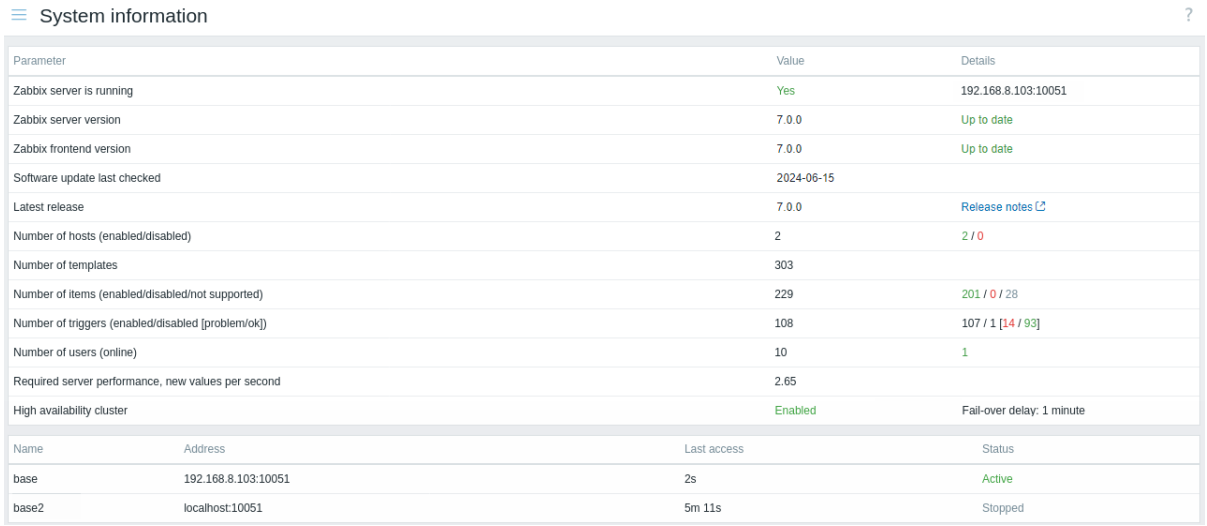
Обратите внимание, что в конфигурации высокой доступности можно перенаправить источник сведений о системе (экземпляр сервера). Для этого отредактируйте файл *zabbix.conf.php* — раскомментируйте и задайте `$ZBX_SERVER` или и `$ZBX_SERVER`, и `$ZBX_SERVER_PORT` на другой сервер, отличный от того, который отображается как активный.

Обратите внимание, что при задании только $ZBX_SERVER будет использоваться значение по умолчанию (10051) для $ZBX_SERVER_PORT.

При включенной конфигурации высокой доступности под системной статистикой отображается отдельный блок с подробными сведениями об узлах высокой доступности. Этот блок виден только пользователям Zabbix с ролью *Super Admin*.

*Сведения о системе* также доступны в виде виджета панели.

Системная статистика




Отображаемые данные:

| Parameter | Value | Details |
|---|---|---|
| *Zabbix server is running* | Статус сервера Zabbix:<br>**Yes** - сервер работает<br>**No** - сервер не работает<br>*Примечание:* Чтобы отобразить остальную информацию, веб-интерфейс требует, чтобы сервер работал, и на сервере должен быть запущен как минимум один процесс trapper (параметр StartTrappers в файле zabbix_server.conf > 0). | Расположение и порт сервера Zabbix. |
| *Zabbix server version* | Отображается номер текущей версии сервера.<br>*Примечание:* Отображается только когда сервер Zabbix работает. | Отображается статус версии сервера:<br>**Up to date** - используется последняя версия;<br>**New update available** - доступна более новая версия;<br>**Outdated** - срок полной поддержки для этой версии истек.<br>Эта информация доступна только если проверка обновлений ПО включена в конфигурации сервера Zabbix. Ничего не отображается, если последняя проверка обновлений ПО была выполнена более недели назад или данные о текущей версии отсутствуют. |

| Parameter | Value | Details |
|---|---|---|
| *Zabbix frontend version* | Отображается номер версии веб-интерфейса Zabbix. | Отображается статус версии веб-интерфейса Zabbix:<br>**Up to date** - используется последняя версия;<br>**New update available** - доступна более новая версия;<br>**Outdated** - срок полной поддержки для этой версии истек.<br>Эта информация доступна только если проверка обновлений ПО включена в конфигурации сервера Zabbix. Ничего не отображается, если последняя проверка обновлений ПО была выполнена более недели назад или данные о текущей версии отсутствуют. |
| *Software update last checked* | Отображается дата последней проверки обновлений ПО Zabbix.<br>Эта информация доступна только если проверка обновлений ПО включена в конфигурации сервера Zabbix. | |
| *Latest release* | Отображается номер более новой версии (если доступна) для текущей версии Zabbix.<br>Эта информация доступна только если проверка обновлений ПО включена в конфигурации сервера Zabbix. Ничего не отображается, если последняя проверка обновлений ПО была выполнена более недели назад или данные о текущей версии отсутствуют. | Отображается ссылка на примечания к выпуску последней доступной версии Zabbix. |
| *Number of hosts* | Отображается общее количество настроенных узлов сети. | Количество отслеживаемых/не отслеживаемых узлов сети. |
| *Number of templates* | Отображается общее количество шаблонов. | |
| *Number of items* | Отображается общее количество элементов данных. | Количество отслеживаемых/отключенных/неподдерживаемых элементов данных на уровне узла сети.<br>Элементы данных на отключенных узлах сети учитываются как отключенные. |
| *Number of triggers* | Отображается общее количество триггеров. | Количество включенных/отключенных триггеров на уровне узла сети; разделение включенных триггеров по состояниям ”Problem”/”OK”.<br><br>Триггеры, перечисленные в состоянии ”OK”, включают триггеры со статусом ”Unknown”.<br>Триггеры, зависящие от отключенных элементов данных или назначенные отключенным узлам сети, учитываются как отключенные. |
| *Number of users* | Отображается общее количество настроенных пользователей. | Количество пользователей онлайн. |

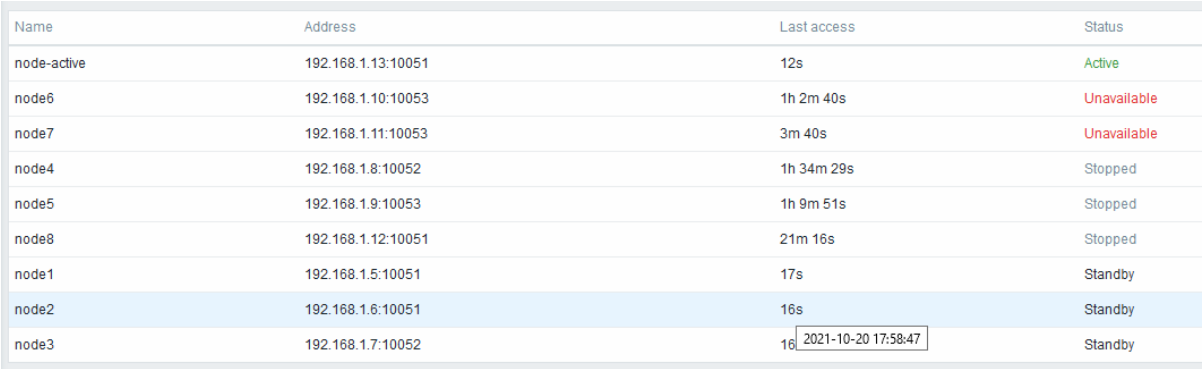
| Parameter | Value | Details |
|---|---|---|
| *Required server performance, new values per second* | Отображается ожидаемое количество новых значений, обрабатываемых сервером Zabbix в секунду. | *Required server performance* - это оценка, которая может быть полезна как ориентир. Для точного количества обработанных значений используйте внутренний элемент данных `zabbix[wcache,values,all]` internal item.<br><br>В расчет включаются включенные элементы данных с отслеживаемых узлов сети. Элементы данных журнала учитываются как одно значение за каждый интервал обновления элемента данных. Учитываются значения с обычным интервалом; значения с гибким интервалом и интервалом по расписанию не учитываются. Расчет не корректируется в течение периода обслуживания "nodata". Элементы данных trapper не учитываются. |
| *Global scripts on Zabbix server* | В этом поле будет отображаться **Disabled**, если глобальные скрипты отключены на сервере Zabbix путем установки EnableGlobalScripts=0 в конфигурации сервера. | |
| *High availability cluster* | Статус кластера высокой доступности для сервера Zabbix:<br>**Disabled** - автономный сервер<br>**Enabled** - существует как минимум один узел высокой доступности | Если включено, отображается задержка переключения при отказе. |

*Системная информация* также будет отображать сообщение об ошибке в следующих случаях:

- Используемая база данных не имеет требуемого набора символов или сопоставления (UTF-8).
- Версия базы данных ниже или выше поддерживаемого диапазона (доступно только пользователям с типом роли *Super admin role*).
- Housekeeping для TimescaleDB настроен неверно (таблицы history или trend содержат сжатые chunks, но параметры *Override item history period* или *Override item trend period* отключены).

Узлы высокой доступности

Если кластер высокой доступности включен, отображается еще один блок данных со статусом каждого узла высокой доступности.



| 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 |

Отображаемые данные:

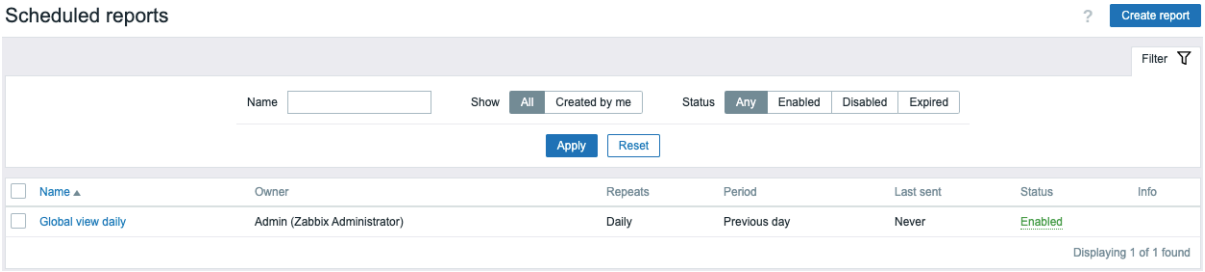
| Column | Description |
|---|---|
| *Name* | Имя узла, как определено в конфигурации сервера. |
| *Address* | IP-адрес и порт узла. |
| *Last access* | Время последнего доступа к узлу.<br>При наведении курсора на ячейку отображается метка времени последнего доступа в длинном формате. |

| Column | Description |
|---|---|
| *Status* | Статус узла. Строки таблицы сортируются по этим статусам в следующем порядке приоритета:<br>**Active** - узел работает и доступен<br>**Unavailable** - узел не был обнаружен дольше, чем задержка переключения при отказе (возможно, стоит выяснить причину)<br>**Stopped** - узел был остановлен или не смог запуститься (возможно, стоит запустить его или удалить)<br>**Standby** - узел работает и ожидает |

### 2 Запланированные отчеты

Обзор

В *Reports* → *Scheduled reports* пользователи с достаточными правами могут настроить плановое создание PDF-версий дашбордов, которые будут отправляться по электронной почте указанным получателям.




На стартовом экране отображается информация о запланированных отчетах, которую можно отфильтровать для удобной навигации — см. раздел Использование фильтра ниже.

Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя отчета. Щелчок по нему открывает форму настройки отчета. |
| *Owner* | Пользователь, создавший отчет. |
| *Repeats* | Частота формирования отчета (ежедневно/еженедельно/ежемесячно/ежегодно). |
| *Period* | Период времени, за который готовится отчет (предыдущий день, неделя, месяц или год). |
| *Last sent* | Дата и время, когда был отправлен последний отчет. |
| *Status* | Текущий статус отчета (включен/выключен/истек). Пользователи с достаточными правами могут изменить статус, щелкнув по нему: с ”Enabled” на ”Disabled” (и обратно); с ”Expired” на ”Disabled” (и обратно). Для пользователей с недостаточными правами статус не является кликабельным. |
| *Info* | Отображает информационные значки:<br>Красный значок указывает, что формирование отчета завершилось неудачно; при наведении на него отображается всплывающая подсказка с информацией об ошибке.<br>Желтый значок указывает, что отчет был сформирован, но отправка некоторым (или всем) получателям завершилась неудачно, либо что срок действия отчета истек; при наведении на него отображается всплывающая подсказка с дополнительной информацией. |

Использование фильтра

Вы можете использовать фильтр, чтобы сузить список отчетов. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Доступны следующие параметры фильтрации:

- *Имя* - допускается частичное совпадение имени
- *Показать* - созданные текущим пользователем или все отчеты
- *Статус* - выберите один из вариантов: ”Any” (показать все отчеты), ”Enabled”, ”Disabled” или ”Expired”

Фильтр расположен под названием раздела *Scheduled reports*. Его можно открыть и свернуть, нажав вкладку *Filter* в правом верхнем углу.

Массовое обновление

Иногда может потребоваться удалить или изменить статус сразу нескольких отчетов. Вместо того чтобы открывать каждый отчет по отдельности для редактирования, можно использовать для этого функцию массового обновления.

Чтобы массово обновить несколько отчетов, выполните следующие действия:

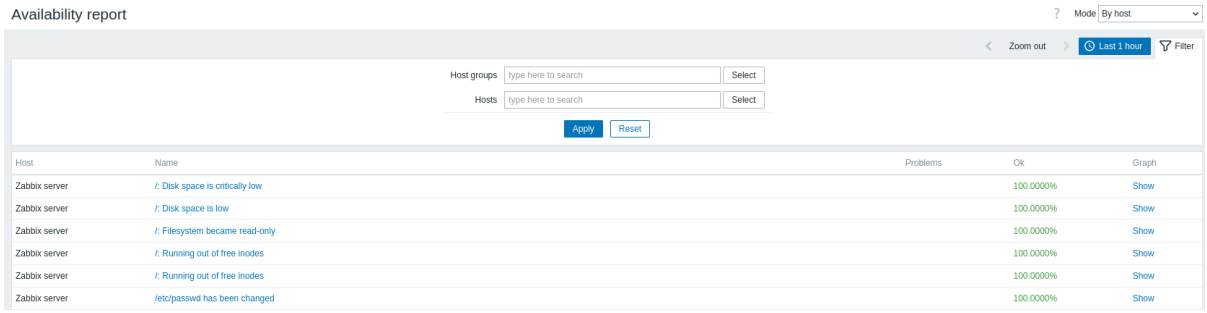
- Отметьте флажками отчеты, которые нужно обновить, в списке
- Нажмите нужную кнопку под списком, чтобы внести изменения (*Enable*, *Disable* или *Delete*)

## 3 Отчет о доступности

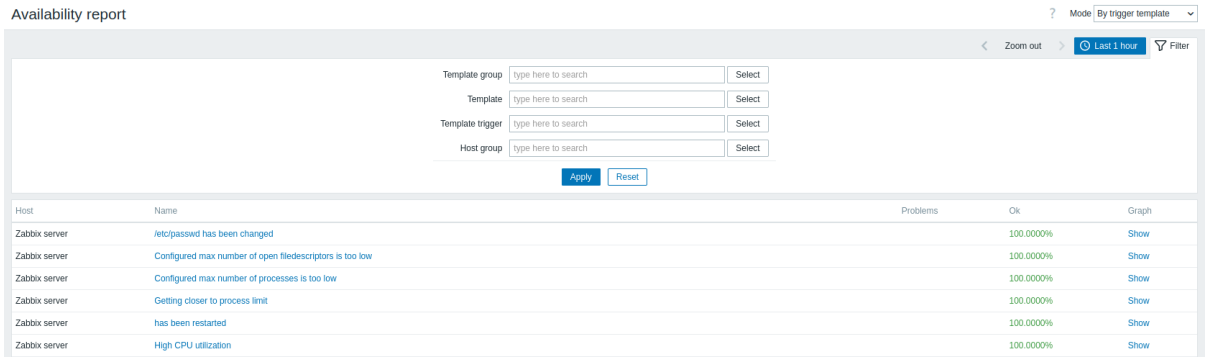
Обзор

В *Reports* > *Availability report* можно увидеть, какую долю времени каждый триггер находился в состоянии проблемы/OK.

Для каждого состояния отображается процент времени, что позволяет легко определить доступность различных элементов в вашей системе.




В раскрывающемся списке в правом верхнем углу можно выбрать режим отображения — показывать триггеры по узлам сети или триггеры, относящиеся к шаблону.




Имя триггера является ссылкой на последние события этого триггера.

Использование фильтра

Фильтр может помочь сузить количество отображаемых узлов сети и/или триггеров. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Фильтр расположен под названием раздела *Отчет о доступности*. Его можно открыть и свернуть, щелкнув вкладку *Фильтр* справа.

Фильтрация по шаблону триггера

В режиме *По шаблону триггера* результаты можно фильтровать по одному или нескольким параметрам, перечисленным ниже.

| Parameter | Description |
|---|---|
| *Template group* | Фильтровать узлы сети по триггерам, унаследованным от шаблонов, принадлежащих выбранной группе шаблонов.<br>При указании родительской группы шаблонов автоматически выбираются все вложенные группы шаблонов. |
| *Template* | Фильтровать узлы сети по триггерам, унаследованным от выбранного шаблона, включая вложенные шаблоны.<br>Если вложенный шаблон имеет собственные триггеры, эти триггеры отображаться не будут. |

| Parameter | Description |
|---|---|
| *Template trigger* | Фильтровать узлы сети по выбранному триггеру. Другие триггеры отфильтрованных узлов сети отображаться не будут. |
| *Host group* | Фильтровать узлы сети, принадлежащие выбранной группе узлов сети. |

Фильтрация по узлу сети

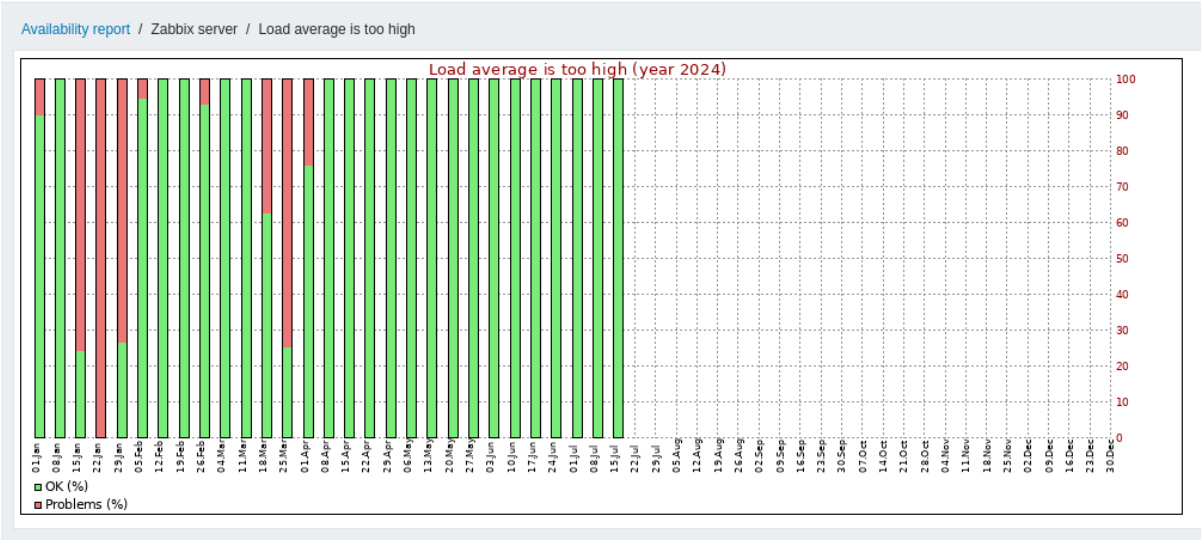
В режиме *By host* результаты можно фильтровать по узлу сети или группе узлов сети.
При указании родительской группы узлов сети автоматически выбираются все вложенные группы узлов сети.

Выбор периода времени

Выбор периода времени позволяет выбрать часто используемые периоды времени одним щелчком. Селектор можно развернуть и свернуть, щелкнув вкладку *Период времени* рядом с фильтром.

Щелчок по *Показать* в столбце *График* отображает информацию о доступности в виде столбчатой диаграммы, где каждый столбец представляет прошедшую неделю текущего года.

Availability report graph




Зеленый цвет столбца обозначает время OK, а красный - время проблемы.

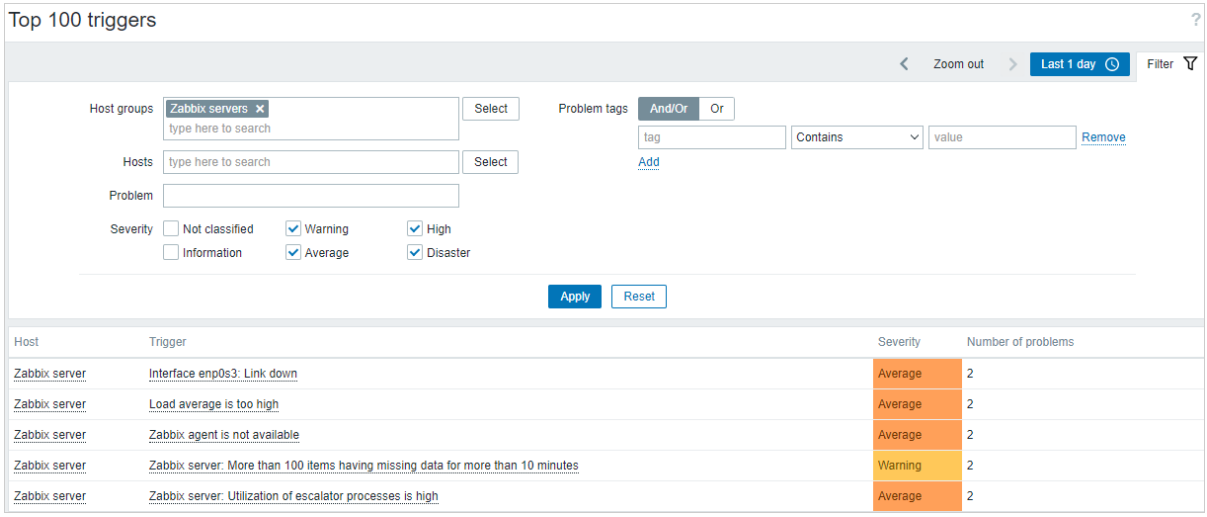
Влияние периодов обслуживания

Обслуживание не исключает время из отчета о доступности автоматически. Только обслуживание, настроенное как *No data collection*, останавливает опрос (поэтому проблемы не создаются) и, следовательно, не изменяет процент доступности для затронутых триггеров.

**4 Топ 100 триггеров**

Обзор

В *Reports* → *Top 100 triggers* можно увидеть триггеры с наибольшим количеством проблем, обнаруженных за выбранный период.




Элементы в столбцах узла сети и триггера являются ссылками, которые предоставляют несколько полезных возможностей:

- для узла сети - щелчок по имени узла сети открывает меню узла сети
- для триггера - щелчок по имени триггера открывает ссылки на последние события, простой график для каждого элемента данных триггера, а также формы настройки самого триггера и каждого элемента данных триггера

**Использование фильтра**

Вы можете использовать фильтр для отображения триггеров по группе узлов сети, узлу сети, имени проблемы, тегам или уровню серьезности триггера. Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети. Для лучшей производительности поиска данные ищутся без раскрытия макросов.

Фильтр расположен под названием раздела *Top 100 triggers*. Его можно развернуть и свернуть, щелкнув вкладку *Filter* справа.

**Выбор периода времени**

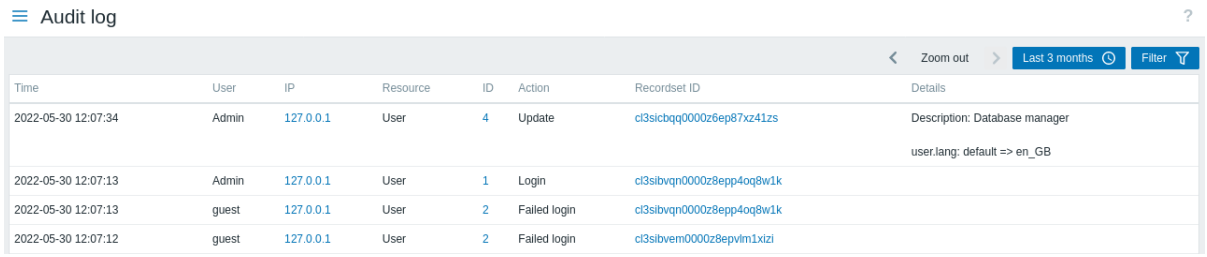
*Time period* selector позволяет выбрать часто используемые периоды одним щелчком мыши. *Time period* selector можно развернуть и свернуть, щелкнув вкладку *Time period* рядом с фильтром.

### 5 Журнал аудита

Обзор

В разделе *Reports* → *Audit log* можно просматривать записи о действиях пользователей и системы.

**Note:**
Чтобы записи аудита собирались и отображались, в разделе *Administration* → *Audit log* должен быть отмечен флажок *Enable audit logging*. Если этот параметр не включен, история действий не будет записываться в базу данных и не будет отображаться в журнале аудита.




Журнал аудита отображает следующие данные:

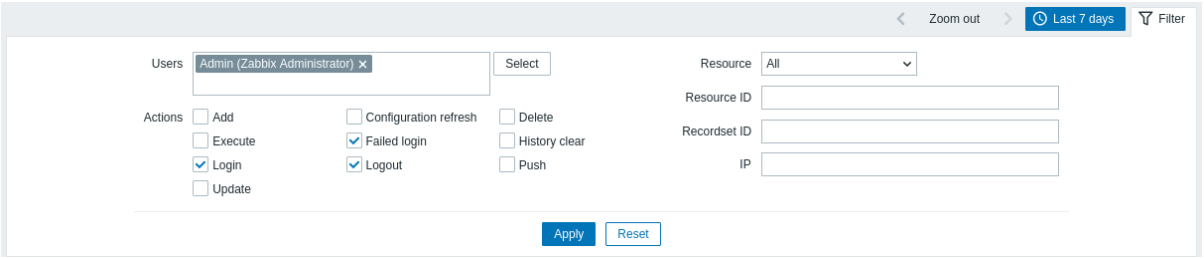
| Column | Description |
|---|---|
| *Time* | Метка времени записи аудита. |
| *User* | Пользователь, выполнивший действие. |
| *IP* | IP-адрес, с которого было инициировано действие.<br>При нажатии на гиперссылку записи журнала аудита будут отфильтрованы по этому IP-адресу. |
| *Resource* | Тип затронутого ресурса (*API token*, *Action*, *Authentication*, *Autoregistration* и т. д.). |

| Column | Description |
|---|---|
| *ID* | ID затронутого ресурса.<br>При нажатии на гиперссылку записи журнала аудита будут отфильтрованы по этому ID ресурса. |
| *Action* | Тип действия (*Add*, *Configuration refresh*, *Delete*, *Execute*, *Failed login*, *History clear*, *Login*, *Logout*, *Push*, *Update*). |
| *Recordset ID* | Общий ID для всех записей журнала аудита, созданных в результате одной и той же операции.<br>Например, при привязке шаблона к узлу сети для каждой унаследованной сущности шаблона (элемент данных, триггер и т. д.) создается отдельная запись журнала аудита — все эти записи будут иметь одинаковый *Recordset ID*.<br>При нажатии на гиперссылку записи журнала аудита будут отфильтрованы по этому *Recordset ID*. |
| *Details* | Описание ресурса и подробная информация о выполненном действии.<br>Если запись содержит более двух строк, дополнительно отображается ссылка *Details*. Нажмите эту ссылку, чтобы просмотреть полный список изменений. |

**Note:**
Когда элемент данных типа trapper или элемент данных HTTP agent (с включенным trapping) получает данные, запись в журнале аудита будет добавлена только в том случае, если данные были отправлены с помощью API-метода `history.push`, а не утилиты Zabbix sender.

Использование фильтра

Фильтр расположен под панелью *Журнал аудита*. Его можно развернуть и свернуть, щелкнув вкладку *Фильтр* в правом верхнем углу.




С помощью фильтра можно сузить список записей по пользователю, затронутому ресурсу, ID ресурса, выполненному действию (*ID набора записей*) и IP-адресу. В зависимости от ресурса в фильтре можно выбрать одно или несколько конкретных действий.

Для повышения производительности поиска все данные ищутся с неразрешенными макросами.

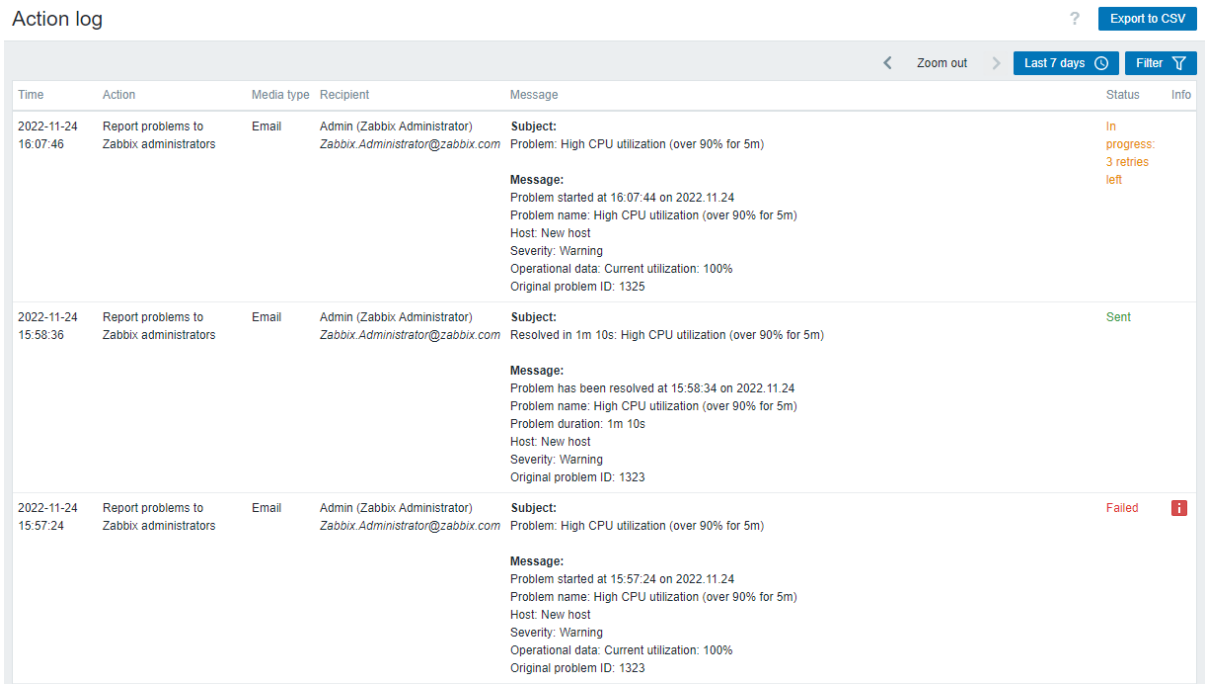
Выбор периода времени

*Выбор периода времени* позволяет выбрать часто используемые периоды одним щелчком мыши. Выбор *периода времени* можно развернуть и свернуть, щелкнув вкладку *Период времени* в правом верхнем углу.

### 6 Журнал действий

Обзор

В разделе *Reports* → *Action log* пользователи могут просматривать подробности операций (уведомления, удаленные команды), выполненных в рамках действия.




Отображаемые данные:

| Column | Description |
|---|---|
| *Time* | Метка времени операции. |
| *Action* | Имя действия, вызвавшего операции. |
| *Media type* | Тип медиа (например, Email, Jira и т. д.), используемый для отправки уведомления.<br>Для операций, которые выполняли удаленные команды, этот столбец будет пустым. |
| *Recipient* | Информация о получателе уведомления - имя пользователя, имя и фамилия (в скобках), а также дополнительная информация в зависимости от типа медиа (email, имя пользователя и т. д.).<br>Для операций, которые выполняли удаленные команды, этот столбец будет пустым. |
| *Message* | Содержимое сообщения, удаленной команды или имени глобального скрипта.<br>Удаленная команда отделяется от целевого узла сети символом двоеточия: `<host>:<command>`. Например, если удаленная команда была выполнена на Zabbix server, то информация будет иметь следующий формат: `Zabbix server:<command>`. |
| *Status* | Статус операции:<br>*In progress* - операция отправки уведомления выполняется (также отображается оставшееся количество попыток, которые сервер предпримет для отправки уведомления)<br>*Sent* - уведомление отправлено<br>*Executed* - удаленная команда выполнена<br>*Failed* - операция не была завершена |
| *Info* | Информация об ошибке (если есть), связанная с выполнением операции. |

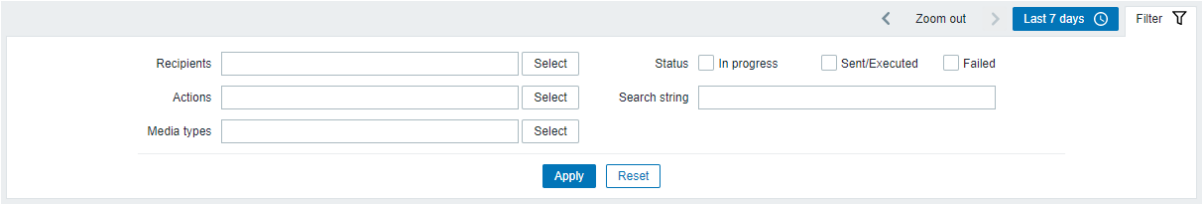
Кнопки

Кнопка в правом верхнем углу страницы предлагает следующую возможность:

| | |
|---|---|
| Export to CSV | Экспортировать записи журнала действий с текущей страницы в CSV-файл. Если применен фильтр, будут экспортированы только отфильтрованные записи.<br>В экспортированном CSV-файле столбцы ”Recipient” и ”Message” разделены на несколько столбцов - ”Recipient’s Zabbix username”, ”Recipient’s name”, ”Recipient’s surname”, ”Recipient”, а также ”Subject”, ”Message”, ”Command”. |

Использование фильтра

Фильтр расположен под панелью *Журнал действий*. Его можно развернуть и свернуть, щелкнув вкладку *Фильтр* в правом верхнем углу страницы.




Вы можете использовать фильтр, чтобы сузить список записей по получателям уведомлений, действиям, типам носителей, статусу или по содержимому сообщения/удаленной команды (*Строка поиска*). Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Выбор периода времени

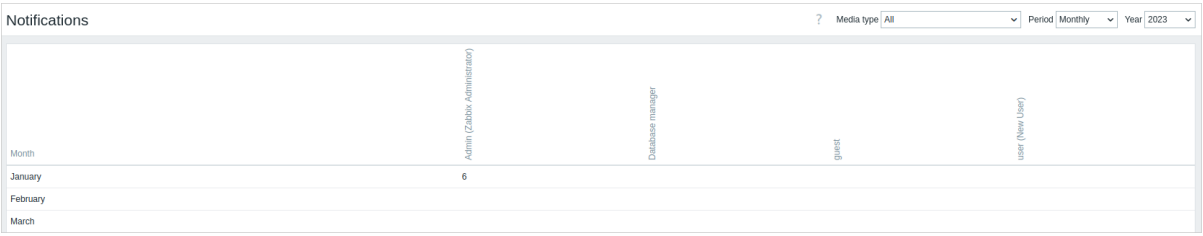
*Выбор периода времени* позволяет выбирать часто используемые периоды одним щелчком мыши. *Выбор периода времени* можно развернуть и свернуть, щелкнув вкладку *Период времени* рядом с фильтром.

**7 Уведомления**

Обзор

В разделе *Reports* → *Notifications* отображается отчет о количестве уведомлений, отправленных каждому пользователю.

В раскрывающихся списках в правом верхнем углу можно выбрать тип медиа (или все), период (данные за каждый день/неделю/месяц/год) и год для отправленных уведомлений.




Каждый столбец отображает итоговые значения для одного системного пользователя.

**6 Сбор данных**

Обзор

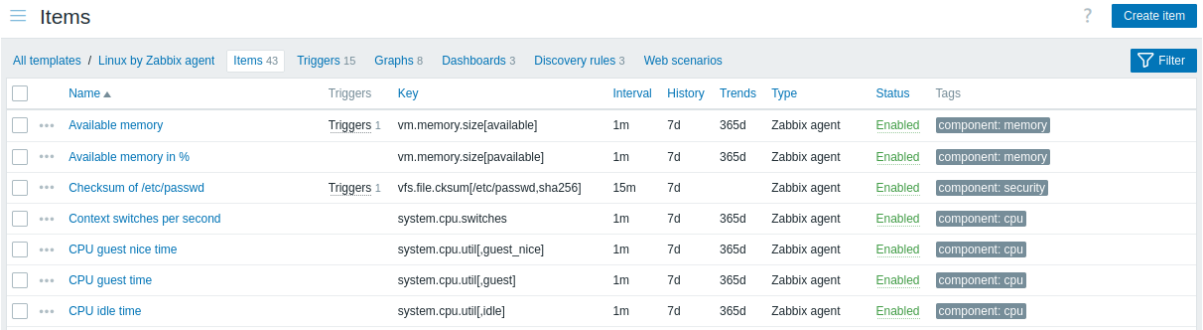
Это меню содержит разделы, связанные с настройкой сбора данных.

1 Элементы данных

Обзор

К списку элементов данных для шаблона можно перейти из *Сбор данных* → *Шаблоны*, нажав *Элементы данных* для соответствующего шаблона.

Отображается список существующих элементов данных.




Отображаемые данные:

| Column | Description |
|---|---|
| *Item menu* | Нажмите значок с тремя точками, чтобы открыть меню элемента данных. |
| *Template* | Шаблон, к которому относится элемент данных.<br>Нажатие на имя шаблона открывает форму настройки шаблона.<br>Этот столбец отображается только если в фильтре выбрано несколько шаблонов или не выбрано ни одного шаблона. |
| *Name* | Имя элемента данных, отображаемое как синяя ссылка на сведения об элементе данных.<br>Нажатие на ссылку с именем элемента данных открывает форму настройки элемента данных.<br>Если элемент данных унаследован от другого шаблона, имя шаблона отображается перед именем элемента данных в виде серой ссылки. Нажатие на ссылку шаблона открывает список элементов данных на уровне этого шаблона. |
| *Triggers* | При наведении указателя мыши на Triggers отображается информационный блок с триггерами, связанными с элементом данных.<br>Количество триггеров отображается серым цветом. |
| *Key* | Отображается ключ элемента данных. |
| *Interval* | Отображается частота проверки. |
| *History* | Отображается, сколько дней будет храниться история данных элемента данных. |
| *Trends* | Отображается, сколько дней будет храниться история трендов элемента данных. |
| *Type* | Отображается тип элемента данных (агент Zabbix, SNMP agent, simple check и т. д.). |
| *Status* | Отображается статус элемента данных - *Enabled* или *Disabled*. Нажав на статус, можно изменить его - с Enabled на Disabled (и обратно). |
| *Tags* | Отображаются теги элемента данных.<br>Можно отобразить до трех тегов (пар имя:значение). Если тегов больше, отображается ссылка "..." , позволяющая увидеть все теги при наведении указателя мыши. |

Чтобы настроить новый элемент данных, нажмите кнопку *Create item* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Enable* - изменить статус элемента данных на *Enabled*.
- *Disable* - изменить статус элемента данных на *Disabled*.
- *Copy* - скопировать элементы данных на другие узлы сети или шаблоны.
- *Mass update* - обновить несколько свойств для нескольких элементов данных одновременно.
- *Delete* - удалить элементы данных.

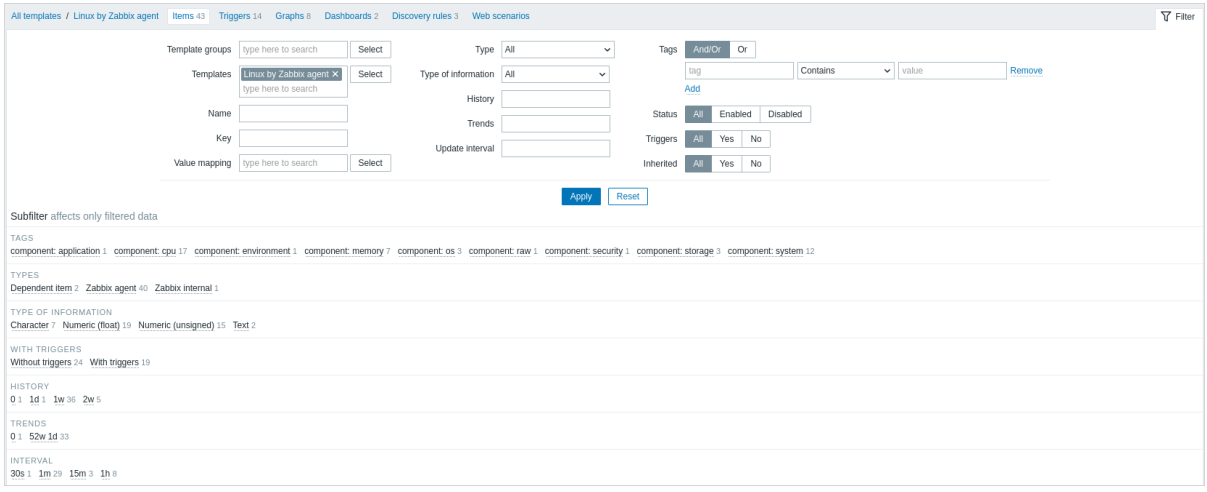
Чтобы использовать эти параметры, отметьте флажки напротив нужных элементов данных, затем нажмите требуемую кнопку.

Использование фильтра

Список элементов данных может содержать большое количество элементов. Используя фильтр, вы можете отфильтровать часть из них, чтобы быстро найти нужные элементы.

Для повышения производительности поиска данные ищутся без подстановки макросов.

Значок *Фильтр* доступен в правом верхнем углу. При нажатии на него откроется фильтр, в котором можно указать нужные критерии фильтрации.

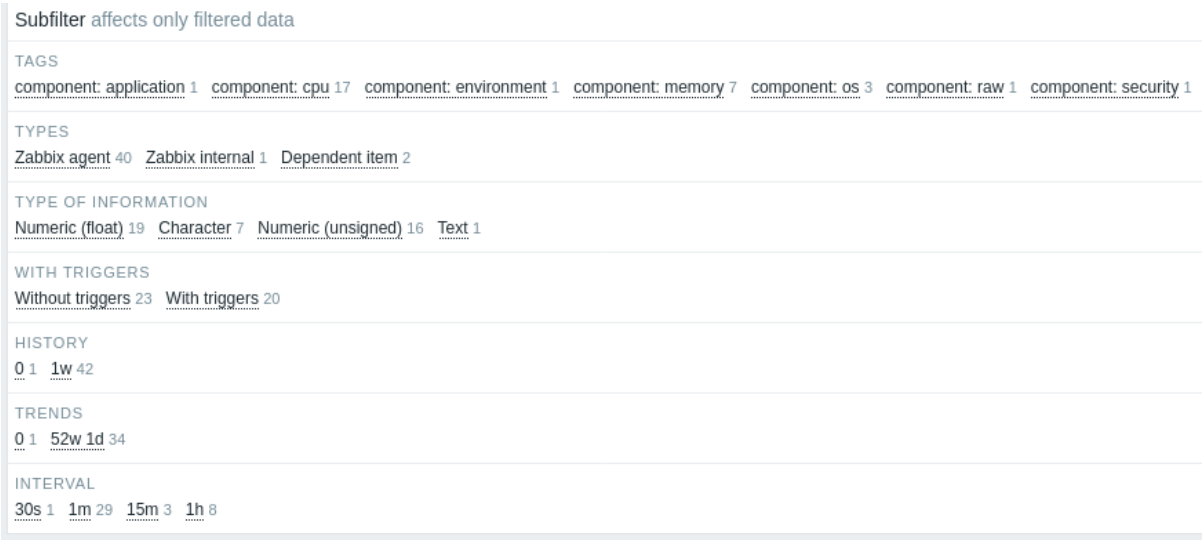


| Parameter | Description |
|---|---|
| *Template groups* | Фильтрация по одной или нескольким группам шаблонов.<br>При указании родительской группы шаблонов автоматически выбираются все вложенные группы. |
| *Templates* | Фильтрация по одному или нескольким шаблонам. |
| *Name* | Фильтрация по имени элемента данных. |
| *Key* | Фильтрация по ключу элемента данных. |
| *Value mapping* | Фильтрация по используемой карте значений.<br>Этот параметр не отображается, если параметр *Templates* пуст. |
| *Type* | Фильтрация по типу элемента данных (агент Zabbix, SNMP agent и т. д.). |
| *Type of information* | Фильтрация по типу информации (беззнаковое число, float и т. д.). |
| *History* | Фильтрация по времени хранения истории элемента данных. |
| *Trends* | Фильтрация по времени хранения трендов элемента данных. |
| *Update interval* | Фильтрация по интервалу обновления элемента данных. |
| *Tags* | Укажите теги, чтобы ограничить количество отображаемых элементов данных. Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, значения тегов которых содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, значения тегов которых содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия |
| *Status* | Фильтрация по статусу элемента данных - *Enabled* или *Disabled*. |
| *Triggers* | Фильтрация элементов данных с триггерами или без них. |
| *Inherited* | Фильтрация унаследованных или неунаследованных элементов данных из связанных шаблонов. |

Использование подфильтра

Подфильтр позволяет дополнительно изменять фильтрацию, заданную основным фильтром.

Он содержит кликабельные ссылки для быстрого доступа к связанным элементам. Элементы связаны по общему признаку - тегу, типу элемента данных, интервалу обновления элемента данных и т. д. При нажатии на признак он выделяется серым фоном, а элементы данных немедленно фильтруются (нет необходимости нажимать *Применить* в основном фильтре). При нажатии на другой признак он добавляется к отфильтрованным результатам. При повторном нажатии на признак фильтрация снимается.




Подфильтры формируются на основе отфильтрованных данных, количество которых ограничено 1000 записями. Если вы

хотите видеть больше записей в подфильтре, необходимо увеличить значение параметра *Limit for search and filter results* в разделе *Administration -> General -> GUI*.

В отличие от основного фильтра, подфильтр обновляется при каждом запросе на обновление таблицы, чтобы всегда содержать актуальную информацию о доступных вариантах фильтрации и их счетчиках.

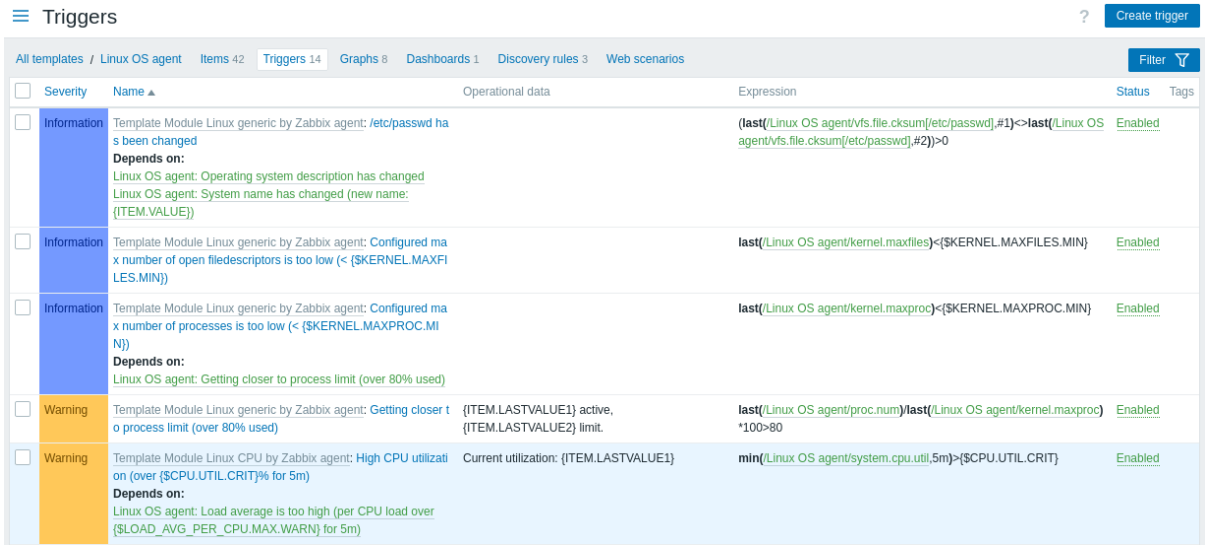
Количество отображаемых признаков ограничено 100 по горизонтали. Если их больше, в конце отображается значок с тремя точками; он не является кликабельным.

Число рядом с каждым кликабельным признаком показывает количество элементов данных, сгруппированных в нем (на основе результатов основного фильтра). Когда признак выбран, числа у других доступных признаков отображаются со знаком плюс, показывая, сколько элементов данных может быть добавлено к текущему выбору. Признаки без элементов данных не отображаются, если только они не были выбраны в подфильтре ранее.

2 Триггеры

Обзор

Список триггеров для шаблона можно открыть в разделе *Сбор данных → Шаблоны*, нажав *Триггеры* для соответствующего шаблона.




Отображаемые данные:

| Column | Description |
|---|---|
| *Severity* | Серьезность триггера отображается как названием, так и цветом фона ячейки. |
| *Template* | Шаблон, к которому принадлежит триггер.<br>Нажатие на имя шаблона открывает форму настройки шаблона.<br>Этот столбец отображается только если в фильтре выбрано несколько шаблонов или не выбрано ни одного. |
| *Name* | Имя триггера, отображаемое как синяя ссылка на сведения о триггере.<br>Нажатие на ссылку с именем триггера открывает форму настройки триггера.<br>Если триггер унаследован от другого шаблона, имя шаблона отображается перед именем триггера в виде серой ссылки. Нажатие на ссылку шаблона открывает список триггеров на уровне этого шаблона. |
| *Operational data* | Определение оперативных данных триггера, содержащее произвольные строки и макросы, которые будут динамически подставляться в *Мониторинг → Проблемы*. |
| *Expression* | Отображается выражение триггера. Часть выражения, относящаяся к шаблону-элементу данных, отображается как ссылка, ведущая к форме настройки элемента данных. |
| *Status* | Отображается статус триггера - *Enabled* или *Disabled*. Нажав на статус, вы можете изменить его - с Enabled на Disabled (и обратно). |
| *Tags* | Если триггер содержит теги, в этом столбце отображаются имя и значение тега. |

Чтобы настроить новый триггер, нажмите кнопку *Create trigger* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

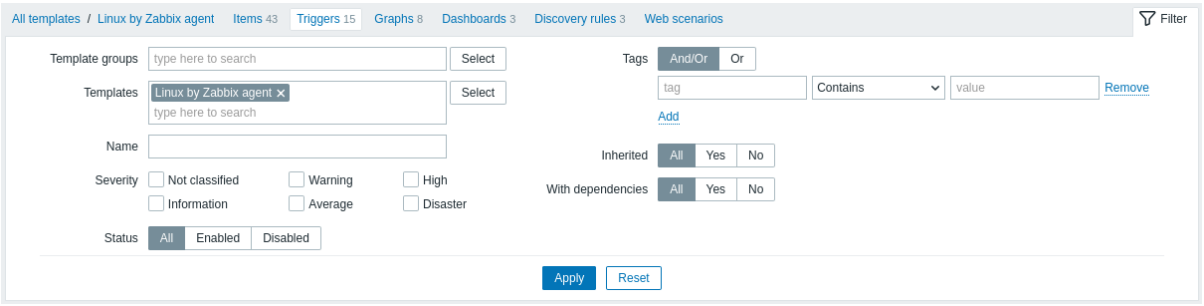
- *Включить* - изменить статус триггера на *Включен*
- *Отключить* - изменить статус триггера на *Отключен*
- *Копировать* - скопировать триггеры на другие узлы сети или шаблоны
- *Массовое обновление* - обновить несколько свойств для ряда триггеров одновременно
- *Удалить* - удалить триггеры

Чтобы использовать эти параметры, отметьте флажки перед нужными триггерами, затем нажмите требуемую кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те триггеры, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Значок *Фильтр* доступен в правом верхнем углу. При нажатии на него откроется фильтр, в котором можно указать нужные критерии фильтрации.



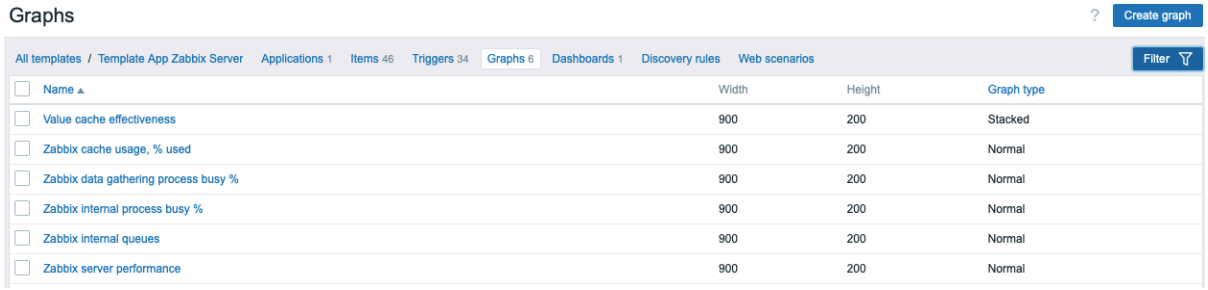

| Parameter | Description |
|---|---|
| *Template groups* | Фильтрация по одной или нескольким группам шаблонов.<br>Указание родительской группы шаблонов неявно выбирает все вложенные группы. |
| *Templates* | Фильтрация по одному или нескольким шаблонам.<br>Если выше уже выбраны группы шаблонов, выбор шаблонов ограничивается этими группами. |
| *Name* | Фильтрация по имени триггера. |
| *Severity* | Выберите, чтобы фильтровать по одной или нескольким степеням важности триггера. |
| *Status* | Фильтрация по статусу триггера. |
| *Tags* | Фильтрация по имени и значению тега триггера. Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия<br>В полях имени тега и значения тега поддерживаются макросы и функции макросов. |
| *Inherited* | Фильтрация триггеров, унаследованных (или не унаследованных) от связанных шаблонов. |
| *With dependencies* | Фильтрация триггеров с зависимостями (или без зависимостей). |

3 Графики

Обзор

Список пользовательских графиков для шаблона можно открыть в разделе *Сбор данных* → *Шаблоны*, нажав *Графики* для соответствующего шаблона.

Отображается список существующих графиков.




Отображаемые данные:

| Column | Description |
|---|---|
| *Template* | Шаблон, которому принадлежит график.<br>Нажатие на имя шаблона открывает форму настройки шаблона.<br>Этот столбец отображается только если в фильтре выбрано несколько шаблонов или не выбрано ни одного шаблона. |
| *Name* | Имя пользовательского графика, отображаемое как синяя ссылка на сведения о графике.<br>Нажатие на ссылку с именем графика открывает форму настройки графика.<br>Если график унаследован от другого шаблона, перед именем графика отображается имя шаблона в виде серой ссылки. Нажатие на ссылку шаблона открывает список графиков на уровне этого шаблона. |
| *Width* | Отображается ширина графика. |
| *Height* | Отображается высота графика. |
| *Graph type* | Отображается тип графика — *Normal*, *Stacked*, *Pie* или *Exploded*. |

Чтобы настроить новый график, нажмите кнопку *Create graph* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Копировать* - копировать графики на другие узлы сети или шаблоны
- *Удалить* - удалить графики

Чтобы использовать эти параметры, отметьте флажки перед соответствующими графиками, затем нажмите нужную кнопку.

Использование фильтра

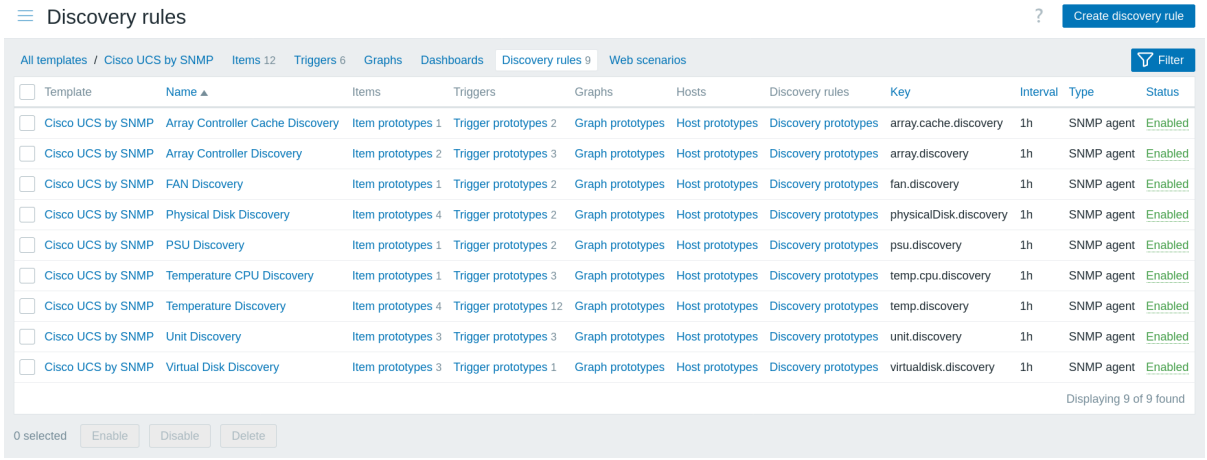
Вы можете фильтровать графики по группе шаблонов и шаблону. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

4 Правила обнаружения

Обзор

Список правил низкоуровневого обнаружения для шаблона можно открыть в разделе *Сбор данных* → *Шаблоны*, нажав *Обнаружение* для соответствующего шаблона.

Отображается список существующих правил низкоуровневого обнаружения. Также можно просматривать все правила обнаружения независимо от шаблона или все правила обнаружения определенной группы шаблонов, изменив параметры фильтра.




Отображаемые данные:

| Column | Description |
|---|---|
| *Template* | Шаблон, которому принадлежит правило обнаружения.<br>При нажатии на имя шаблона открывается форма настройки шаблона. |
| *Name* | Имя правила, отображаемое в виде синей ссылки.<br>При нажатии на имя правила открывается форма настройки правила низкоуровневого обнаружения.<br>Если правило обнаружения унаследовано от другого шаблона, имя шаблона отображается перед именем правила в виде серой ссылки. При нажатии на ссылку шаблона открывается список правил обнаружения на уровне этого шаблона. |
| *Items* | Отображается ссылка на список прототипов элементов данных.<br>Количество существующих прототипов элементов данных отображается серым цветом. |
| *Triggers* | Отображается ссылка на список прототипов триггеров.<br>Количество существующих прототипов триггеров отображается серым цветом. |
| *Graphs* | Отображается ссылка на список прототипов графиков.<br>Количество существующих прототипов графиков отображается серым цветом. |
| *Hosts* | Отображается ссылка на список прототипов узлов сети.<br>Количество существующих прототипов узлов сети отображается серым цветом. |
| *Key* | Отображается ключ элемента данных, используемый для обнаружения. |
| *Interval* | Отображается частота выполнения обнаружения. |
| *Type* | Отображается тип элемента данных, используемый для обнаружения (агент Zabbix, SNMP agent и т. д.). |
| *Status* | Отображается статус правила обнаружения - *Enabled* или *Disabled*. Нажав на статус, вы можете изменить его - с Enabled на Disabled (и обратно). |

Чтобы настроить новое правило низкоуровневого обнаружения, нажмите кнопку *Create discovery rule* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

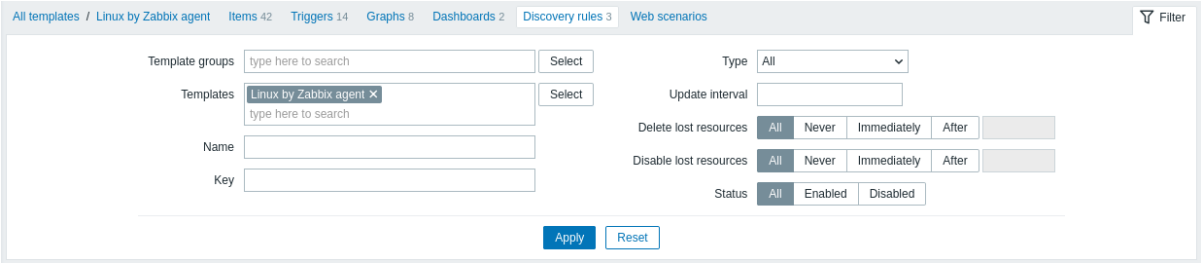
- *Включить* - изменить статус правила низкоуровневого обнаружения на *Включено*
- *Отключить* - изменить статус правила низкоуровневого обнаружения на *Отключено*
- *Удалить* - удалить правила низкоуровневого обнаружения

Чтобы использовать эти параметры, отметьте флажки перед соответствующими правилами обнаружения, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те правила обнаружения, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Значок *Filter* доступен в правом верхнем углу. При нажатии на него откроется фильтр, в котором можно указать нужные критерии фильтрации, такие как шаблон, имя правила обнаружения, ключ элемента данных, тип элемента данных и т. д.



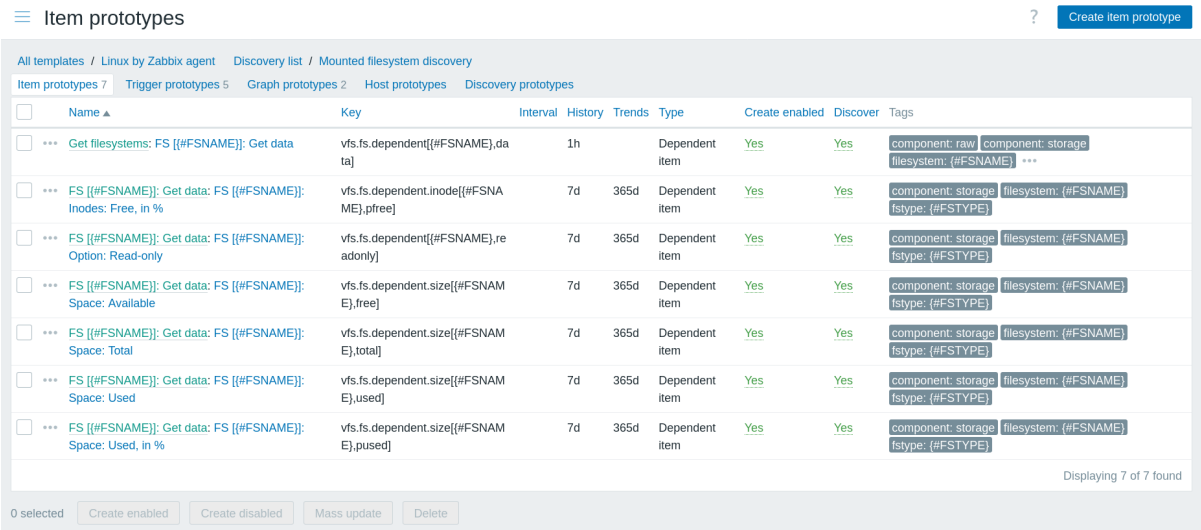

| Parameter | Description |
|---|---|
| *Template groups* | Фильтрация по одной или нескольким группам шаблонов.<br>При указании родительской группы шаблонов автоматически выбираются все вложенные группы. |
| *Templates* | Фильтрация по одному или нескольким шаблонам. |
| *Name* | Фильтрация по имени правила обнаружения. |
| *Key* | Фильтрация по ключу элемента данных обнаружения. |
| *Type* | Фильтрация по типу элемента данных обнаружения. |
| *Update interval* | Фильтрация по интервалу обновления.<br>Недоступно для Zabbix trapper и зависимых элементов данных. |
| *Delete lost resources* | Фильтрация по периоду *Delete lost resources*. |
| *Disable lost resources* | Фильтрация по периоду *Disable lost resources*. |
| *Status* | Фильтрация по статусу правила обнаружения (All/Enabled/Disabled). |

1 Прототипы элементов данных

Обзор

В этом разделе отображаются настроенные прототипы элементов данных правила обнаружения низкого уровня на шаблоне.

Если шаблон связан с узлом сети, прототипы элементов данных станут основой для создания реальных элементов данных узла сети items во время обнаружения низкого уровня.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа элемента данных, отображается как синяя ссылка.<br>При нажатии на имя открывается форма настройки прототипа элемента данных.<br>Если прототип элемента данных принадлежит связанному шаблону, имя шаблона отображается перед именем прототипа элемента данных серой ссылкой. При нажатии на ссылку шаблона открывается список прототипов элементов данных на уровне связанного шаблона.<br>Если прототип элемента данных имеет тип ”Зависимый элемент”, имя мастер-элемента отображается перед именем прототипа элемента данных зеленым цветом (как и в случае прототипов элементов данных на уровне узла сети). |
| *Key* | Отображается ключ прототипа элемента данных. |
| *Interval* | Отображается частота проверки. |
| *History* | Отображается, сколько дней хранить историю данных элемента. |
| *Trends* | Отображается, сколько дней хранить историю трендов элемента. |
| *Type* | Отображается тип прототипа элемента данных (агент Zabbix, SNMP agent, простая проверка и т. д.). |
| *Create enabled* | Создавать элемент на основе этого прототипа как:<br>**Yes** - включено<br>**No** - отключено. Можно переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Discover* | Обнаруживать элемент на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Можно переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Tags* | Отображаются теги прототипа элемента данных. |

Чтобы настроить новый прототип элемента данных, нажмите кнопку *Create item prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Создать включенными* - создать эти элементы данных как *Включенные*
- *Создать отключенными* - создать эти элементы данных как *Отключенные*
- *Массовое обновление* - массово обновить эти прототипы элементов данных
- *Удалить* - удалить эти прототипы элементов данных

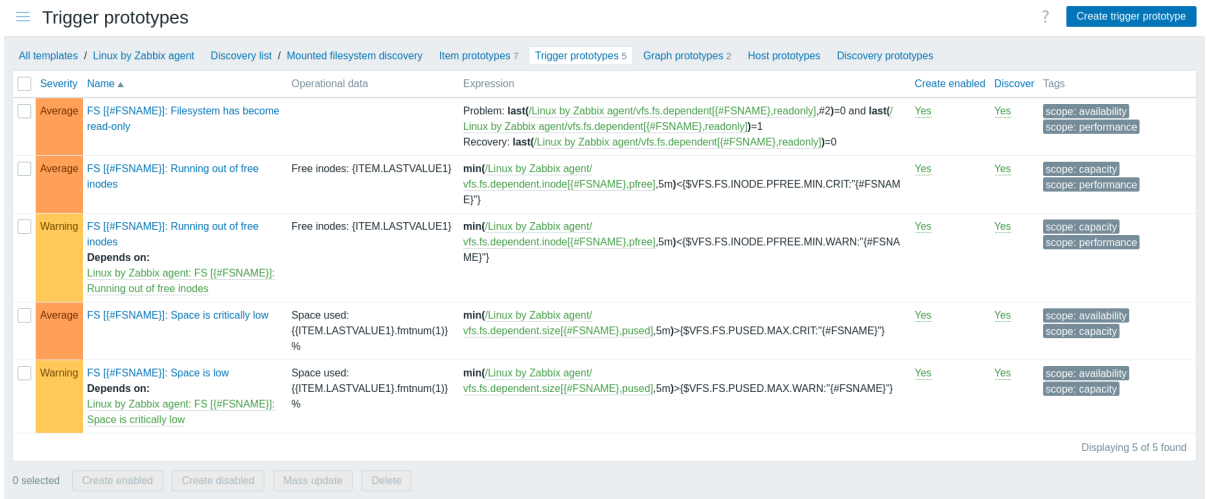
Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами элементов данных, затем нажмите нужную кнопку.

2 Прототипы триггеров

Обзор

В этом разделе отображаются настроенные прототипы триггеров правила низкоуровневого обнаружения на шаблоне.

Если шаблон привязан к узлу сети, прототипы триггеров станут основой для создания реальных триггеров узла сети triggers во время низкоуровневого обнаружения.



Отображаемые данные:

| Column | Description |
|---|---|
| *Severity* | Серьезность триггера отображается как названием, так и цветом фона ячейки. |
| *Name* | Имя прототипа триггера отображается как синяя ссылка.<br>Щелчок по имени открывает форму настройки прототипа триггера.<br>Если прототип триггера принадлежит связанному шаблону, имя шаблона отображается перед именем триггера в виде серой ссылки. Щелчок по ссылке шаблона открывает список прототипов триггеров на уровне связанного шаблона. |
| *Expression* | Отображается выражение триггера. Часть выражения, относящаяся к шаблону-элементу данных, отображается как ссылка, ведущая к форме настройки элемента данных. |
| *Operational data* | Отображается формат оперативных данных триггера, содержащий произвольные строки и макросы, которые будут динамически подставляться в *Monitoring → Problems*. |
| *Create enabled* | Создать триггер на основе этого прототипа как:<br>**Yes** - включен<br>**No** - отключен. Вы можете переключаться между 'Yes' и 'No', щелкая по ним. |
| *Discover* | Обнаруживать триггер на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Вы можете переключаться между 'Yes' и 'No', щелкая по ним. |
| *Tags* | Отображаются теги прототипа триггера. |

Чтобы настроить новый прототип триггера, нажмите кнопку *Create trigger prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Создать включенными* - создать эти триггеры как *Включенные*
- *Создать отключенными* - создать эти триггеры как *Отключенные*
- *Массовое обновление* - массово обновить эти прототипы триггеров
- *Удалить* - удалить эти прототипы триггеров

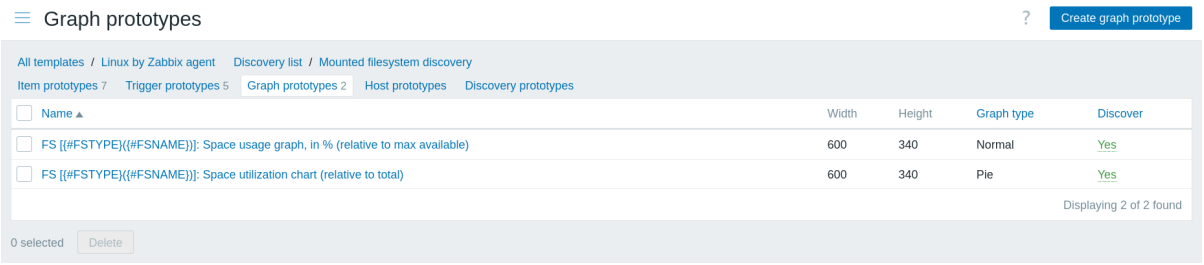
Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами триггеров, затем нажмите нужную кнопку.

3 Прототипы графиков

Обзор

В этом разделе отображаются настроенные прототипы графиков правила обнаружения низкого уровня на шаблоне.

Если шаблон связан с узлом сети, прототипы графиков станут основой для создания реальных графиков узла сети graphs во время обнаружения низкого уровня.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа графика, отображается как синяя ссылка.<br>Щелчок по имени открывает форму настройки прототипа графика.<br>Если прототип графика принадлежит связанному шаблону, имя шаблона отображается перед именем графика как серая ссылка. Щелчок по ссылке шаблона открывает список прототипов графиков на уровне связанного шаблона. |
| *Width* | Отображается ширина прототипа графика. |
| *Height* | Отображается высота прототипа графика. |

| Column | Description |
|---|---|
| *Type* | Отображается тип прототипа графика - *Normal*, *Stacked*, *Pie* или *Exploded*. |
| *Discover* | Обнаруживать график на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Вы можете переключаться между ’Yes’ и ’No’, щелкая по ним. |

Чтобы настроить новый прототип графика, нажмите кнопку *Create graph prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Удалить* - удалить эти прототипы графиков

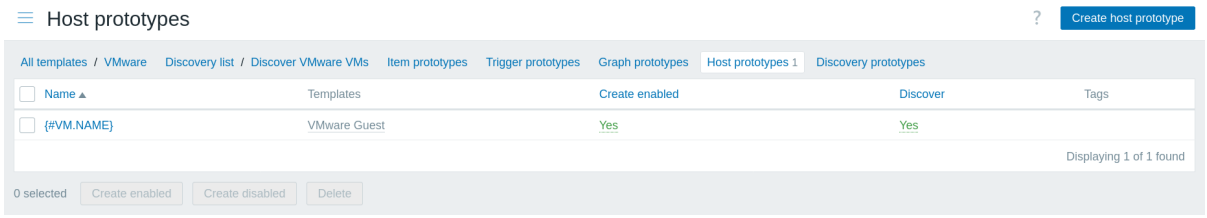
Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами графиков, затем нажмите нужную кнопку.

4 Прототипы узлов сети

Обзор

В этом разделе отображаются настроенные прототипы узлов сети правила обнаружения низкого уровня на шаблоне.

Если шаблон связан с узлом сети, прототипы узлов сети станут основой для создания реальных узлов сети во время обнаружения низкого уровня.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа узла сети, отображается как синяя ссылка.<br>При нажатии на имя открывается форма настройки прототипа узла сети.<br>Если прототип узла сети принадлежит связанному шаблону, имя шаблона отображается перед именем узла сети в виде серой ссылки. При нажатии на ссылку шаблона открывается список прототипов узлов сети на уровне связанного шаблона. |
| *Templates* | Отображаются шаблоны прототипа узла сети. |
| *Create enabled* | Создавать узел на основе этого прототипа как:<br>**Yes** - включено<br>**No** - отключено. Вы можете переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Discover* | Обнаруживать узел на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Вы можете переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Tags* | Отображаются теги прототипа узла сети. |

Чтобы настроить новый прототип узла сети, нажмите кнопку *Create host prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Создать включенными* - создать эти узлы сети как *Включенные*
- *Создать отключенными* - создать эти узлы сети как *Отключенные*
- *Удалить* - удалить эти прототипы узлов сети

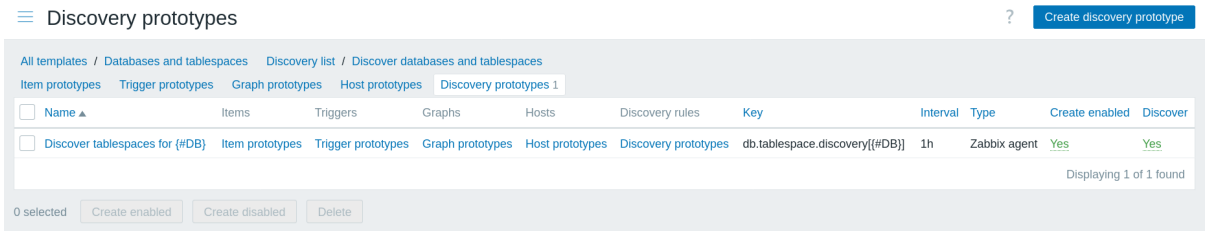
Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами узлов сети, затем нажмите нужную кнопку.

5 Прототипы обнаружения

Обзор

В этом разделе отображаются настроенные прототипы обнаружения правила низкоуровневого обнаружения на шаблоне.

Если шаблон связан с узлом сети, прототипы обнаружения станут основой для создания реальных прототипов обнаружения родительского правила обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа обнаружения, отображаемое как синяя ссылка.<br>При нажатии на имя открывается форма настройки прототипа обнаружения.<br>Если прототип обнаружения принадлежит связанному шаблону, имя шаблона отображается перед именем прототипа обнаружения как серая ссылка. При нажатии на ссылку шаблона открывается список прототипов обнаружения на уровне связанного шаблона. |
| *Items* | Отображается ссылка на список прототипов элементов данных.<br>Количество существующих прототипов элементов данных отображается серым цветом. |
| *Triggers* | Отображается ссылка на список прототипов триггеров.<br>Количество существующих прототипов триггеров отображается серым цветом. |
| *Graphs* | Отображается ссылка на список прототипов графиков.<br>Количество существующих прототипов графиков отображается серым цветом. |
| *Hosts* | Отображается ссылка на список прототипов узлов сети.<br>Количество существующих прототипов узлов сети отображается серым цветом. |
| *Discovery rules* | Отображается ссылка на список прототипов обнаружения.<br>Количество существующих прототипов обнаружения отображается серым цветом. |
| *Key* | Отображается ключ элемента данных, используемый для обнаружения. |
| *Interval* | Отображается частота выполнения обнаружения. |
| *Type* | Отображается тип элемента данных, используемый для обнаружения (агент Zabbix, SNMP agent и т. д.). |
| *Create enabled* | Отображается статус создания прототипа обнаружения - создание включено (*Yes*) или создание отключено (*No*). Нажав на статус, вы можете изменить его - с Yes на No (и обратно). |
| *Discover* | Отображается статус обнаружения прототипа обнаружения - обнаруживать (*Yes*) или не обнаруживать (*No*). Нажав на статус, вы можете изменить его - с Yes на No (и обратно). |

Чтобы настроить новый прототип обнаружения, нажмите кнопку *Create discovery prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Создать включенными* - создать эти прототипы обнаружения как *Включенные*
- *Создать отключенными* - создать эти прототипы обнаружения как *Отключенные*
- *Удалить* - удалить эти прототипы обнаружения

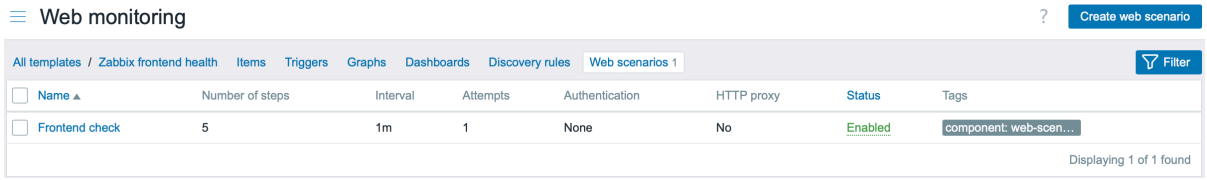
Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами обнаружения, затем нажмите нужную кнопку.

5 Веб-сценарии

Обзор

Список веб-сценариев для шаблона можно открыть в разделе *Сбор данных* → *Шаблоны*, нажав *Web* для соответствующего шаблона.

Отображается список существующих веб-сценариев.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя веб-сценария. При нажатии на имя веб-сценария открывается форма настройки веб-сценария.<br>Если веб-сценарий унаследован от другого шаблона, имя шаблона отображается перед именем веб-сценария в виде серой ссылки. При нажатии на ссылку шаблона открывается список веб-сценариев на уровне этого шаблона. |
| *Number of steps* | Количество шагов, которые содержит сценарий. |
| *Update interval* | Как часто выполняется сценарий. |
| *Attempts* | Сколько попыток выполнения шагов веб-сценария выполняется. |
| *Authentication* | Отображается метод аутентификации - Basic, NTLM, Kerberos, Digest или None. |
| *HTTP proxy* | Отображается HTTP proxy или 'No', если он не используется. |
| *Status* | Отображается статус веб-сценария - *Enabled* или *Disabled*.<br>Нажав на статус, вы можете изменить его. |
| *Tags* | Отображаются теги веб-сценария.<br>Можно отобразить до трех тегов (пар name:value). Если тегов больше, отображается ссылка "..." , позволяющая увидеть все теги при наведении курсора мыши. |

Чтобы настроить новый веб-сценарий, нажмите кнопку *Create web scenario* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Включить* - изменить статус сценария на *Включено*
- *Отключить* - изменить статус сценария на *Отключено*
- *Удалить* - удалить веб-сценарии

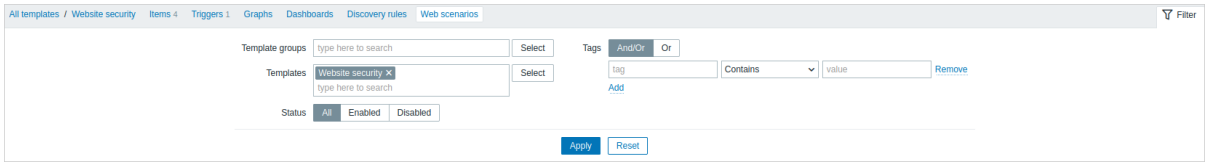
Чтобы использовать эти параметры, отметьте флажки перед соответствующими веб-сценариями, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те сценарии, которые вас интересуют.

Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком веб-сценариев. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно отфильтровать сценарии по группе шаблонов, шаблону, статусу и тегам.

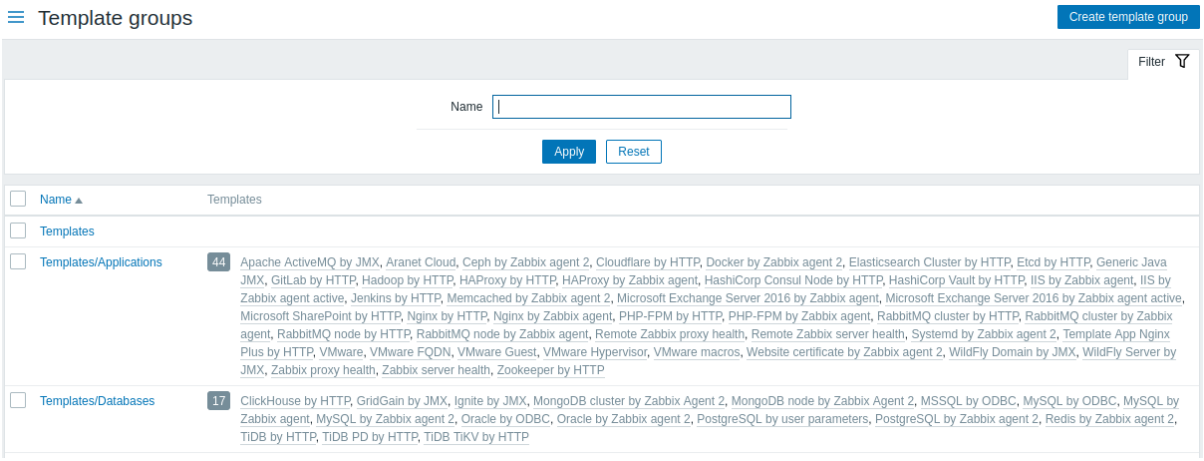



**1 Группы шаблонов**

Обзор

В разделе *Сбор данных* → *Группы шаблонов* пользователи могут настраивать и поддерживать группы шаблонов.

Отображается список существующих групп шаблонов с их подробностями. Вы можете искать и фильтровать группы шаблонов по имени.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя группы шаблонов. Щелчок по имени группы открывает форму настройки группы. |
| *Templates* | Количество шаблонов в группе (отображается серым цветом), за которым следует список участников группы.<br>Щелчок по имени шаблона откроет форму настройки шаблона.<br>Щелчок по числу открывает список шаблонов в этой группе. |

Параметры массового редактирования

Чтобы удалить несколько групп шаблонов одновременно, отметьте флажки перед соответствующими группами, затем нажмите кнопку Delete под списком.

Использование фильтра

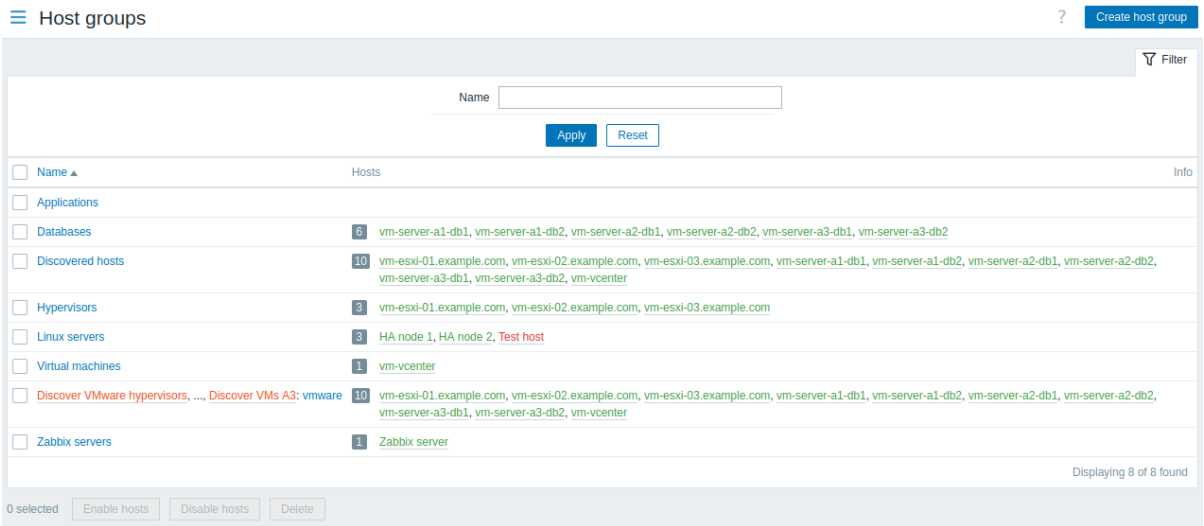
Вы можете использовать фильтр, чтобы отображать только те группы шаблонов, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

### 2 Группы узлов сети

Обзор

В разделе *Сбор данных* → *Группы узлов сети* пользователи могут настраивать и поддерживать группы узлов сети.

Отображается список существующих групп узлов сети с их подробностями. Вы можете выполнять поиск и фильтрацию групп узлов сети по имени.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя группы узлов сети. Щелчок по имени группы открывает форму настройки группы.<br>Обнаруженные группы узлов сети отображаются с именами правил обнаружения низкого уровня в качестве префикса. Щелчок по имени правила LLD открывает форму настройки прототипа узла сети. Обратите внимание, что обнаруженные группы узлов сети удаляются, когда удаляются все правила LLD, которые их обнаружили. |
| *Hosts* | Количество узлов сети в группе (отображается серым цветом), за которым следует список участников группы.<br>Щелчок по имени узла сети откроет форму настройки узла сети.<br>Щелчок по числу отфильтрует в общем списке узлов сети те, которые принадлежат к этой группе. |
| *Info* | Отображается информация об ошибке (если есть), относящаяся к группе узлов сети. |

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Включить узлы сети* - изменить статус всех узлов сети в группе на ”Контролируется”
- *Отключить узлы сети* - изменить статус всех узлов сети в группе на ”Не контролируется”
- *Удалить* - удалить группы узлов сети

Чтобы использовать эти параметры, отметьте флажки перед соответствующими группами узлов сети, затем нажмите нужную кнопку.

Использование фильтра

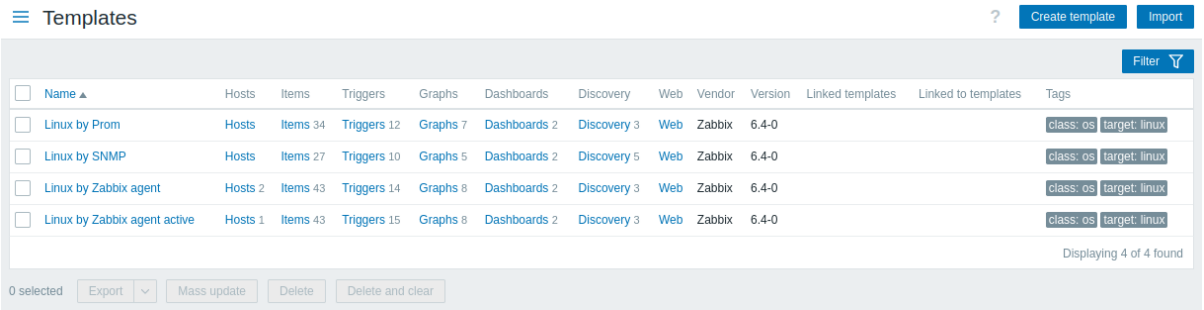
Вы можете использовать фильтр, чтобы отображать только те группы узлов сети, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

## 3 Шаблоны

Обзор

В разделе *Сбор данных* → *Шаблоны* пользователи могут настраивать и поддерживать шаблоны.

Отображается список существующих шаблонов с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя шаблона.<br>При нажатии на имя шаблона открывается форма настройки. |
| *Hosts* | Количество редактируемых узлов сети, к которым привязан шаблон; узлы сети только для чтения не учитываются.<br>При нажатии *Hosts* откроется список узлов сети с фильтром только по тем узлам, которые привязаны к шаблону. |
| *Entities (Items, Triggers, Graphs, Dashboards, Discovery, Web)* | Количество соответствующих сущностей в шаблоне (отображается серым цветом).<br>При нажатии на имя сущности в общем списке этой сущности будут отфильтрованы те, которые относятся к шаблону. |
| *Linked templates* | Шаблоны, связанные с шаблоном. |
| *Linked to templates* | Шаблоны, с которыми шаблон связан. |

| Column | Description |
|---|---|
| *Vendor*, *Version* | Поставщик и версия шаблона; отображаются, если конфигурация шаблона содержит такую информацию, и только для шаблонов из коробки, импортированных шаблонов или шаблонов, измененных через Template API.<br>Для шаблонов из коробки версия отображается следующим образом: основная версия Zabbix, разделитель (”-”), номер ревизии (увеличивается с каждой новой версией шаблона и сбрасывается с каждой новой основной версией Zabbix). Например, 7.0-0, 7.0-3, 7.4-0, 7.4-3. |
| *Tags* | Теги шаблона, без раскрытия макросов. |

Чтобы настроить новый шаблон, нажмите кнопку *Create template* в правом верхнем углу.

Чтобы импортировать шаблон из файла YAML, XML или JSON, нажмите кнопку *Import* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько параметров массового редактирования:

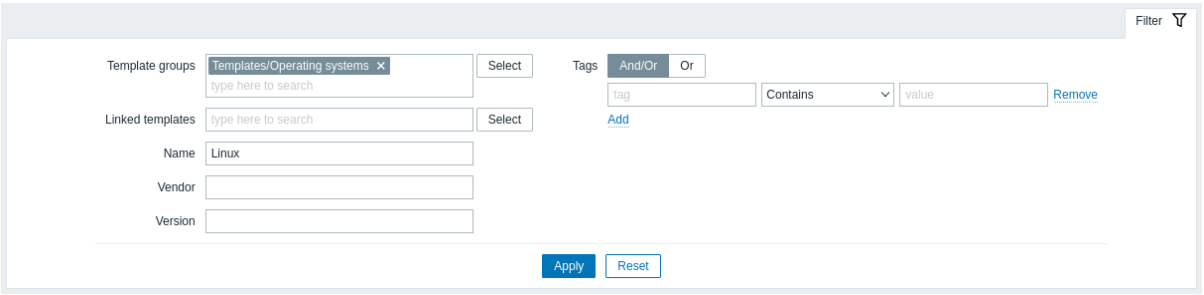
- *Export* - экспортировать шаблон в файл YAML, XML или JSON;
- *Mass update* - обновить несколько свойств для нескольких шаблонов одновременно;
- *Delete* - удалить шаблон, оставив его связанные сущности (элементы данных, триггеры и т. д.) на узлах сети;
- *Delete and clear* - удалить шаблон и его связанные сущности с узлов сети.

Чтобы использовать эти параметры, отметьте флажки перед соответствующими шаблонами, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только интересующие вас шаблоны. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна под кнопками *Create template* и *Import*. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать шаблоны по группе шаблонов, связанным шаблонам, имени и тегам.




| Parameter | Description |
|---|---|
| *Template groups* | Фильтрация по одной или нескольким группам шаблонов.<br>При указании родительской группы шаблонов автоматически выбираются все вложенные группы. |
| *Linked templates* | Фильтрация по непосредственно связанным шаблонам. |
| *Name* | Фильтрация по имени шаблона. |
| *Vendor* | Фильтрация по поставщику шаблона. |
| *Version* | Фильтрация по версии шаблона. |

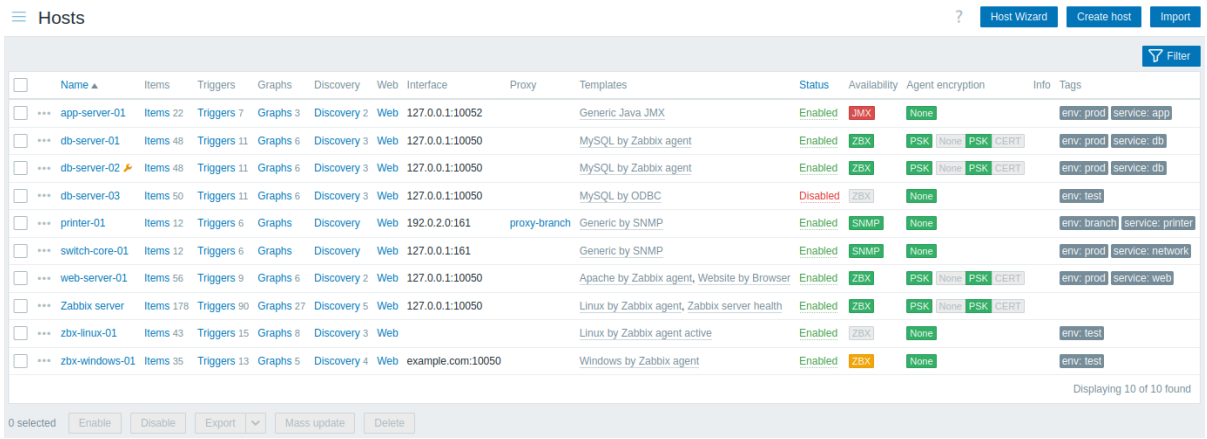
| Parameter | Description |
|---|---|
| *Tags* | Фильтрация по имени и значению тега шаблона.<br>Фильтрация возможна только по тегам уровня шаблона (не унаследованным). Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия. |

### 4 Узлы сети

Обзор

В разделе *Сбор данных* → *Узлы сети* пользователи могут настраивать и обслуживать узлы сети.

Отображается список существующих узлов сети с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Host menu* | Нажмите значок с тремя точками, чтобы открыть меню узла сети. |
| *Name* | Имя узла сети.<br>Щелчок по имени узла сети открывает форму настройки.<br>Оранжевый значок гаечного ключа после имени узла сети указывает, что этот узел сети находится в обслуживании. Подробности обслуживания отображаются при наведении указателя мыши на значок. |
| *Entities (Items, Triggers, Graphs, Discovery, Web)* | Щелчок по имени сущности отображает элементы данных, триггеры и т. д. узла сети.<br>Количество соответствующих сущностей отображается серым цветом. |
| *Interface* | Отображается основной интерфейс узла сети. |

| Column | Description |
|---|---|
| *Proxy* | Назначенные прокси отображаются в этом столбце:<br><Proxy name> - узел сети контролируется отдельным прокси (даже если прокси входит в группу прокси);<br><Proxy group name: proxy name> - узел сети контролируется группой прокси, и сервер Zabbix назначил прокси для контроля узла сети;<br><Proxy group name> - узел сети контролируется группой прокси без каких-либо прокси, либо если сервер Zabbix не назначил прокси для контроля узла сети;<br>Nothing - узел сети не контролируется ни прокси, ни группой прокси.<br>Этот столбец отображается только если для параметра фильтра *Monitored by* установлено значение 'Any', 'Proxy' или 'Proxy group'. |
| *Templates* | Отображаются шаблоны, связанные с узлом сети.<br>Если в связанном шаблоне содержатся другие шаблоны, они отображаются в скобках через запятую.<br>Щелчок по имени шаблона открывает его форму настройки. |
| *Status* | Отображается статус узла сети - *Enabled* или *Disabled*.<br>Щелкнув по статусу, вы можете изменить его вручную.<br>Обнаруженные узлы сети, которые были потеряны, помечаются значком информации.<br>Текст всплывающей подсказки содержит сведения об их статусе. |

| Column | Description |
|---|---|
| *Availability* | Отображается доступность узла сети по настроенным интерфейсам.<br><br>Значки доступности отражают текущее состояние интерфейса узла сети на сервере Zabbix. Поэтому если вы отключите узел сети во веб-интерфейсе, его доступность обновится после того, как сервер Zabbix синхронизирует изменения конфигурации. Аналогично, если вы включите узел сети, его доступность обновится после того, как сервер Zabbix синхронизирует изменения конфигурации и опросит узел сети.<br><br>Значки доступности отражают только те типы интерфейсов (Agent, SNMP, IPMI, JMX), которые настроены.<br>При наведении указателя на значок отображается всплывающее окно со списком всех интерфейсов того же типа с подробностями, статусом и ошибками. Для интерфейса Agent во всплывающем окне отображаются интерфейсы (passive) и активные проверки. Если у узла сети настроены только активные проверки, значок интерфейса Agent отображается даже в том случае, если у узла сети не настроен интерфейс Agent.<br>Для узлов сети без интерфейсов этот столбец пуст.<br><br>Статус отдельного интерфейса узла сети определяется соединением между включенным элементом данных, использующим этот интерфейс, и узлом сети. Статус может быть следующим:<br>**Available** - соединение с узлом сети успешно;<br>**Not available** - соединение с узлом сети не удалось (тайм-аут, проблемы с брандмауэром и т. д.);<br>**Unknown** - попытка соединения с узлом сети не выполнялась или результат неизвестен.<br>Дополнительные сведения о том, как сервер Zabbix определяет статус интерфейса, см. в разделах Unknown interface status и Unreachable/unavailable host interface settings.<br><br>Статус всех интерфейсов узла сети одного типа (Agent, SNMP, IPMI, JMX) определяется по тем интерфейсам, которые используются как минимум одним включенным элементом данных. Статус обозначается цветом значка:<br>**Green** - все интерфейсы доступны;<br>**Yellow** - как минимум один интерфейс недоступен, и как минимум один доступен или неизвестен;<br>**Red** - все интерфейсы недоступны;<br>**Gray** - как минимум один интерфейс неизвестен, но ни один не недоступен.<br><br>**Доступность активных проверок.** Если на узле сети включена хотя бы одна активная проверка, активные проверки также влияют на общую доступность интерфейса Agent, как описано выше. Для определения доступности активных проверок сообщения heartbeat отправляются в потоке активных проверок агента. Частота сообщений heartbeat задается параметром `HeartbeatFrequency` в конфигурации Zabbix агент или agent 2 (по умолчанию 60 секунд, диапазон 0-3600). Активные проверки считаются недоступными, если heartbeat активной проверки старше, чем 2 x HeartbeatFrequency секунд.<br>**Примечание:** агенты Zabbix версии ниже 6.2.x не отправляют heartbeat активных проверок, поэтому доступность их узлов сети остается неизвестной. |
| *Agent encryption* | Отображается статус шифрования для соединений с узлом сети и от узла сети:<br>**None** - шифрование отсутствует;<br>**PSK** - используется предварительно общий ключ;<br>**Cert** - используется сертификат. |
| *Info* | Отображается информация об ошибках, если они есть, для узла сети. |
| *Tags* | Теги узла сети с неразрешенными макросами. |

Чтобы настроить новый узел сети, нажмите кнопку *Host Wizard* или *Create host* в правом верхнем углу. Чтобы импортировать узел сети из файла YAML, XML или JSON, нажмите кнопку *Import* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько параметров массового редактирования:

- *Enable* - изменить статус узла сети на *Monitored*;
- *Disable* - изменить статус узла сети на *Not monitored*;
- *Export* - экспортировать узлы сети в файл YAML, XML или JSON;

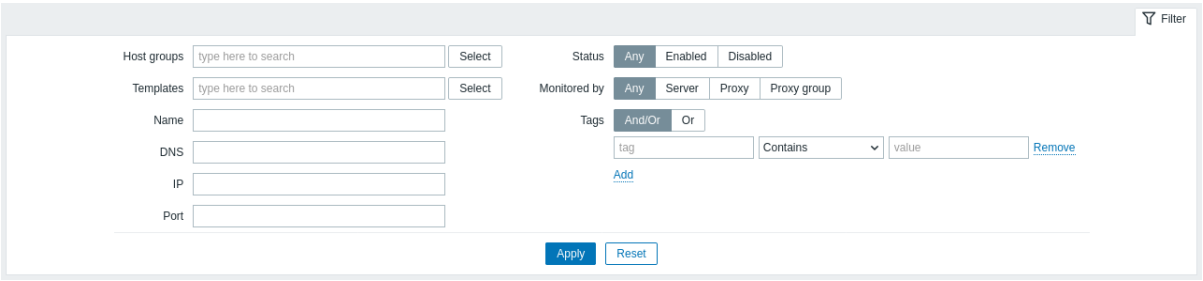
- *Mass update* - обновить несколько свойств для нескольких узлов сети одновременно;
- *Delete* - удалить узлы сети.

Чтобы использовать эти параметры, отметьте флажки перед соответствующими узлами сети, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только интересующие вас узлы сети. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Значок *Фильтр* доступен в правом верхнем углу. При нажатии на него откроется фильтр, в котором можно указать нужные критерии фильтрации.




| Parameter | Description |
|---|---|
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>При указании родительской группы узлов сети автоматически выбираются все вложенные группы узлов сети. |
| *Templates* | Фильтрация по связанным шаблонам. |
| *Name* | Фильтрация по видимому имени узла сети. |
| *DNS* | Фильтрация по DNS-имени. |
| *IP* | Фильтрация по IP-адресу. |
| *Port* | Фильтрация по номеру порта. |
| *Status* | Фильтрация по статусу узла сети. |
| *Monitored by* | Фильтрация узлов сети, которые контролируются сервером Zabbix, прокси или группой прокси. |
| *Proxies* | Фильтрация узлов сети, которые контролируются указанными здесь прокси. Это поле доступно только если в поле *Monitored by* выбрано значение ”Proxy”. |
| *Proxy groups* | Фильтрация узлов сети, которые контролируются указанными здесь группами прокси. Это поле доступно только если в поле *Monitored by* выбрано значение ”Proxy group”. |
| *Tags* | Фильтрация по имени и значению тега узла сети.<br>Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, значения которых содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, значения которых содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия. |

Неизвестный статус интерфейса

Сервер Zabbix отображает статус ”Unknown” для интерфейса узла сети (Agent, SNMP, IPMI, JMX) в следующих случаях:

- Узел сети отключен.
- Для узла сети задано наблюдение через прокси, другой прокси или сервер, если ранее он наблюдался через прокси.

- Узел сети наблюдается прокси, который, по-видимому, не в сети (обновления от прокси не получались в течение максимального интервала heartbeat - 1 час).
- Все элементы данных узла сети с этим типом интерфейса отключены.
- Не настроены poller'ы для этого типа интерфейса (например, параметр конфигурации сервера `StartAgentPollers` или `StartSNMPPollers` установлен в 0).

Доступность интерфейса устанавливается в "Unknown" после синхронизации кэша конфигурации сервера Zabbix.

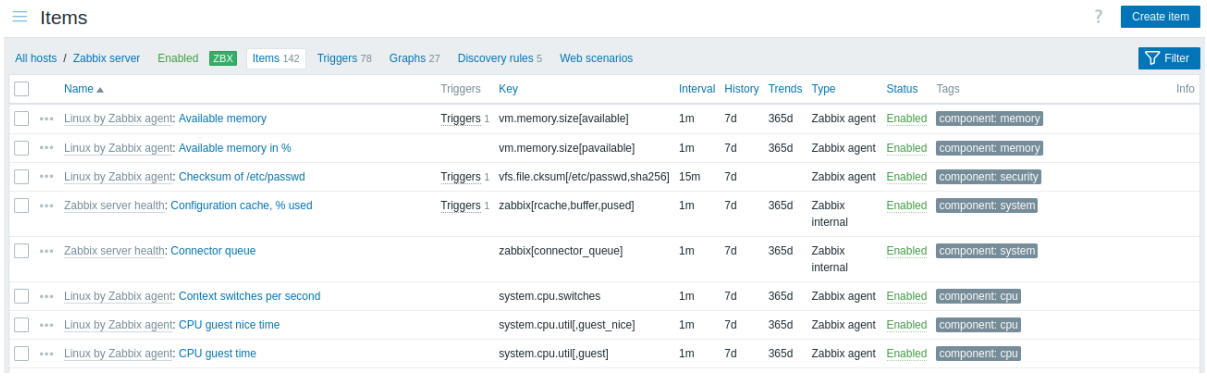
Доступность интерфейса (Available/Not available) на узлах сети, наблюдаемых через прокси, восстанавливается после синхронизации кэша конфигурации прокси.

1 Элементы данных

Обзор

Список элементов данных для узла сети можно открыть из *Сбор данных* → *Узлы сети*, нажав *Элементы данных* для соответствующего узла сети.

Отображается список существующих элементов данных.




Отображаемые данные:

| Column | Description |
|---|---|
| *Item menu* | Нажмите значок с тремя точками, чтобы открыть меню элемента данных. |
| *Host* | Узел сети элемента данных.<br>Нажатие на имя узла сети открывает форму настройки.<br>Этот столбец отображается только если в фильтре выбрано несколько узлов сети или не выбрано ни одного. |
| *Name* | Имя элемента данных, отображаемое в виде синей ссылки на подробные сведения об элементе данных.<br>Нажатие на ссылку с именем элемента данных открывает форму настройки.<br>Если элемент данных узла сети принадлежит шаблону, перед именем элемента данных серой ссылкой отображается имя шаблона. Нажатие на ссылку шаблона откроет список элементов данных на уровне шаблона.<br>Если элемент данных был создан из прототипа элемента данных, перед его именем оранжевым цветом отображается имя правила низкоуровневого обнаружения. Нажатие на имя правила обнаружения откроет список прототипов элементов данных. |
| *Triggers* | При наведении указателя мыши на Triggers будет отображаться информационное окно с триггерами, связанными с элементом данных.<br>Количество триггеров отображается серым цветом. |
| *Key* | Отображается ключ элемента данных. |
| *Interval* | Отображается частота проверки.<br>*Примечание*: пассивные элементы данных также можно проверить немедленно, нажав кнопку *Execute now* button. |
| *History* | Отображается, сколько дней будет храниться история данных элемента данных. |
| *Trends* | Отображается, сколько дней будет храниться история трендов элемента данных. |
| *Type* | Отображается тип элемента данных (агент Zabbix, SNMP agent, simple check и т. д.). |

| Column | Description |
|---|---|
| *Status* | Отображается статус элемента данных - *Enabled*, *Disabled* или *Not supported*. Вы можете изменить статус вручную, щелкнув по нему - с Enabled на Disabled (и обратно); с Not supported на Disabled (и обратно).<br>Обнаруженные элементы данных, которые были потеряны, помечаются значком информации. Текст всплывающей подсказки содержит сведения об их статусе. |
| *Tags* | Отображаются теги элемента данных.<br>Можно отобразить до трех тегов (пар имя:значение). Если тегов больше, отображается ссылка "..." , позволяющая увидеть все теги при наведении указателя мыши. |
| *Info* | Если элемент данных работает корректно, в этом столбце значок не отображается. В случае ошибок отображается квадратный значок с буквой "i". Наведите указатель мыши на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |

Чтобы настроить новый элемент данных, нажмите кнопку *Create item* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

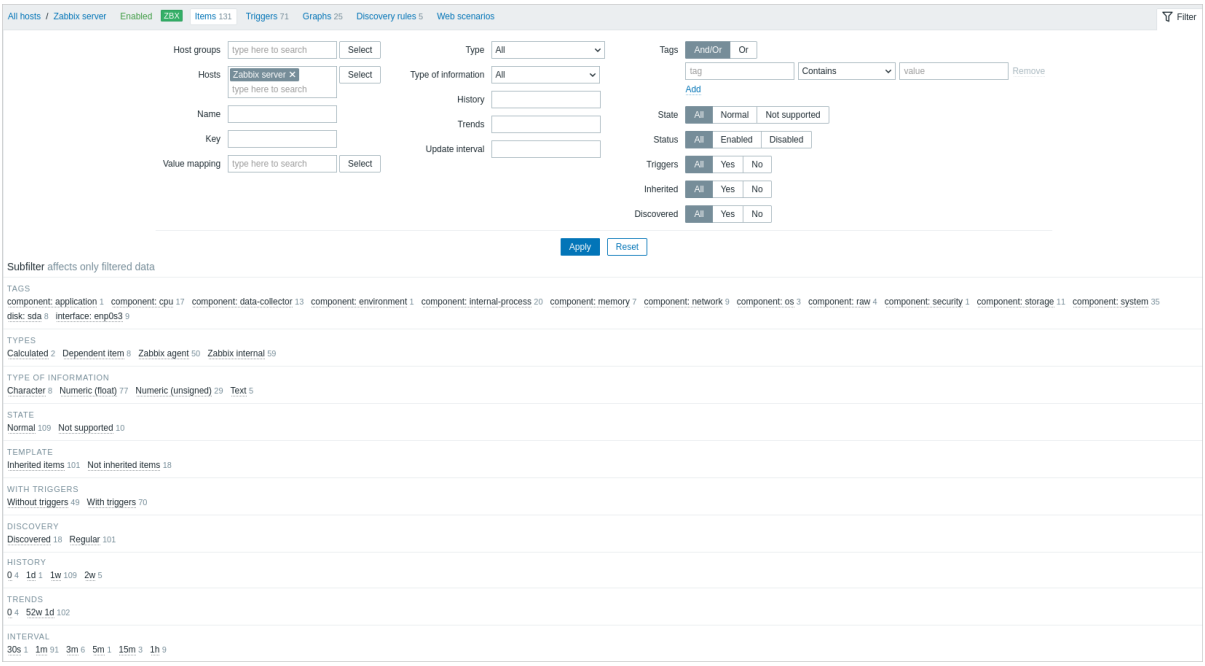
- *Enable* - изменить статус элемента данных на *Enabled*
- *Disable* - изменить статус элемента данных на *Disabled*
- *Execute now* - немедленно выполнить проверку новых значений элемента данных. Поддерживается только для **пассивных** проверок (см. подробнее). Обратите внимание, что при немедленной проверке значений кэш конфигурации не обновляется, поэтому значения не будут отражать самые недавние изменения в конфигурации элемента данных.
- *Clear history and trends* - удалить историю и данные трендов для элементов данных.
- *Copy* - скопировать элементы данных на другие узлы сети или шаблоны.
- *Mass update* - обновить несколько свойств для ряда элементов данных одновременно.
- *Delete* - удалить элементы данных.

Чтобы использовать эти параметры, отметьте флажки напротив соответствующих элементов данных, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только интересующие вас элементы данных. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Значок *Filter* доступен в правом верхнем углу. При нажатии на него откроется фильтр, где можно указать нужные критерии фильтрации.

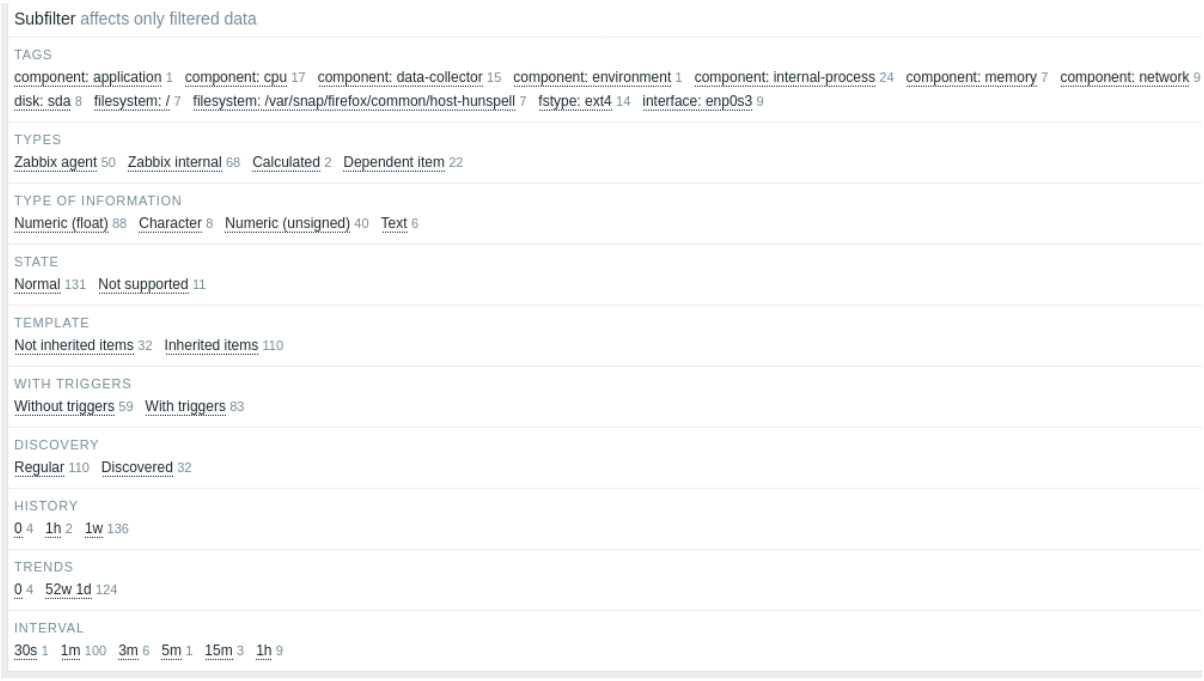


| Parameter | Description |
|---|---|
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети.<br>Группы узлов сети, содержащие только шаблоны, выбрать нельзя. |
| *Hosts* | Фильтрация по одному или нескольким узлам сети. |
| *Name* | Фильтрация по имени элемента данных. |
| *Key* | Фильтрация по ключу элемента данных. |
| *Value mapping* | Фильтрация по используемой карте значений.<br>Этот параметр не отображается, если параметр *Hosts* пуст. |
| *Type* | Фильтрация по типу элемента данных (агент Zabbix, SNMP agent и т. д.). |
| *Type of information* | Фильтрация по типу информации (беззнаковое число, float и т. д.). |
| *History* | Фильтрация по времени хранения истории элемента данных. |
| *Trends* | Фильтрация по времени хранения трендов элемента данных. |
| *Update interval* | Фильтрация по интервалу обновления элемента данных. |
| *Tags* | Укажите теги, чтобы ограничить число отображаемых элементов данных. Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, где значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, где значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия |
| *State* | Фильтрация по состоянию элемента данных - *Normal* или *Not supported*. |
| *Status* | Фильтрация по статусу элемента данных - *Enabled* или *Disabled*. |
| *Triggers* | Фильтрация элементов данных с триггерами или без них. |
| *Inherited* | Фильтрация элементов данных, унаследованных или не унаследованных от шаблона. |
| *Discovery* | Фильтрация элементов данных, обнаруженных или не обнаруженных низкоуровневым обнаружением. |

Использование подфильтра

Подфильтр позволяет дополнительно изменять фильтрацию, заданную в основном фильтре.

Он содержит кликабельные ссылки для быстрого доступа к связанным элементам. Элементы связаны по общему признаку - тегу, типу элемента данных, состоянию элемента данных, интервалу обновления элемента данных и т. д. При нажатии на признак он выделяется серым фоном, а элементы данных немедленно фильтруются (нет необходимости нажимать *Применить* в основном фильтре). При нажатии на другой признак он добавляется к отфильтрованным результатам. При повторном нажатии на признак фильтрация снимается.




Подфильтры формируются на основе отфильтрованных данных, количество которых ограничено 1000 записей. Если вы хотите видеть больше записей в подфильтре, необходимо увеличить значение параметра *Limit for search and filter results* в разделе *Administration -> General -> GUI*.

В отличие от основного фильтра, подфильтр обновляется при каждом запросе обновления таблицы, чтобы всегда иметь актуальную информацию о доступных вариантах фильтрации и их счетчиках.

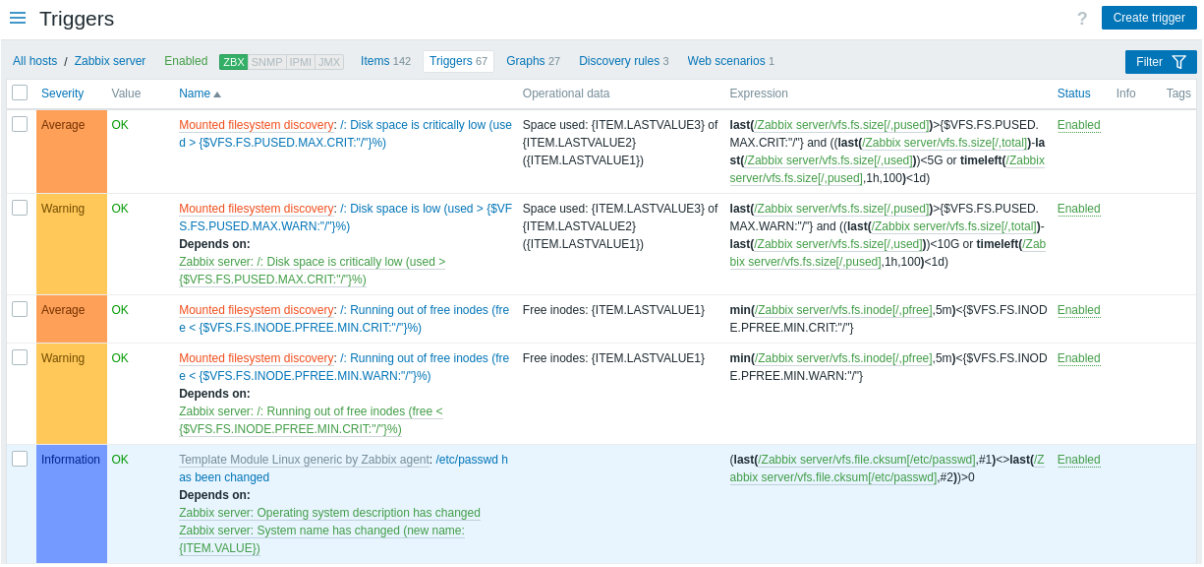
Количество отображаемых признаков ограничено 100 по горизонтали. Если их больше, в конце отображается значок с тремя точками; он не является кликабельным.

Число рядом с каждым кликабельным признаком показывает количество элементов данных, сгруппированных в нем (на основе результатов основного фильтра). Когда признак выбран, числа у других доступных признаков отображаются со знаком плюс, показывая, сколько элементов данных можно добавить к текущему выбору. Признаки без элементов данных не отображаются, если только они не были выбраны в подфильтре ранее.

2 Триггеры

Обзор

Список триггеров для узла сети можно открыть в разделе *Сбор данных → Узлы сети* , нажав *Триггеры* для соответствующего узла сети.




Отображаемые данные:

| Column | Description |
|---|---|
| *Severity* | Серьезность триггера отображается как названием, так и цветом фона ячейки. |
| *Value* | Отображается значение триггера:<br>**OK** - триггер находится в состоянии OK<br>**PROBLEM** - триггер находится в состоянии Problem |
| *Host* | Узел сети триггера.<br>Щелчок по имени узла сети открывает форму настройки узла сети.<br>Этот столбец отображается только если в фильтре выбрано несколько узлов сети или не выбрано ни одного. |
| *Name* | Имя триггера, отображаемое как синяя ссылка на сведения о триггере.<br>Щелчок по ссылке с именем триггера открывает форму настройки триггера.<br>Если триггер узла сети принадлежит шаблону, имя шаблона отображается перед именем триггера в виде серой ссылки. Щелчок по ссылке шаблона откроет список триггеров на уровне шаблона.<br>Если триггер был создан из прототипа триггера, перед его именем отображается имя правила низкоуровневого обнаружения оранжевым цветом. Щелчок по имени правила обнаружения открывает список прототипов триггеров. |
| *Operational data* | Определение оперативных данных триггера, содержащее произвольные строки и макросы, которые будут динамически подставляться в *Мониторинг → Проблемы*. |
| *Expression* | Отображается выражение триггера. Часть выражения, относящаяся к узлу сети и элементу данных, отображается как ссылка, ведущая к форме настройки элемента данных. |
| *Status* | Отображается статус триггера - *Enabled*, *Disabled* или *Unknown*. Нажав на статус, можно вручную изменить его - с Enabled на Disabled (и обратно); с Unknown на Disabled (и обратно).<br>Проблемы отключенного триггера больше не отображаются во веб-интерфейсе, но не удаляются.<br>Обнаруженные триггеры, которые были потеряны, помечаются значком информации. Подсказка содержит сведения об их статусе. |
| *Info* | Если все работает корректно, в этом столбце не отображается значок. В случае ошибок отображается квадратный значок с буквой ”i”. Наведите указатель мыши на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |
| *Tags* | Если триггер содержит теги, в этом столбце отображаются имя и значение тега. |

Чтобы настроить новый триггер, нажмите кнопку *Create trigger* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

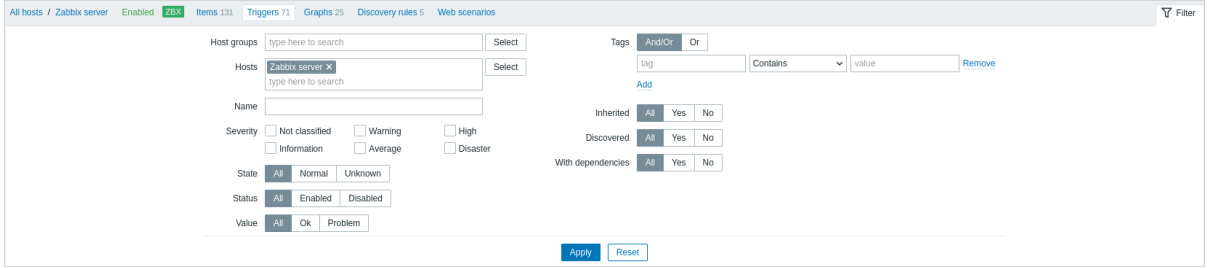
- *Включить* - изменить статус триггера на *Включен*.
- *Отключить* - изменить статус триггера на *Отключен*.
- *Копировать* - скопировать триггеры на другие узлы сети или шаблоны.
- *Массовое обновление* - обновить несколько свойств для ряда триггеров одновременно.
- *Удалить* - удалить триггеры.

Чтобы использовать эти параметры, отметьте флажки перед нужными триггерами, затем нажмите требуемую кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те триггеры, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Значок *Filter* доступен в правом верхнем углу. При нажатии на него откроется фильтр, в котором можно указать нужные критерии фильтрации.



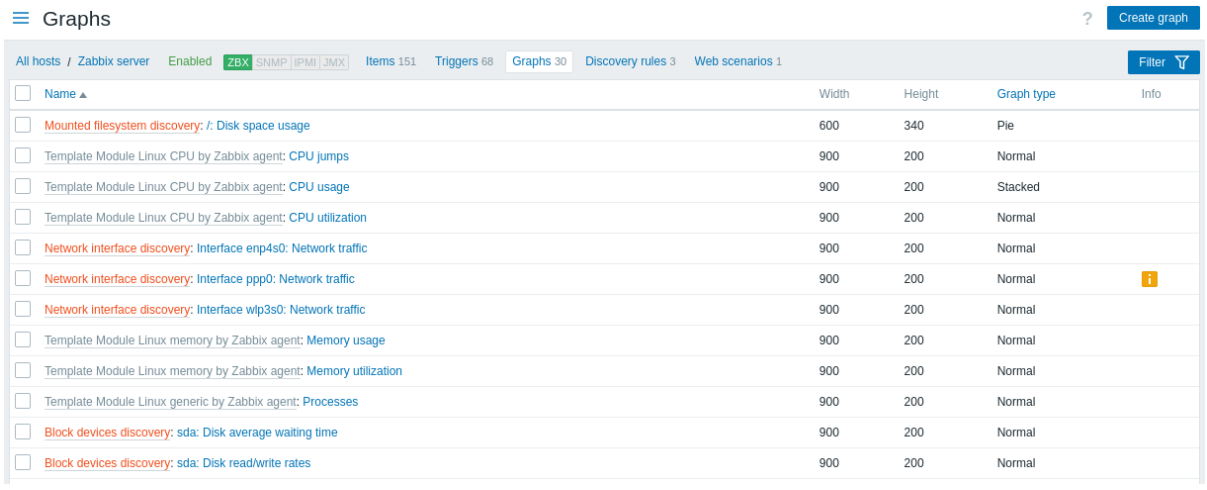
| Параметр | Описание |
|---|---|
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>При указании родительской группы узлов сети автоматически выбираются все вложенные группы узлов сети.<br>Группы узлов сети, содержащие только шаблоны, выбрать нельзя. |
| *Hosts* | Фильтрация по одному или нескольким узлам сети.<br>Если выше уже выбраны группы узлов сети, выбор узлов сети ограничивается этими группами. |
| *Name* | Фильтрация по имени триггера. |
| *Severity* | Выберите для фильтрации по одному или нескольким уровням серьезности триггера. |
| *State* | Фильтрация по состоянию триггера. |
| *Status* | Фильтрация по статусу триггера. |
| *Value* | Фильтрация по значению триггера. |
| *Tags* | Фильтрация по имени и значению тега триггера. Можно как включать, так и исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра)<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>**Does not exist** - исключить указанные имена тегов<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра)<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра)<br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, при этом условия с одинаковым именем тега будут сгруппированы по условию Or<br>**Or** - достаточно выполнения одного условия<br>В полях имени тега и значения тега поддерживаются макросы и macro functions. |
| *Inherited* | Фильтрация триггеров, унаследованных (или не унаследованных) от шаблона. |
| *Discovered* | Фильтрация триггеров, обнаруженных (или не обнаруженных) с помощью низкоуровневого обнаружения. |
| *With dependencies* | Фильтрация триггеров с зависимостями (или без зависимостей). |

3 Графики

Обзор

Список пользовательских графиков для узла сети можно открыть в разделе *Сбор данных* → *Узлы сети*, нажав *Графики* для соответствующего узла сети.

Отображается список существующих графиков.




Отображаемые данные:

| Column | Description |
|---|---|
| *Host* | Узел сети графика.<br>При нажатии на имя узла сети открывается форма настройки.<br>Этот столбец отображается только если в фильтре выбрано несколько узлов сети или не выбрано ни одного. |
| *Name* | Имя пользовательского графика, отображаемое как синяя ссылка на сведения о графике.<br>При нажатии на ссылку с именем графика открывается форма настройки.<br>Если график узла сети принадлежит шаблону, имя шаблона отображается перед именем графика в виде серой ссылки. При нажатии на ссылку шаблона откроется список графиков на уровне шаблона.<br>Если график был создан на основе прототипа графика, перед его именем указывается имя правила обнаружения низкого уровня, выделенное оранжевым цветом. При нажатии на имя правила обнаружения открывается список прототипов графиков. |
| *Width* | Отображается ширина графика. |
| *Height* | Отображается высота графика. |
| *Graph type* | Отображается тип графика - *Normal*, *Stacked*, *Pie* или *Exploded*. |
| *Info* | Если график работает корректно, в этом столбце значок не отображается. В случае ошибок отображается квадратный значок с буквой "i". Наведите указатель на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |

Чтобы настроить новый график, нажмите кнопку *Create graph* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Копировать* - копировать графики на другие узлы сети или шаблоны
- *Удалить* - удалить графики

Чтобы использовать эти параметры, отметьте флажки перед соответствующими графиками, затем нажмите нужную кнопку.

Использование фильтра

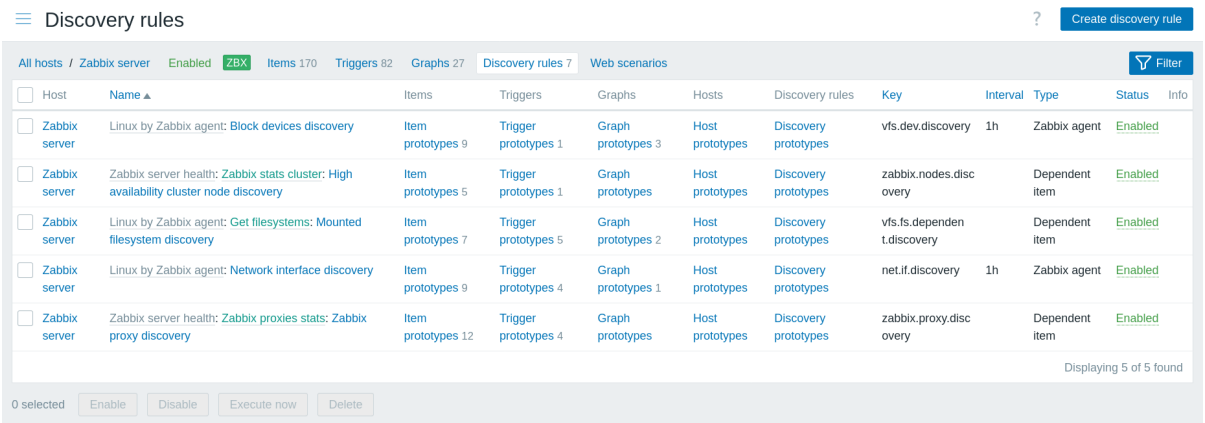
Вы можете фильтровать графики по группе узлов сети и узлу сети. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

4 Правила обнаружения

Обзор

Список правил низкоуровневого обнаружения для узла сети можно открыть из *Сбор данных* → *Узлы сети*, нажав *Обнаружение* для соответствующего узла сети.

Отображается список существующих правил низкоуровневого обнаружения. Также можно просматривать все правила обнаружения независимо от узла сети или все правила обнаружения определенной группы узлов сети, изменив параметры фильтра.




Отображаемые данные:

| Column | Description |
|---|---|
| *Host* | Отображается видимое имя узла сети.<br>При нажатии на имя узла сети открывается форма настройки.<br>Если видимое имя узла сети отсутствует, отображается техническое имя узла сети. |
| *Name* | Имя правила, отображаемое как синяя ссылка.<br>При нажатии на имя правила открывается форма настройки правила низкоуровневого обнаружения.<br>Если правило обнаружения принадлежит шаблону, имя шаблона отображается перед именем правила в виде серой ссылки. При нажатии на ссылку шаблона открывается список правил на уровне шаблона. |
| *Items* | Отображается ссылка на список прототипов элементов данных.<br>Количество существующих прототипов элементов данных отображается серым цветом. |
| *Triggers* | Отображается ссылка на список прототипов триггеров.<br>Количество существующих прототипов триггеров отображается серым цветом. |
| *Graphs* | Отображается ссылка на список прототипов графиков.<br>Количество существующих прототипов графиков отображается серым цветом. |
| *Hosts* | Отображается ссылка на список прототипов узлов сети.<br>Количество существующих прототипов узлов сети отображается серым цветом. |
| *Key* | Отображается ключ элемента данных, используемый для обнаружения. |
| *Interval* | Отображается частота выполнения обнаружения.<br>*Примечание.* Обнаружение также можно выполнить немедленно, нажав кнопку *Execute now* под списком. |
| *Type* | Отображается тип элемента данных, используемый для обнаружения (агент Zabbix, SNMP agent и т. д.). |
| *Status* | Отображается статус правила обнаружения - *Enabled*, *Disabled* или *Not supported*. Нажав на статус, можно изменить его - с Enabled на Disabled (и обратно); с Not supported на Disabled (и обратно). |
| *Info* | Если все в порядке, в этом столбце не отображается значок. В случае ошибок отображается квадратный значок с буквой ”i”. Наведите указатель на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |

Чтобы настроить новое правило низкоуровневого обнаружения, нажмите кнопку *Create discovery rule* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

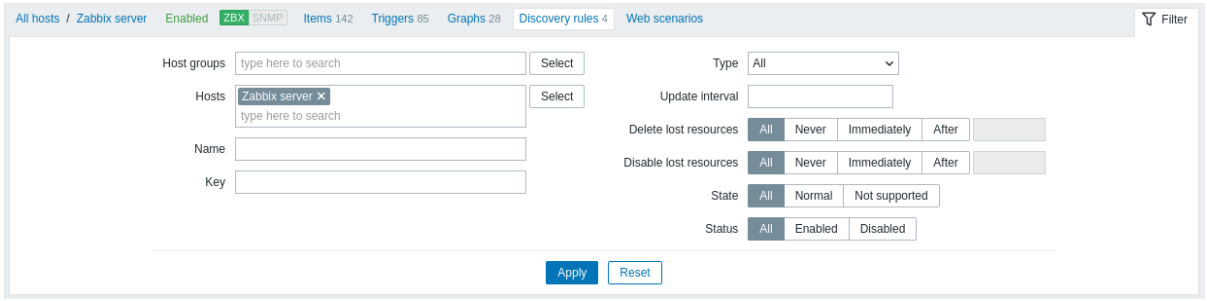
- *Enable* - изменить статус правила низкоуровневого обнаружения на *Enabled*.
- *Disable* - изменить статус правила низкоуровневого обнаружения на *Disabled*.
- *Execute now* - немедленно выполнить обнаружение на основе правил обнаружения. См. подробности. Обратите внимание, что при немедленном выполнении обнаружения кэш конфигурации не обновляется, поэтому результат не будет отражать самые недавние изменения в конфигурации правила обнаружения.
- *Delete* - удалить правила низкоуровневого обнаружения.

Чтобы использовать эти параметры, отметьте флажки перед соответствующими правилами обнаружения, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те правила обнаружения, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком правил обнаружения. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать правила обнаружения по группе узлов сети, узлу сети, имени, ключу элемента данных, типу элемента данных и другим параметрам.

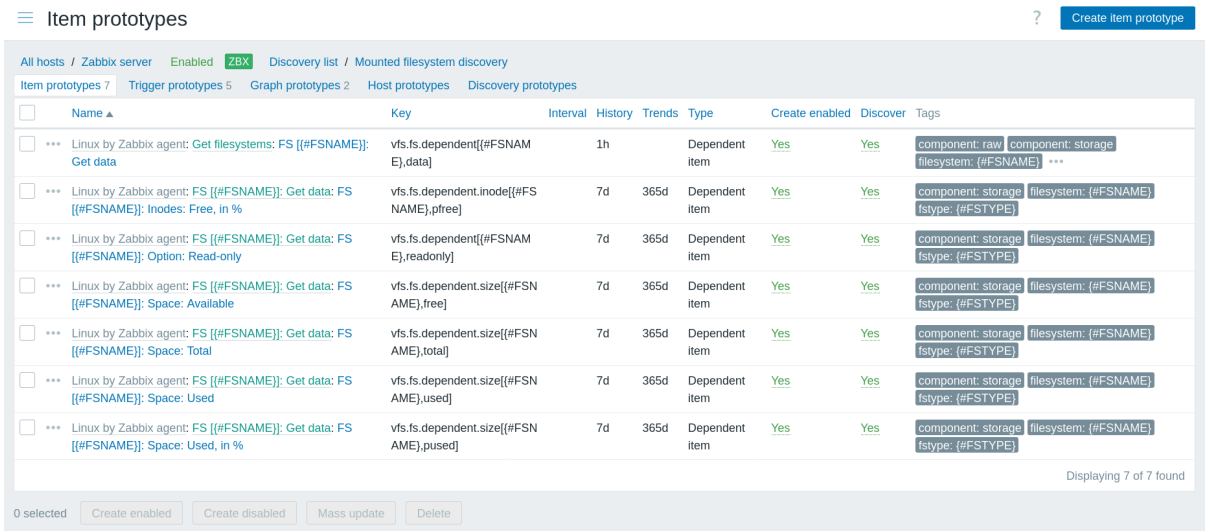



| Parameter | Description |
|---|---|
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети. |
| *Hosts* | Фильтрация по одному или нескольким узлам сети. |
| *Name* | Фильтрация по имени правила обнаружения. |
| *Key* | Фильтрация по ключу элемента данных обнаружения. |
| *Type* | Фильтрация по типу элемента данных обнаружения. |
| *Update interval* | Фильтрация по интервалу обновления.<br>Недоступно для Zabbix trapper и зависимых элементов данных. |
| *Delete lost resources* | Фильтрация по периоду *Delete lost resources*. |
| *Disable lost resources* | Фильтрация по периоду *Disable lost resources*. |
| *SNMP OID* | Фильтрация по SNMP OID.<br>Доступно только если в качестве типа выбран *SNMP agent*. |
| *State* | Фильтрация по состоянию правила обнаружения (All/Normal/Not supported). |
| *Status* | Фильтрация по статусу правила обнаружения (All/Enabled/Disabled). |

1 Прототипы элементов данных

Обзор

В этом разделе отображаются прототипы элементов данных правила низкоуровневого обнаружения на узле сети. Прототипы элементов данных являются основой реальных элементов данных узла сети, которые создаются во время низкоуровневого обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа элемента данных отображается как синяя ссылка.<br>При нажатии на имя открывается форма настройки прототипа элемента данных.<br>Если прототип элемента данных принадлежит шаблону, имя шаблона отображается перед именем правила как серая ссылка. При нажатии на ссылку шаблона открывается список прототипов элементов данных на уровне шаблона.<br>Если прототип элемента данных имеет тип ”Dependent item”, имя мастер-элемента отображается перед именем прототипа элемента данных зеленым цветом. |
| *Key* | Отображается ключ прототипа элемента данных. |
| *Interval* | Отображается частота проверки. |
| *History* | Отображается, сколько дней хранить историю данных элемента данных. |
| *Trends* | Отображается, сколько дней хранить историю трендов элемента данных. |
| *Type* | Отображается тип прототипа элемента данных (агент Zabbix, SNMP agent, simple check и т. д.). |
| *Create enabled* | Создавать элемент на основе этого прототипа как:<br>**Yes** - включено<br>**No** - отключено. Можно переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Discover* | Обнаруживать элемент на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Можно переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Tags* | Отображаются теги прототипа элемента данных. |

Чтобы настроить новый прототип элемента данных, нажмите кнопку *Create item prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

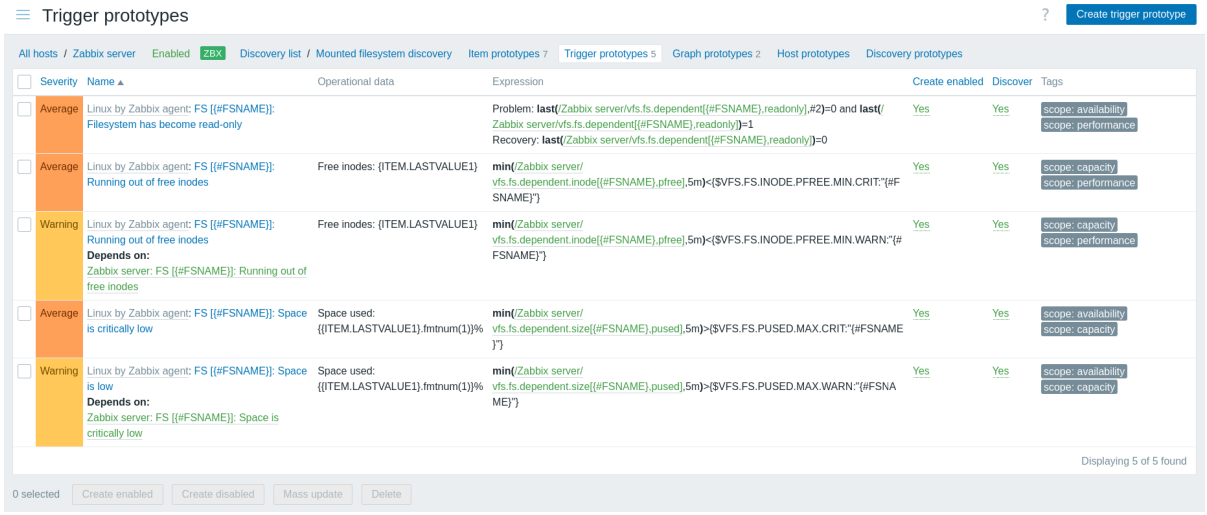
- *Создать включенными* - создать эти элементы данных как *Включенные*
- *Создать отключенными* - создать эти элементы данных как *Отключенные*
- *Массовое обновление* - массово обновить эти прототипы элементов данных
- *Удалить* - удалить эти прототипы элементов данных

Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами элементов данных, затем нажмите нужную кнопку.

2 Прототипы триггеров

Обзор

В этом разделе отображаются прототипы триггеров правила низкоуровневого обнаружения на узле сети. Прототипы триггеров являются основой реальных триггеров узла сети, которые создаются во время низкоуровневого обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа триггера, отображается как синяя ссылка.<br>Щелчок по имени открывает форму настройки прототипа триггера.<br>Если прототип триггера принадлежит связанному шаблону, имя шаблона отображается перед именем триггера в виде серой ссылки. Щелчок по ссылке шаблона открывает список прототипов триггеров на уровне связанного шаблона. |
| *Operational data* | Отображается формат оперативных данных триггера, содержащий произвольные строки и макросы, которые будут динамически подставляться в *Monitoring* → *Problems*. |
| *Create enabled* | Создать триггер на основе этого прототипа как:<br>**Yes** - включен<br>**No** - выключен. Вы можете переключаться между 'Yes' и 'No', щелкая по ним. |
| *Discover* | Обнаруживать триггер на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Вы можете переключаться между 'Yes' и 'No', щелкая по ним. |
| *Tags* | Отображаются теги прототипа триггера. |

Чтобы настроить новый прототип триггера, нажмите кнопку *Create trigger prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

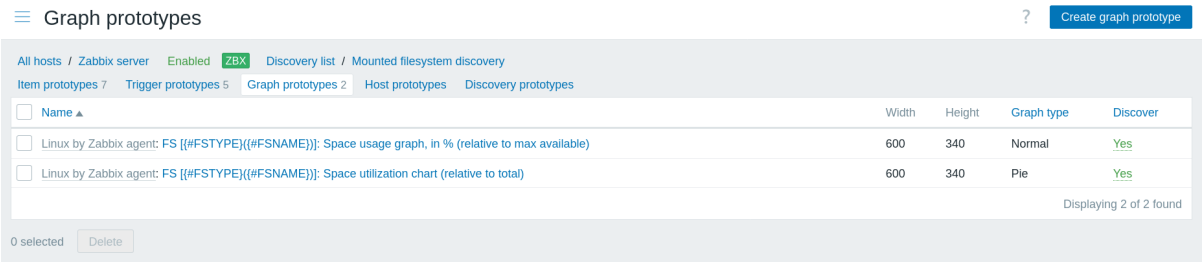
- *Создать включенными* - создать эти триггеры как *Включенные*
- *Создать отключенными* - создать эти триггеры как *Отключенные*
- *Массовое обновление* - массово обновить эти прототипы триггеров
- *Удалить* - удалить эти прототипы триггеров

Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами триггеров, затем нажмите нужную кнопку.

3 Прототипы графиков

Обзор

В этом разделе отображаются прототипы графиков правила низкоуровневого обнаружения на узле сети. Прототипы графиков являются основой реальных графиков узла сети, которые создаются во время низкоуровневого обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа графика, отображается как синяя ссылка.<br>Щелчок по имени открывает форму настройки прототипа графика.<br>Если прототип графика принадлежит связанному шаблону, имя шаблона отображается перед именем графика в виде серой ссылки. Щелчок по ссылке шаблона открывает список прототипов графиков на уровне связанного шаблона. |
| *Width* | Отображается ширина прототипа графика. |
| *Height* | Отображается высота прототипа графика. |
| *Type* | Отображается тип прототипа графика - *Normal*, *Stacked*, *Pie* или *Exploded*. |
| *Discover* | Обнаруживать график на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Вы можете переключаться между 'Yes' и 'No', щелкая по ним. |

Чтобы настроить новый прототип графика, нажмите кнопку *Create graph prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

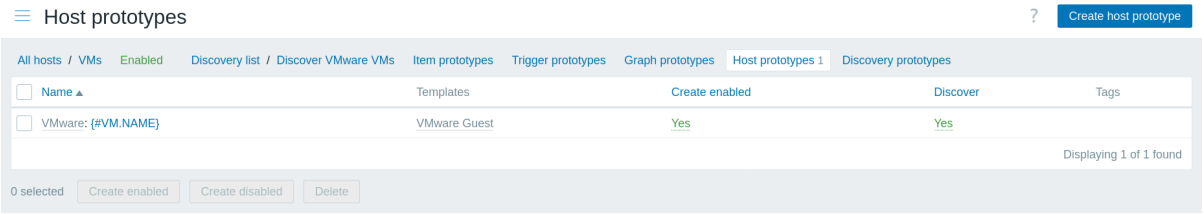
- *Удалить* - удалить эти прототипы графиков

Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами графиков, затем нажмите нужную кнопку.

4 Прототипы узлов сети

Обзор

В этом разделе отображаются прототипы узлов сети правила низкоуровневого обнаружения на узле сети. Прототипы узлов сети являются основой реальных узлов сети, которые создаются во время низкоуровневого обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа узла сети, отображаемое как синяя ссылка.<br>При нажатии на имя открывается форма настройки прототипа узла сети.<br>Если прототип узла сети принадлежит связанному шаблону, имя шаблона отображается перед именем узла сети в виде серой ссылки. При нажатии на ссылку шаблона открывается список прототипов узлов сети на уровне связанного шаблона. |
| *Templates* | Отображаются шаблоны прототипа узла сети. |
| *Create enabled* | Создать узел сети на основе этого прототипа как:<br>**Yes** - включено<br>**No** - отключено. Вы можете переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Discover* | Обнаруживать узел сети на основе этого прототипа:<br>**Yes** - обнаруживать<br>**No** - не обнаруживать. Вы можете переключаться между ’Yes’ и ’No’, нажимая на них. |
| *Tags* | Отображаются теги прототипа узла сети. |

Чтобы настроить новый прототип узла сети, нажмите кнопку *Create host prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

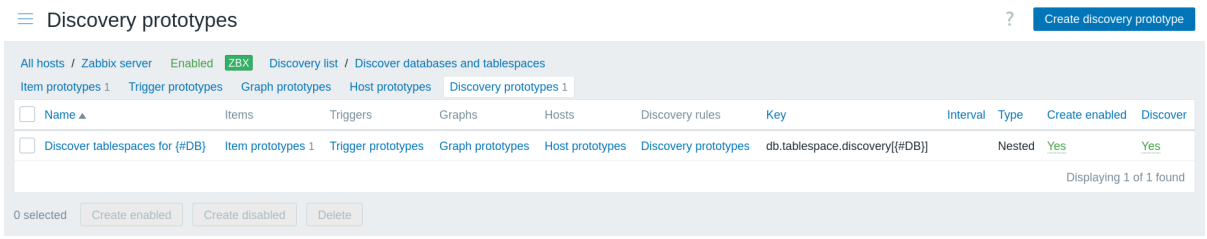
- *Создать включенными* - создать эти узлы сети как *Включенные*
- *Создать отключенными* - создать эти узлы сети как *Отключенные*
- *Удалить* - удалить эти прототипы узлов сети

Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами узлов сети, затем нажмите нужную кнопку.

5 Прототипы обнаружения

Обзор

В этом разделе отображаются прототипы обнаружения правила низкоуровневого обнаружения на узле сети. Прототипы обнаружения — это вложенные правила обнаружения родительского правила обнаружения.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прототипа обнаружения, отображаемое в виде синей ссылки.<br>Щелчок по имени открывает форму настройки прототипа обнаружения.<br>Если прототип обнаружения принадлежит связанному шаблону, имя шаблона отображается перед именем узла сети в виде серой ссылки. Щелчок по ссылке шаблона открывает список прототипов обнаружения на уровне связанного шаблона. |
| *Items* | Отображается ссылка на список прототипов элементов данных.<br>Количество существующих прототипов элементов данных отображается серым цветом. |
| *Triggers* | Отображается ссылка на список прототипов триггеров.<br>Количество существующих прототипов триггеров отображается серым цветом. |
| *Graphs* | Отображается ссылка на список прототипов графиков.<br>Количество существующих прототипов графиков отображается серым цветом. |
| *Hosts* | Отображается ссылка на список прототипов узлов сети.<br>Количество существующих прототипов узлов сети отображается серым цветом. |
| *Discovery rules* | Отображается ссылка на список прототипов обнаружения.<br>Количество существующих прототипов обнаружения отображается серым цветом. |
| *Key* | Отображается ключ элемента данных, используемый для обнаружения. |
| *Interval* | Отображается частота выполнения обнаружения.<br>*Примечание*: обнаружение также может быть выполнено немедленно нажатием кнопки *Execute now* под списком. |
| *Type* | Отображается тип элемента данных, используемый для обнаружения (агент Zabbix, SNMP agent и т. д.). |
| *Create enabled* | Отображается статус создания прототипа обнаружения - создание включено (*Yes*) или создание отключено (*No*). Щелчком по статусу можно изменить его - с Yes на No (и обратно). |
| *Discover* | Отображается статус обнаружения прототипа обнаружения - обнаруживать (*Yes*) или не обнаруживать (*No*). Щелчком по статусу можно изменить его - с Yes на No (и обратно). |

Чтобы настроить новый прототип обнаружения, нажмите кнопку *Create discovery prototype* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Создать включенными* - создать эти прототипы обнаружения как *Включенные*
- *Создать отключенными* - создать эти прототипы обнаружения как *Отключенные*
- *Удалить* - удалить эти прототипы обнаружения

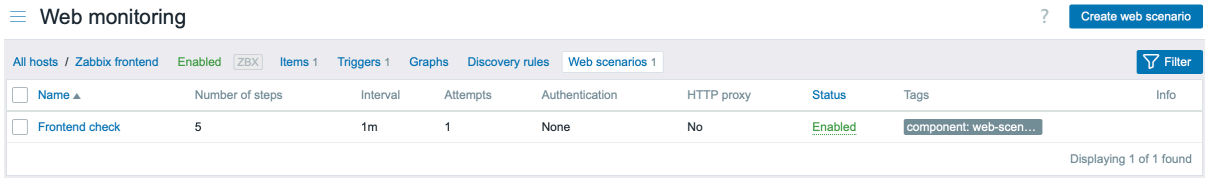
Чтобы использовать эти параметры, отметьте флажки перед соответствующими прототипами обнаружения, затем нажмите нужную кнопку.

5 Веб-сценарии

Обзор

Список веб-сценариев для узла сети можно открыть из раздела *Сбор данных* → *Узлы сети*, нажав *Web* для соответствующего узла сети.

Отображается список существующих веб-сценариев.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя веб-сценария. При нажатии на имя веб-сценария открывается форма настройки веб-сценария.<br>Если веб-сценарий узла сети принадлежит шаблону, перед именем веб-сценария серой ссылкой отображается имя шаблона. При нажатии на ссылку шаблона откроется список веб-сценариев на уровне шаблона. |
| *Number of steps* | Количество шагов, которые содержит сценарий. |
| *Update interval* | Как часто выполняется сценарий. |
| *Attempts* | Сколько попыток выполнения шагов веб-сценария выполняется. |
| *Authentication* | Отображается метод аутентификации - Basic, NTLM, Kerberos, Digest или None. |
| *HTTP proxy* | Отображается HTTP proxy или 'No', если он не используется. |
| *Status* | Отображается статус веб-сценария - *Enabled* или *Disabled*.<br>Нажав на статус, вы можете изменить его. |
| *Tags* | Отображаются теги веб-сценария.<br>Можно отобразить до трех тегов (пар имя:значение). Если тегов больше, отображается ссылка "..." , позволяющая увидеть все теги при наведении курсора. |
| *Info* | Если все работает корректно, в этом столбце не отображается значок. В случае ошибок отображается квадратный значок с буквой "i". Наведите курсор на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |

Чтобы настроить новый веб-сценарий, нажмите кнопку *Create web scenario* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

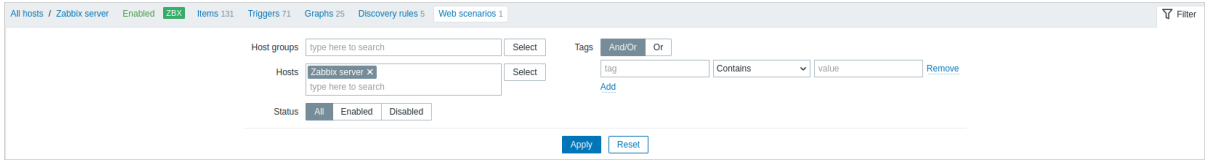
- *Включить* - изменить статус сценария на *Включен*
- *Отключить* - изменить статус сценария на *Отключен*
- *Очистить историю и тренды* - очистить данные истории и трендов для сценариев
- *Удалить* - удалить веб-сценарии

Чтобы использовать эти параметры, отметьте флажки перед соответствующими веб-сценариями, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те сценарии, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком веб-сценариев. При нажатии на нее откроется фильтр, где можно указать нужные критерии фильтрации.




| Parameter | Description |
|---|---|
| *Host groups* | Фильтрация по одной или нескольким группам узлов сети.<br>Указание родительской группы узлов сети неявно выбирает все вложенные группы узлов сети. |
| *Hosts* | Фильтрация по одному или нескольким узлам сети.<br>Если выше уже выбраны группы узлов сети, выбор узлов сети ограничивается этими группами. |
| *Status* | Фильтрация по статусу веб-сценария. |

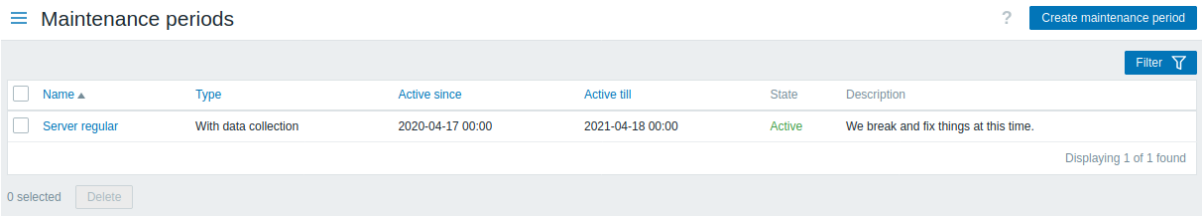
| Parameter | Description |
|---|---|
| *Tags* | Фильтрация по имени и значению тега веб-сценария.<br>Можно включать, а также исключать определенные теги и значения тегов. Можно задать несколько условий. Сопоставление имени тега всегда чувствительно к регистру.<br><br>Для каждого условия доступны следующие операторы:<br>**Exists** - включить указанные имена тегов;<br>**Equals** - включить указанные имена тегов и значения (с учетом регистра);<br>**Contains** - включить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра);<br>**Does not exist** - исключить указанные имена тегов;<br>**Does not equal** - исключить указанные имена тегов и значения (с учетом регистра);<br>**Does not contain** - исключить указанные имена тегов, у которых значения тегов содержат введенную строку (поиск подстроки, без учета регистра).<br><br>Для условий доступны два типа вычисления:<br>**And/Or** - должны быть выполнены все условия, условия с одинаковым именем тега будут сгруппированы по условию Or;<br>**Or** - достаточно выполнения одного условия. |

**5 Обслуживание**

Обзор

В разделе *Сбор данных* → *Обслуживание* пользователи могут настраивать и поддерживать периоды обслуживания для узлов сети.

Отображается список существующих периодов обслуживания с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя периода обслуживания. При нажатии на имя периода обслуживания открывается форма настройки периода обслуживания. |
| *Type* | Отображается тип обслуживания: *With data collection* или *No data collection* |
| *Active since* | Дата и время, с которого выполнение периодов обслуживания становится активным.<br>Примечание: это время не активирует период обслуживания; периоды обслуживания нужно задавать отдельно. |
| *Active till* | Дата и время, после которого выполнение периодов обслуживания перестает быть активным. |
| *State* | Состояние периода обслуживания:<br>**Approaching** - скоро станет активным<br>**Active** - активен<br>**Expired** - больше не активен |
| *Description* | Отображается описание периода обслуживания. |

Чтобы настроить новый период обслуживания, нажмите кнопку *Create maintenance period* в правом верхнем углу.

Параметры массового редактирования

Кнопка под списком предлагает один вариант массового редактирования:

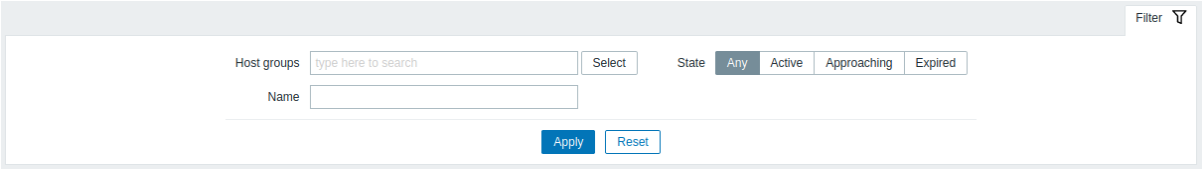
- *Удалить* - удалить периоды обслуживания

Чтобы использовать этот вариант, отметьте флажки перед соответствующими периодами обслуживания и нажмите *Удалить*.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те периоды обслуживания, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком периодов обслуживания. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно отфильтровать периоды обслуживания по группе узлов сети, имени и состоянию.



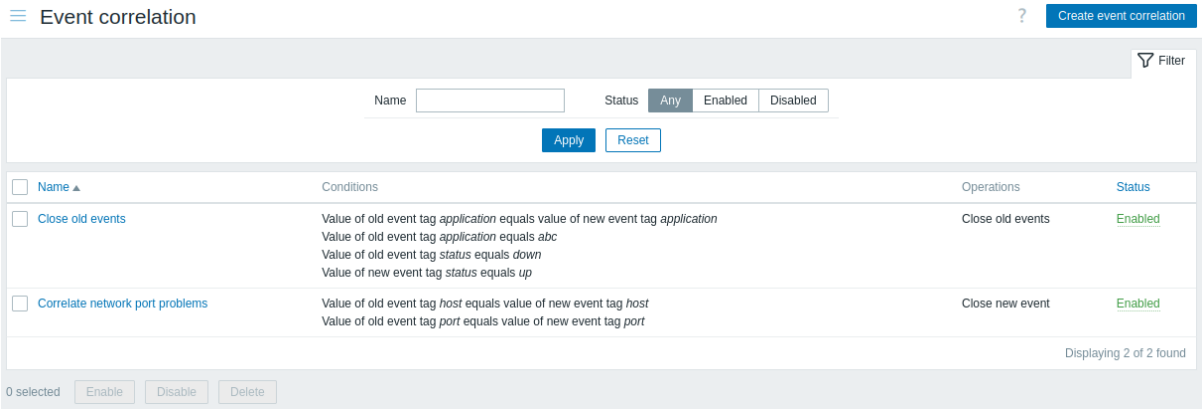

Расчет очередей во время обслуживания

**Attention:**
Прокси Zabbix не знает о периодах обслуживания; подробности см. в разделе Расчет очередей во время обслуживания.

## 6 Корреляция событий

Обзор

В разделе *Сбор данных* → *Корреляция событий* пользователи могут настраивать и поддерживать правила глобальной корреляции для событий Zabbix.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя правила корреляции. При нажатии на имя правила корреляции открывается форма настройки. |
| *Conditions* | Отображаются условия правила корреляции. |
| *Operations* | Отображаются операции правила корреляции. |
| *Status* | Отображается статус правила корреляции - *Enabled* или *Disabled*.<br>Нажав на статус, вы можете изменить его. |

Чтобы настроить новое правило корреляции, нажмите кнопку *Create event correlation* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

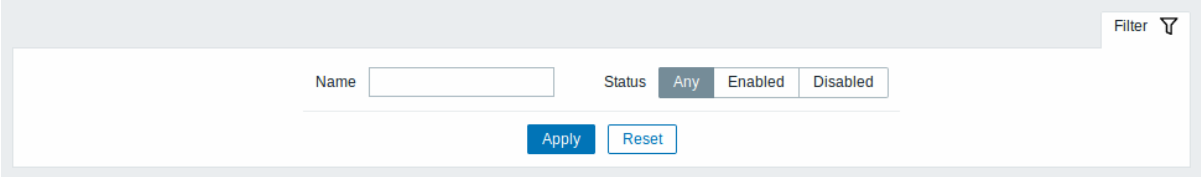
- *Включить* - изменить статус правила корреляции на *Включено*
- *Отключить* - изменить статус правила корреляции на *Отключено*
- *Удалить* - удалить правила корреляции

Чтобы использовать эти параметры, отметьте флажки перед соответствующими правилами корреляции, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те правила корреляции, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком правил корреляции. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно отфильтровать правила корреляции по имени и статусу.

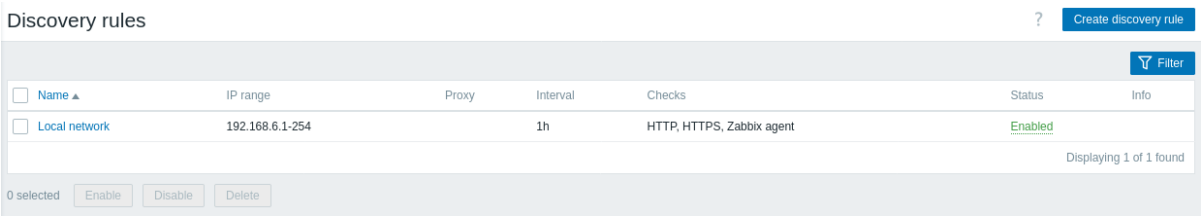



## 7 Обнаружение

Обзор

В разделе *Сбор данных* → *Обнаружение* пользователи могут настраивать и поддерживать правила обнаружения.

Отображается список существующих правил обнаружения с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя правила обнаружения. При нажатии на имя правила обнаружения открывается форма настройки. |
| *IP range* | Отображается диапазон IP-адресов, используемый для сканирования сети. |
| *Proxy* | Отображается имя прокси, если обнаружение выполняется через прокси. |
| *Interval* | Отображается частота выполнения обнаружения. |
| *Checks* | Отображаются типы проверок, используемых для обнаружения. |
| *Status* | Отображается статус правила обнаружения - *Enabled* или *Disabled*.<br>Нажав на статус, вы можете изменить его. |
| *Info* | Если все работает корректно, в этом столбце ничего не отображается. В случае ошибок отображается красный значок информации с буквой ”i”. Наведите указатель на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |

Чтобы настроить новое правило обнаружения, нажмите кнопку *Create discovery rule* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

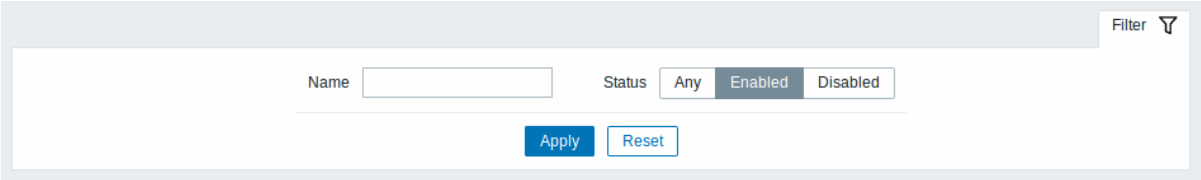
- *Включить* - изменить статус правила обнаружения на *Включено*
- *Отключить* - изменить статус правила обнаружения на *Отключено*
- *Удалить* - удалить правила обнаружения

Чтобы использовать эти параметры, отметьте флажки перед соответствующими правилами обнаружения, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те правила обнаружения, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком правил обнаружения. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать правила обнаружения по имени и статусу.




## 7 Оповещения

Обзор

В этом меню есть разделы, связанные с настройкой оповещений в Zabbix.

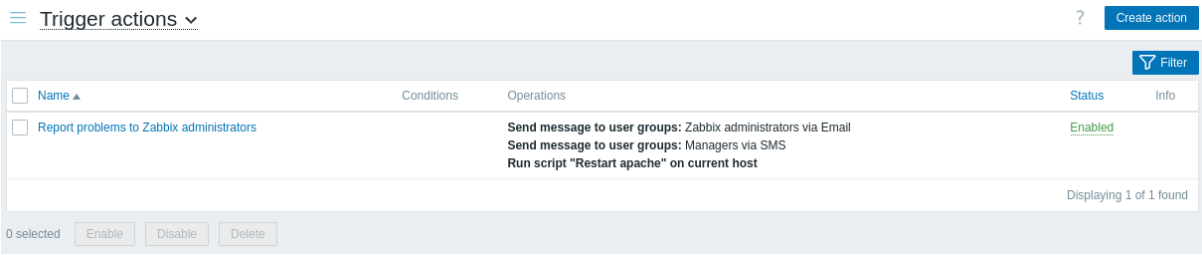
### 1 Действия

Обзор

В разделе *Alerts* → *Actions* пользователи могут настраивать и обслуживать действия.

Отображаемые действия — это действия, назначенные выбранному источнику событий (триггер, services, discovery, autoregistration, internal actions).

Чтобы переключиться на другой источник событий, нажмите *Actions* в разделе меню *Alerts*. Также можно переключаться между источниками с помощью выпадающего списка в заголовке в левом верхнем углу.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя действия. При нажатии на имя действия открывается форма настройки. |
| *Conditions* | Отображаются условия действия. |
| *Operations* | Отображаются операции действия.<br>Список операций также показывает тип медиа (email, SMS или script), используемый для уведомления, а также имя и фамилию (в скобках после имени пользователя) получателя уведомления.<br>Операция действия может быть как уведомлением, так и удаленной командой в зависимости от выбранного типа операции. |
| *Status* | Отображается статус действия — *Enabled* или *Disabled*.<br>Нажав на статус, вы можете изменить его.<br>См. раздел Escalations для получения дополнительных сведений о том, что происходит, если действие отключено во время выполняющейся эскалации. |
| *Info* | Если все работает корректно, в этом столбце не отображается значок. В случае ошибок, например отсутствующих операций действия или условий после обновления/обнаружения на низком уровне, отображается значок предупреждения. Наведите указатель на значок, чтобы увидеть всплывающую подсказку с описанием ошибки. |

Чтобы настроить новое действие, нажмите кнопку *Create action* в правом верхнем углу.

Для пользователей без прав Super admin действия отображаются в соответствии с настройками разрешений. Это означает, что в некоторых случаях пользователь без прав Super admin может не видеть полный список действий из-за определенных ограничений разрешений. Действие отображается пользователю без прав Super admin, если выполняются следующие условия:

- Пользователь имеет доступ на чтение и запись к группам узлов сети, узлам сети, шаблонам и триггерам в условиях действия
- Пользователь имеет доступ на чтение и запись к группам узлов сети, узлам сети и шаблонам в операциях действия, операциях восстановления и операциях обновления

- Пользователь имеет доступ на чтение к группам пользователей и пользователям в операциях действия, операциях восстановления и операциях обновления

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

- *Включить* - изменить статус действия на *Включено*
- *Отключить* - изменить статус действия на *Отключено*
- *Удалить* - удалить действия

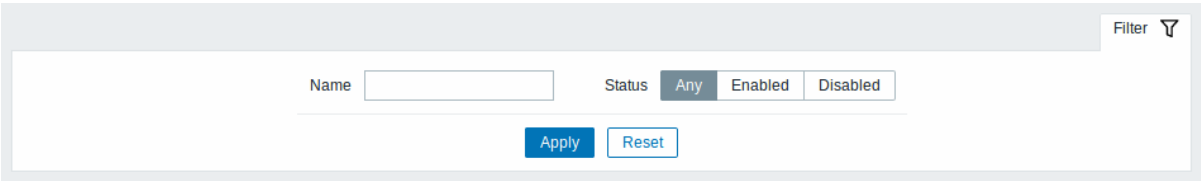
Чтобы использовать эти параметры, отметьте флажки перед соответствующими действиями, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те действия, которые вас интересуют.

Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком действий. Если вы нажмете на нее, станет доступен фильтр, с помощью которого можно фильтровать действия по имени и статусу.




**2 Типы медиа**

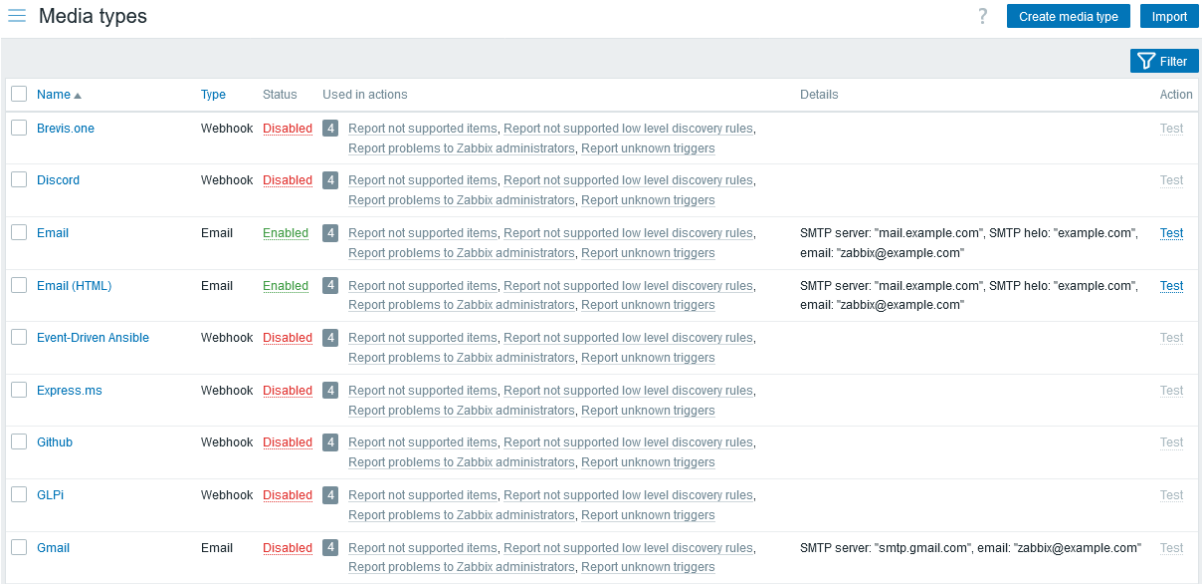
Обзор

В разделе *Оповещения* → *Типы оповещений* пользователи могут настраивать и поддерживать информацию о типах оповещений.

Это меню доступно только пользователям типа SuperAdmin.

Информация о типе оповещения содержит общие инструкции по использованию канала доставки уведомлений. Специфические сведения, такие как отдельные адреса электронной почты для отправки уведомлений, хранятся для каждого пользователя отдельно.

Отображается список существующих типов оповещений с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя типа оповещения. При нажатии на имя открывается форма настройки типа оповещения. |

| Column | Description |
|---|---|
| *Type* | Отображается тип канала оповещения (email, SMS и т. д.). |
| *Status* | Отображается статус типа оповещения - *Enabled* или *Disabled*.<br>Нажав на статус, вы можете изменить его. |
| *Used in actions* | Отображаются действия, в которых используется тип оповещения, с указанием общего числа этих действий.<br>При нажатии на имя действия открывается форма настройки действия. Если у пользователя нет прав на действие, имя не является ссылкой. |
| *Details* | Отображается подробная информация о типе оповещения. |
| *Actions* | Доступно следующее действие:<br>**Test** - нажмите, чтобы открыть форму тестирования, где можно указать параметры типа оповещения (например, адрес получателя, тему и текст тестового сообщения) и отправить тестовое сообщение, чтобы проверить, что настроенный тип оповещения работает. См. также: тестирование типа оповещения для Email, Webhook или Script. |

Чтобы настроить новый тип оповещения, нажмите кнопку *Create media type* в правом верхнем углу.

Чтобы импортировать тип оповещения, нажмите кнопку *Import* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

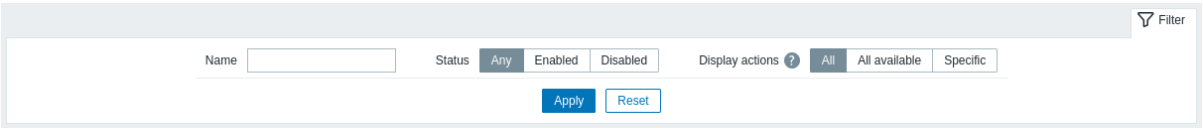
- *Включить* - изменить статус типа оповещений на *Включено*
- *Отключить* - изменить статус типа оповещений на *Отключено*
- *Экспорт* - экспортировать типы оповещений в файл YAML, XML или JSON
- *Удалить* - удалить типы оповещений

Чтобы использовать эти параметры, отметьте флажки перед соответствующими типами оповещений, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те типы медиа, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком типов медиа. Если щелкнуть ее, станет доступен фильтр, с помощью которого можно фильтровать типы медиа по имени и статусу. Кроме того, вы можете использовать фильтр для отображения действий в столбце *Used in actions* в зависимости от области использования их типа медиа (определяется параметром *Send to media type* в деталях операции действия).




## 3 Скрипты

Обзор

В разделе *Alerts* > *Scripts* можно настраивать и поддерживать пользовательские глобальные скрипты.

Это меню доступно только пользователям типа SuperAdmin.

Глобальные скрипты, в зависимости от настроенной области действия, а также прав пользователя, доступны для выполнения:

- из меню узла сети в различных разделах веб-интерфейса (*Dashboard*, *Problems*, *Latest data*, *Maps* и т. д.)
- из меню события
- могут запускаться как операция действия

Скрипты выполняются на агенте Zabbix, сервере Zabbix (прокси) или только на сервере Zabbix См. также Command execution.

И на агенте Zabbix, и на прокси Zabbix удаленные скрипты по умолчанию отключены. Их можно включить следующим образом:

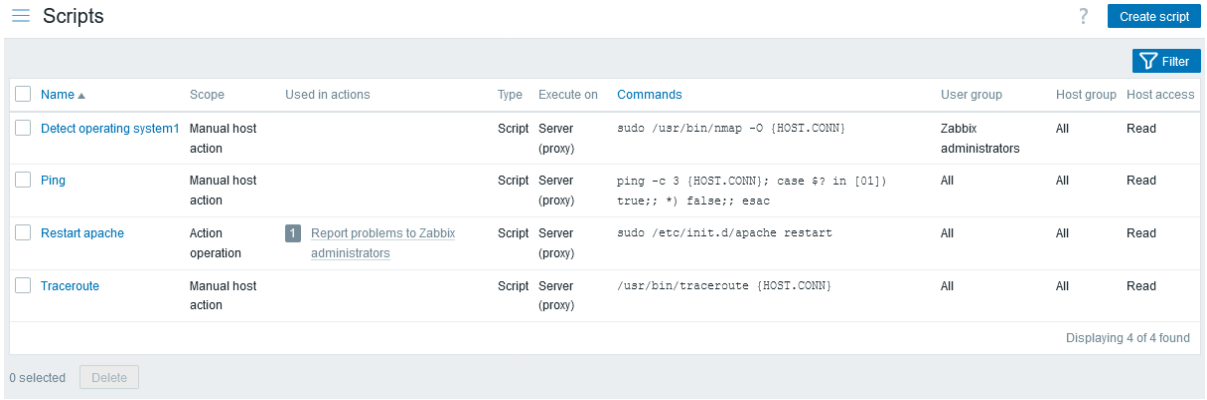
- Для удаленных команд, выполняемых на агенте Zabbix:

- добавьте параметр AllowKey=system.run[<command>,*] для каждой разрешенной команды в конфигурации агента, * означает режим wait и nowait;

- Для удаленных команд, выполняемых на прокси Zabbix:
  - **Предупреждение: включать удаленные команды на прокси Zabbix не требуется, если удаленные команды выполняются на агенте Zabbix, который контролируется прокси Zabbix.** Однако, если необходимо выполнять удаленные команды на прокси Zabbix, установите параметр *EnableRemoteCommands* в значение '1' в конфигурации прокси.

Выполнение глобальных скриптов на сервере Zabbix можно отключить, установив `EnableGlobalScripts=0` в конфигурации сервера. Для новых установок, начиная с Zabbix 7.0, выполнение глобальных скриптов на сервере Zabbix по умолчанию отключено.

Отображается список существующих скриптов с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя скрипта. Щелчок по имени скрипта открывает форму настройки скрипта. |
| *Scope* | Область действия скрипта - операция действия, ручное действие для узла сети или ручное действие для события. Этот параметр определяет, где доступен скрипт. |
| *Used in actions* | Отображаются все действия, в которых используется скрипт, с указанием общего числа этих действий.<br>Щелчок по имени действия открывает форму настройки действия. Если у пользователя нет прав на действие, имя не является ссылкой. |
| *Type* | Отображается тип скрипта - *URL*, *Webhook*, *Script*, *SSH*, *Telnet* или команда *IPMI*. |
| *Execute on* | Показывает, будет ли скрипт выполняться на агенте Zabbix, прокси Zabbix или сервере, либо только на сервере Zabbix. |
| *Commands* | Отображаются все команды, которые будут выполнены в рамках скрипта.<br>Для вебхуков здесь ничего не отображается. |
| *User group* | Отображается группа пользователей, которой доступен скрипт (или *All* для всех групп пользователей). |
| *Host group* | Отображается группа узлов сети, для которой доступен скрипт (или *All* для всех групп узлов сети). |
| *Host access* | Отображается уровень прав для группы узлов сети - *Read* или *Write*. Доступ к выполнению скрипта будут иметь только пользователи с требуемым уровнем прав. |

Чтобы настроить новый скрипт, нажмите кнопку *Create script* в правом верхнем углу.

Параметры массового редактирования

Кнопка под списком предлагает один вариант массового редактирования:

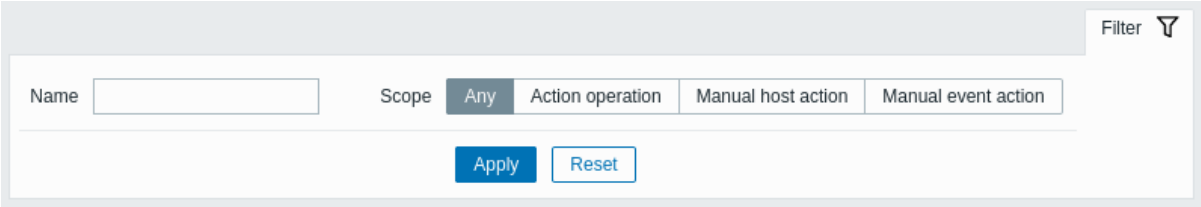
- *Delete* - удалить скрипты

Чтобы использовать этот вариант, отметьте флажки перед соответствующими скриптами и нажмите *Delete*.

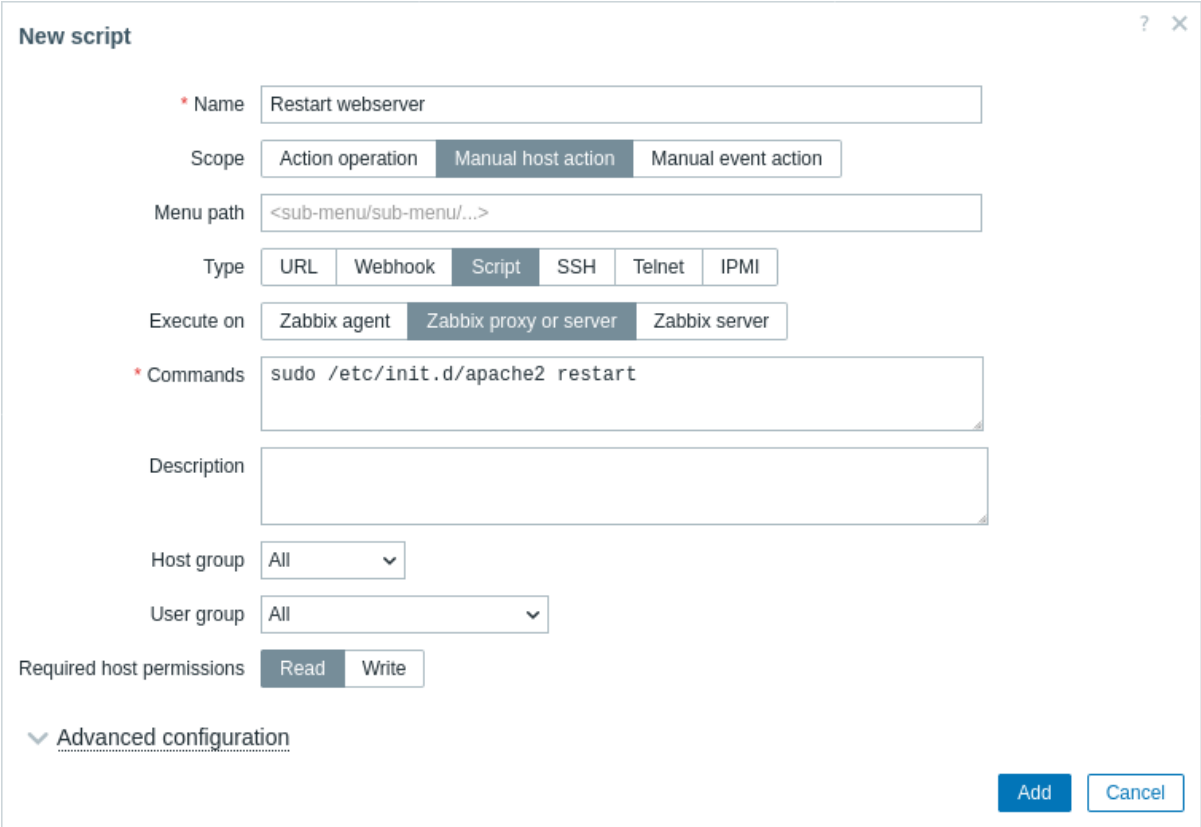
Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те скрипты, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком скриптов. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать скрипты по имени и области применения.




Настройка глобального скрипта




**Note:**
Рекомендуется использовать функции макросов вместо обычных макросов в глобальных скриптах для повышения безопасности, так как макросы не экранируются автоматически.

Атрибуты скрипта:

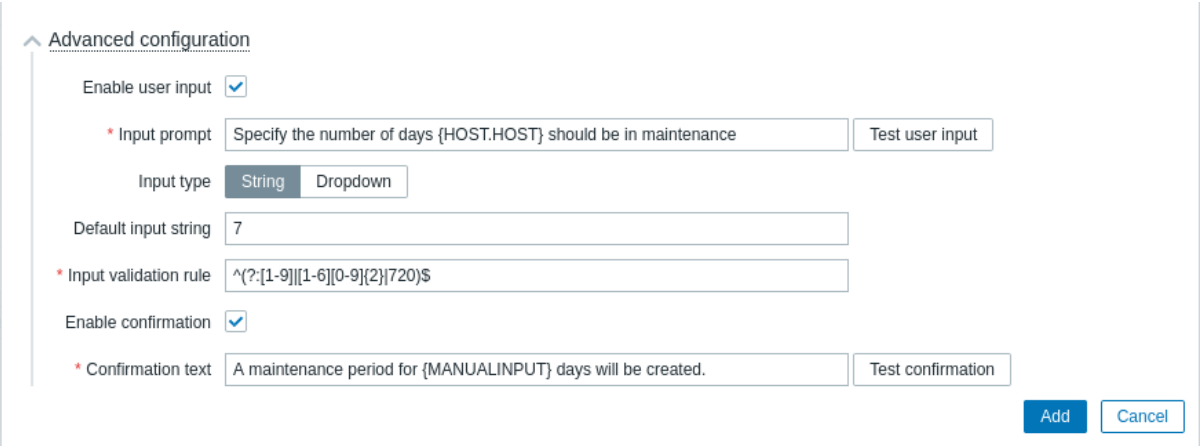
| Параметр | Описание |
|---|---|
| *Name* | Уникальное имя скрипта.<br>Например, `Clear /tmp filesystem` |

| Параметр | Описание |
|---|---|
| *Scope* | Область применения скрипта - операция действия, ручное действие узла сети или ручное действие события. Этот параметр определяет, где можно использовать скрипт - в удаленных командах операций действия, из меню узла сети или из меню события соответственно.<br>Если для области применения указано 'Action operation', скрипт будет доступен всем пользователям с доступом к *Alerts* > *Actions*.<br>Если скрипт фактически используется в действии, изменить его область применения с 'action operation' нельзя.<br>**Поддержка макросов**<br>Область применения влияет на набор доступных макросов. Например, пользовательские макросы ({USER.*}) поддерживаются в скриптах, чтобы можно было передавать информацию о пользователе, запустившем скрипт. Однако они не поддерживаются, если область применения скрипта - action operation, так как операции действия выполняются автоматически.<br>Макрос {MANUALINPUT} позволяет указать ввод вручную во время выполнения скрипта. Он поддерживается для скриптов ручного действия узла сети и ручного действия события.<br>Чтобы узнать, какие еще макросы поддерживаются, выполните поиск по 'Trigger-based notifications and commands/Trigger-based commands', 'Manual host action scripts' и 'Manual event action scripts' в таблице поддерживаемых макросов. Обратите внимание: если макрос может быть преобразован в значение с пробелами (например, имя узла сети), не забудьте при необходимости заключить его в кавычки. |
| *Menu path* | Требуемый путь меню к скрипту. Например, `Default` или `Default/` отобразит скрипт в соответствующем каталоге. Меню могут быть вложенными, например `Main menu/Sub menu1/Sub menu2`. При доступе к скриптам через меню узла сети/события в разделах мониторинга они будут организованы в соответствии с указанными каталогами.<br>Это поле отображается только если в качестве *Scope* выбрано 'Manual host action' или 'Manual event action'. |
| *Type* | Нажмите соответствующую кнопку, чтобы выбрать тип скрипта: **URL**, **Webhook**, **Script**, **SSH**, **Telnet** или команду **IPMI**.<br>Тип **URL** доступен только если в качестве *Scope* выбрано 'Manual host action' или 'Manual event action'. |
| Тип скрипта: **URL** | |
| *URL* | Укажите URL для быстрого доступа из меню узла сети или меню события.<br>Поддерживаются макросы и пользовательские макросы пользователя. Поддержка макросов зависит от области применения скрипта (см. *Scope* выше).<br>Используйте макрос {MANUALINPUT} в этом поле, чтобы можно было указать ввод вручную во время выполнения скрипта, например:<br>`http://{MANUALINPUT}/zabbix/zabbix.php?action=dashboard.view`<br>Значения макросов не должны быть URL-кодированы. |
| *Open in new window* | Определяет, должен ли URL открываться в новой вкладке браузера или в той же. |
| Тип скрипта: **Webhook** | |
| *Parameters* | Укажите переменные вебхука в виде пар атрибут-значение.<br>См. также конфигурацию медиа-типа Webhook.<br>Поддерживаются макросы и пользовательские макросы пользователя в значениях параметров. Поддержка макросов зависит от области применения скрипта (см. *Scope* выше). |
| *Script* | Введите код JavaScript в модальном редакторе, который открывается при щелчке в поле параметра или по значку карандаша рядом с ним.<br>Поддержка макросов зависит от области применения скрипта (см. *Scope* выше).<br>См. также конфигурацию медиа-типа Webhook, Additional Javascript objects. |
| *Timeout* | Тайм-аут выполнения JavaScript (1-60 с, по умолчанию 30 с).<br>Поддерживаются суффиксы времени, например 30s, 1m. |
| Тип скрипта: **Script** | |

| Параметр | Описание |
|---|---|
| *Execute on* | Нажмите соответствующую кнопку, чтобы выполнить shell-скрипт на:<br>**Zabbix agent** - скрипт будет выполнен Zabbix agent (если элемент данных system.run разрешен) на узле сети<br>**Zabbix proxy or server** - скрипт будет выполнен Zabbix proxy или server - в зависимости от того, контролируется ли узел сети через proxy или server.<br>Он будет выполнен на proxy, если это разрешено параметром EnableRemoteCommands.<br>Он будет выполнен на server, если глобальные скрипты включены параметром сервера EnableGlobalScripts.<br>**Zabbix server** - скрипт будет выполнен только Zabbix server.<br>Этот параметр будет недоступен, если глобальные скрипты отключены параметром сервера EnableGlobalScripts. |
| *Commands* | Введите полный путь к командам, которые будут выполняться в скрипте.<br>Поддержка макросов зависит от области применения скрипта (см. *Scope* выше).<br>Поддерживаются пользовательские макросы пользователя. |
| Тип скрипта: **SSH** | |
| *Authentication method* | Выберите метод аутентификации - пароль или публичный ключ. |
| *Username* | Введите имя пользователя. |
| *Password* | Введите пароль.<br>Это поле доступно, если в качестве метода аутентификации выбран 'Password'. |
| *Public key file* | Введите путь к файлу публичного ключа.<br>Это поле доступно, если в качестве метода аутентификации выбран 'Public key'. |
| *Private key file* | Введите путь к файлу закрытого ключа.<br>Это поле доступно, если в качестве метода аутентификации выбран 'Public key'. |
| *Passphrase* | Введите парольную фразу.<br>Это поле доступно, если в качестве метода аутентификации выбран 'Public key'. |
| *Port* | Введите порт удаленной службы SSH на целевом узле сети, к которому Zabbix будет подключаться. |
| *Commands* | Введите команды.<br>Поддержка макросов зависит от области применения скрипта (см. *Scope* выше).<br>Поддерживаются пользовательские макросы пользователя. |
| Тип скрипта: **Telnet** | |
| *Username* | Введите имя пользователя. |
| *Password* | Введите пароль. |
| *Port* | Введите порт удаленной службы Telnet на целевом узле сети, к которому Zabbix будет подключаться. |
| *Commands* | Введите команды.<br>Поддержка макросов зависит от области применения скрипта (см. *Scope* выше).<br>Поддерживаются пользовательские макросы пользователя. |
| Тип скрипта: **IPMI** | |
| *Command* | Введите команду IPMI.<br>Поддержка макросов зависит от области применения скрипта (см. *Scope* выше).<br>Поддерживаются пользовательские макросы пользователя. |
| *Description* | Введите описание скрипта. |
| *Host group* | Выберите группу узлов сети, для которой будет доступен скрипт (или *All* для всех групп узлов сети). |
| *User group* | Выберите группу пользователей, которой будет доступен скрипт (или *All* для всех групп пользователей).<br>Это поле отображается только если в качестве *Scope* выбрано 'Manual host action' или 'Manual event action'. |
| *Required host permissions* | Выберите уровень прав для группы узлов сети - *Read* или *Write*. Доступ к выполнению скрипта будут иметь только пользователи с требуемым уровнем прав.<br>Это поле отображается только если в качестве *Scope* выбрано 'Manual host action' или 'Manual event action'. |
| *Advanced configuration* | Нажмите заголовок *Advanced configuration*, чтобы отобразить параметры расширенной настройки.<br>Это поле отображается только если в качестве *Scope* выбрано 'Manual host action' или 'Manual event action'. |

Расширенная настройка

Параметры расширенной настройки доступны в сворачиваемом разделе *Расширенная настройка*:




| Parameter | Description |
|---|---|
| *Enable user input* | Установите флажок, чтобы включить ручной ввод пользователя перед выполнением скрипта.<br>Ручной ввод пользователя заменит значение макроса {MANUALINPUT} в скрипте.<br>См. также: Ручной ввод пользователя. |
| *Input prompt* | Введите собственный текст-подсказку для пользовательского ввода. Этот текст будет отображаться над полем ввода во всплывающем окне *Manual input*.<br>Чтобы увидеть предварительный просмотр всплывающего окна *Manual input*, нажмите *Test user input*. Предварительный просмотр также позволяет проверить, соответствует ли строка ввода правилу проверки ввода (см. параметры ниже).<br>Поддержка макросов и пользовательских макросов зависит от области действия скрипта (см. *Scope* в общих параметрах настройки скрипта). |
| *Input type* | Выберите тип ручного ввода:<br>**String** - одна строка;<br>**Dropdown** - значение выбирается из нескольких вариантов в раскрывающемся списке. |
| *Dropdown options* | Введите уникальные значения для раскрывающегося списка пользовательского ввода в виде списка, разделенного запятыми.<br>Чтобы включить пустой вариант в раскрывающийся список, добавьте дополнительную запятую в начале, середине или конце списка.<br>Это поле отображается только если в качестве *Input type* выбран вариант 'Dropdown'. |
| *Default input string* | Введите строку по умолчанию для пользовательского ввода (или не указывайте ничего).<br>Это поле будет проверяться по регулярному выражению, указанному в поле *Input validation rule*.<br>Введенное здесь значение будет отображаться по умолчанию во всплывающем окне *Manual input*.<br>Это поле отображается только если в качестве *Input type* выбран вариант 'String'. |
| *Input validation rule* | Введите регулярное выражение для проверки строки пользовательского ввода.<br>Поддерживаются глобальные регулярные выражения.<br>Это поле отображается только если в качестве *Input type* выбран вариант 'String'. |
| *Enable confirmation* | Установите флажок, чтобы отображать сообщение с подтверждением перед выполнением скрипта. Эта функция может быть особенно полезна для потенциально опасных операций (например, скрипта перезагрузки) или операций, которые могут занять много времени. |
| *Confirmation text* | Введите собственный текст подтверждения для всплывающего окна подтверждения, включаемого флажком выше (например, *Remote system will be rebooted. Are you sure?*).<br>Чтобы увидеть, как будет выглядеть текст, нажмите *Test confirmation* рядом с полем.<br>Поддерживаются макросы и пользовательские user macros.<br>*Примечание:* макросы не будут раскрыты при проверке сообщения подтверждения. |

Если настроены и ручной ввод пользователя, и сообщение подтверждения, они будут отображаться в последовательных всплывающих окнах.

Ручной ввод пользователем

Ручной ввод пользователем позволяет задавать пользовательский параметр при каждом выполнении скрипта. Это избавляет от необходимости создавать несколько похожих пользовательских скриптов, отличающихся только одним

параметром.

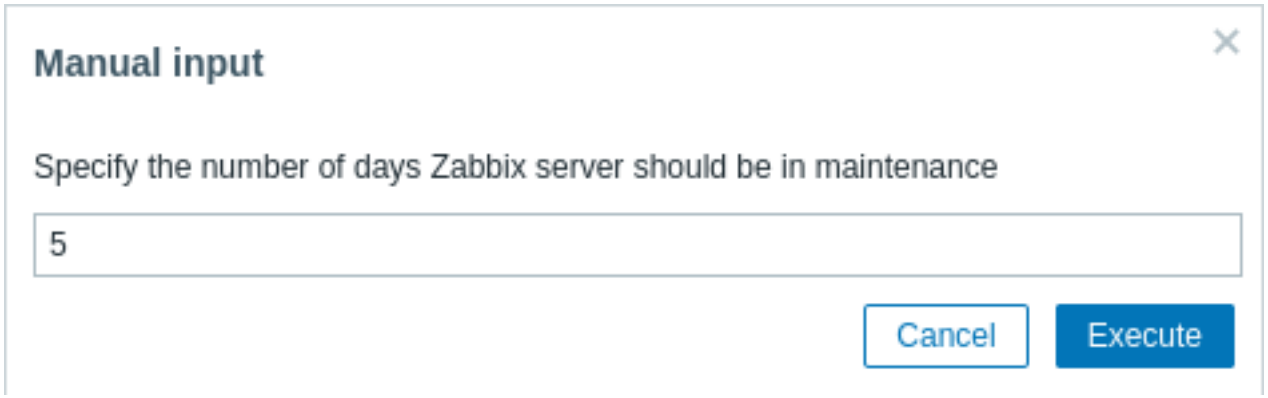
Например, во время выполнения скрипта вы можете передать ему другое целое число или другой URL-адрес.

Чтобы включить ручной ввод пользователем:

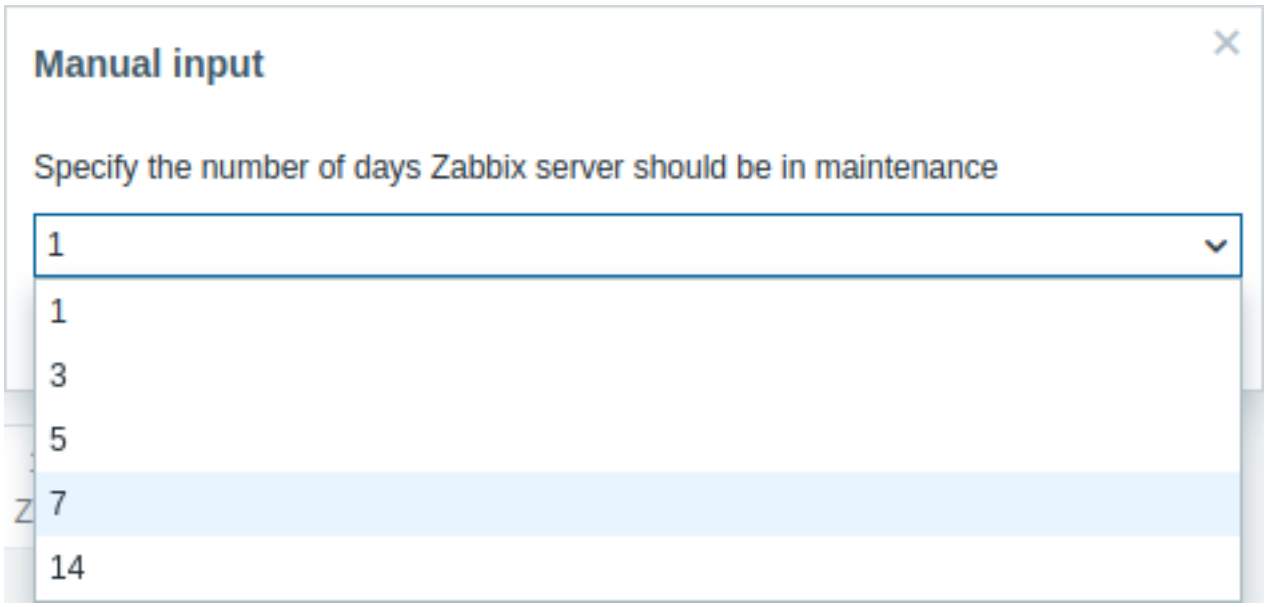
- используйте макрос {MANUALINPUT} в скрипте (команды, скрипт, параметр скрипта), где это требуется; или в поле URL для URL-скриптов;
- в расширенной конфигурации скрипта включите ручной ввод пользователем и настройте параметры ввода.

Если ввод пользователем включен, перед выполнением скрипта пользователю будет показано всплывающее окно *Ручной ввод*, в котором будет предложено указать пользовательское значение. Указанное значение заменит {MANUALINPUT} в скрипте.

В зависимости от конфигурации пользователю будет предложено ввести строковое значение:




Или выбрать значение из раскрывающегося списка заранее определенных вариантов:




Ручной ввод пользователем доступен только для скриптов, у которых область действия — 'Manual host action' или 'Manual event action'.

Выполнение скрипта и результат

Скрипты, запускаемые сервером Zabbix, выполняются в порядке, описанном на странице выполнения команд.

Результат скрипта отображается во всплывающем окне, которое появляется после выполнения скрипта. Возвращаемое значение скрипта — это стандартный вывод:

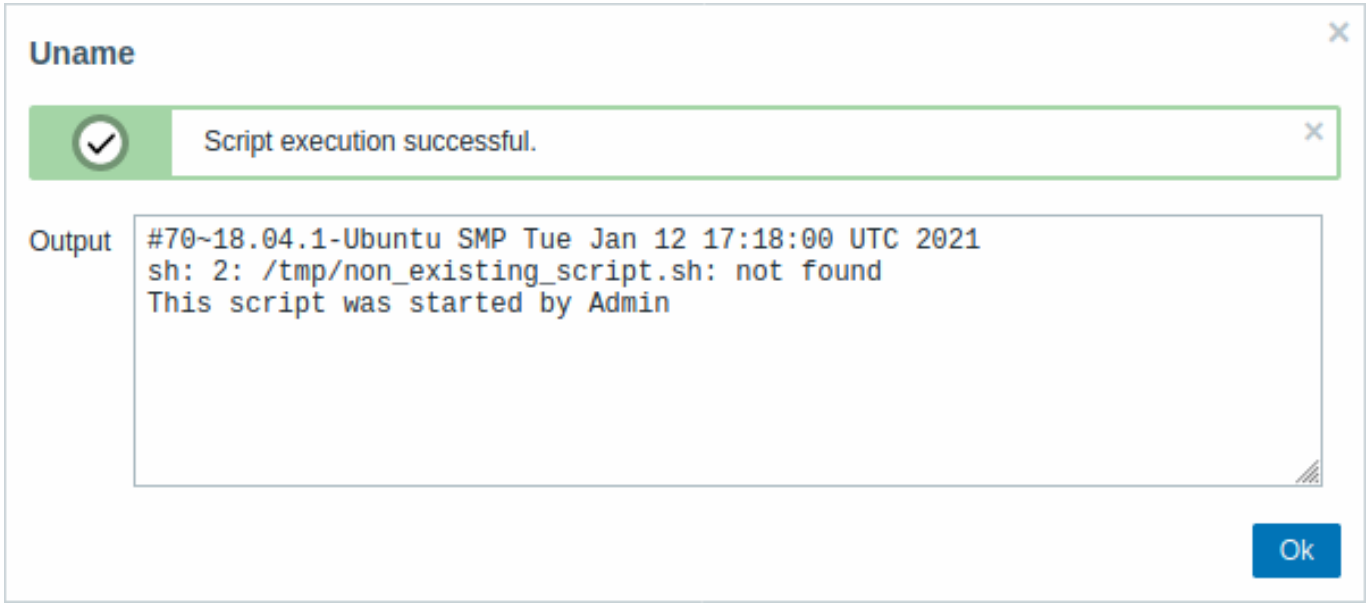
- Если скрипт завершается успешно (код завершения 0), возвращаемое значение ограничено 16 МБ (включая завершающие пробелы, которые обрезаются).
- Если скрипт завершается с ошибкой (ненулевой код завершения), возвращаемое значение стандартной ошибки ограничено 2 КБ.

По умолчанию Zabbix не сохраняет расширенный вывод скрипта. Чтобы сохранить полный вывод, можно реализовать ведение журнала непосредственно в самом скрипте (например, перенаправить вывод в локальный файл журнала).

Обратите внимание, что для скриптов, выполняемых либо на сервере Zabbix, либо на прокси Zabbix, также применяются ограничения базы данных.

Ниже приведен пример скрипта и окна результата:

```
uname -v
/tmp/non_existing_script.sh
echo "This script was started by {USER.USERNAME}"
```



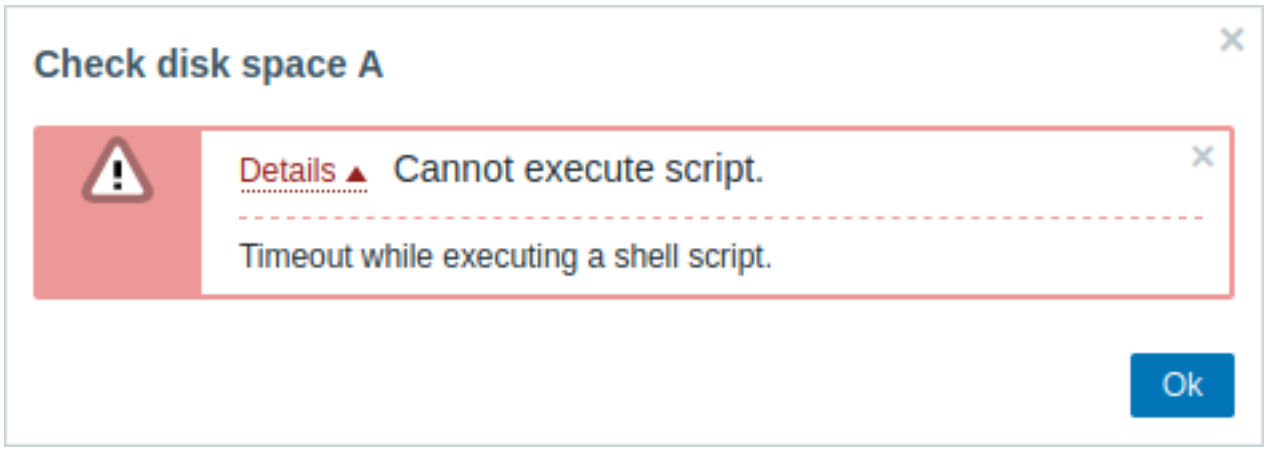

В результате выполнения скрипта сам скрипт не отображается.

Тайм-аут скрипта в агенте Zabbix

Вы можете столкнуться с ситуацией, когда при выполнении скрипта возникает тайм-аут.

Ниже приведен пример выполнения скрипта на агенте Zabbix и окно с результатом:

```
sleep 5
df -h
```




В этом случае сообщение об ошибке будет следующим:

```
Timeout while executing a shell script.
```

Чтобы избежать таких ситуаций, рекомендуется оптимизировать сам скрипт (в приведенном выше примере — ”5”), а не изменять параметр Timeout в конфигурации агента Zabbix и конфигурации сервера Zabbix.

Если параметр Timeout был изменен в конфигурации агента Zabbix, появится следующее сообщение об ошибке:

```
Get value from agent failed: ZBX_TCP_READ() timed out.
```

Это означает, что изменение было внесено в конфигурацию агента Zabbix, но также требуется изменить параметр Timeout в конфигурации сервера Zabbix.

Тайм-аут выполнения скрипта на активных агентах

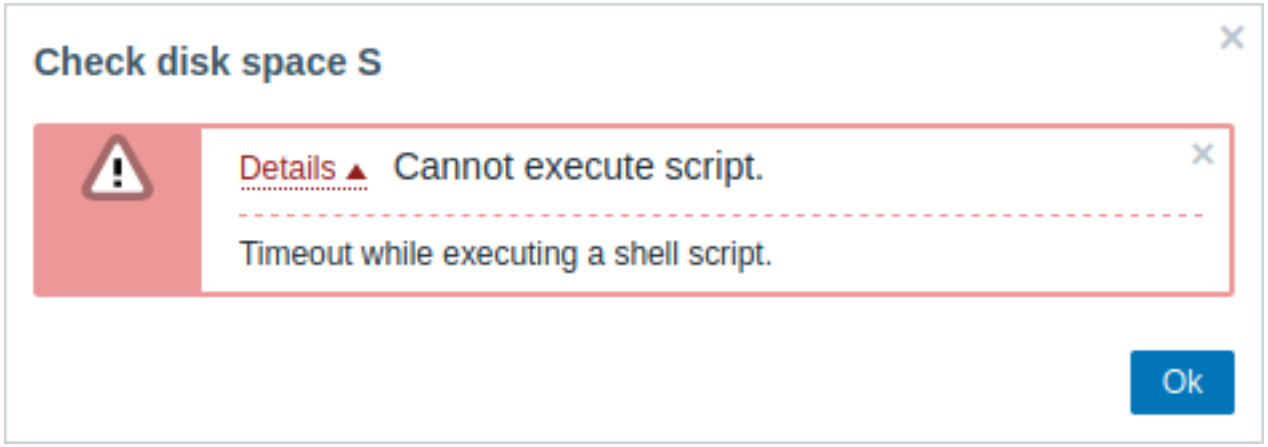
Выполнять скрипты можно только на агентах, работающих в активном режиме. После того как выполнение скрипта будет инициировано операцией действия или ручным запуском скрипта, команда включается в конфигурацию активных проверок и выполняется, когда активный агент ее получает.

Ручные скрипты отправляются активному агенту вместе с тайм-аутом сервера/прокси для выполнения скрипта. Увеличьте значение тайм-аута по умолчанию для выполнения скрипта на сервере/прокси. Тайм-аут должен быть больше, чем частота обновления активных проверок, иначе тайм-аут будет превышен до того, как активный агент получит скрипт и сможет вернуть результат.

Тайм-аут скрипта на сервере/прокси Zabbix

Ниже приведен пример выполнения скрипта на сервере Zabbix и окно с результатом:

```
sleep 11
df -h
```




Также рекомендуется оптимизировать сам скрипт (вместо настройки параметра `TrapperTimeout` на соответствующее значение (в нашем случае, > 11) путем изменения конфигурации сервера Zabbix).

**8 Пользователи**

Обзор

Это меню содержит разделы, связанные с настройкой пользователей в Zabbix. Это меню доступно только пользователям типа SuperAdmin.

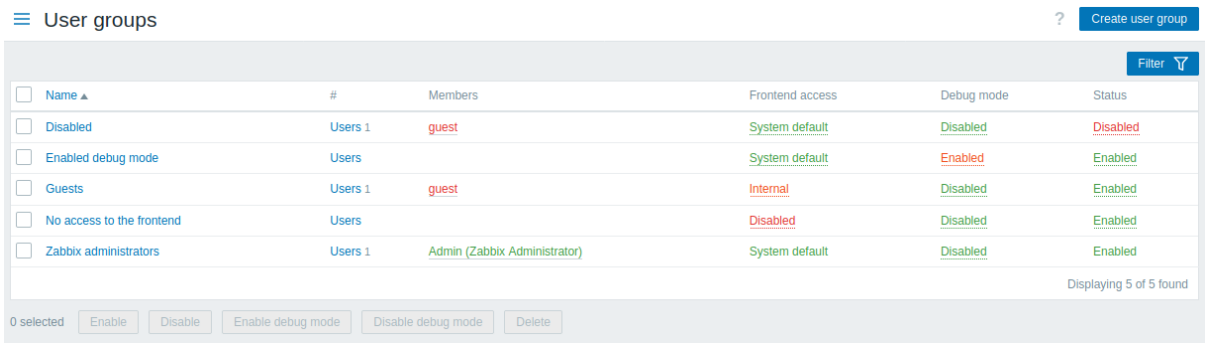
**1 Группы пользователей**

Обзор

В разделе *Пользователи → Группы пользователей* ведется учет групп пользователей.

Группы пользователей

Отображается список существующих групп пользователей с их подробными сведениями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя группы пользователей. При нажатии на имя группы пользователей открывается форма настройки группы пользователей. |
| *#* | Количество пользователей в группе. При нажатии на *Users* будут отображены соответствующие пользователи, отфильтрованные в списке пользователей. |
| *Members* | Имена пользователей отдельных пользователей в группе пользователей (с именем и фамилией в скобках). При нажатии на имя пользователя откроется форма настройки пользователя. Пользователи из отключенных групп отображаются красным цветом. |
| *Frontend access* | Отображается уровень доступа к веб-интерфейсу:<br>**System default** - пользователи проходят аутентификацию через Zabbix, LDAP или HTTP (в зависимости от глобально заданного метода аутентификации);<br>**Internal** - пользователи проходят аутентификацию через Zabbix; игнорируется, если HTTP-аутентификация задана как глобальное значение по умолчанию;<br>**LDAP** - пользователи проходят аутентификацию через LDAP; игнорируется, если HTTP-аутентификация задана как глобальное значение по умолчанию;<br>**Disabled** - доступ к веб-интерфейсу Zabbix запрещен для этой группы.<br>Нажав на текущий уровень, вы можете изменить его. |
| *Debug mode* | Отображается состояние режима отладки - *Enabled* или *Disabled*. Нажав на состояние, вы можете изменить его. |
| *Status* | Отображается статус группы пользователей - *Enabled* или *Disabled*. Нажав на статус, вы можете изменить его. |

Чтобы настроить новую группу пользователей, нажмите кнопку *Create user group* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

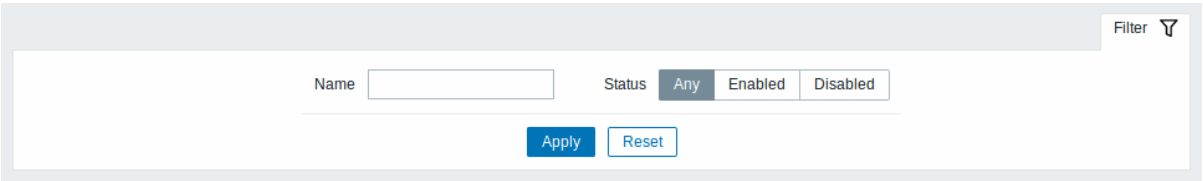
- *Включить* - изменить статус группы пользователей на *Включено*
- *Отключить* - изменить статус группы пользователей на *Отключено*
- *Включить режим отладки* - включить режим отладки для групп пользователей
- *Отключить режим отладки* - отключить режим отладки для групп пользователей
- *Удалить* - удалить группы пользователей

Чтобы использовать эти параметры, отметьте флажки перед нужными группами пользователей, затем нажмите требуемую кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те группы пользователей, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Фильтр* доступна над списком групп пользователей. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать группы пользователей по имени и статусу.




### 2 Роли пользователей

Обзор

В разделе *Users* > *User roles* вы можете создавать роли пользователей.

Роли пользователей позволяют создавать детализированные разрешения на основе изначально выбранного типа пользователя (*User*, *Admin*, *Super admin*).

При выборе типа пользователя все доступные разрешения для этого типа предоставляются (по умолчанию отмечены).

Разрешения можно отзывать только из подмножества, доступного для данного типа пользователя; их нельзя расширить за пределы того, что доступно для данного типа пользователя.

Флажки для недоступных разрешений отображаются серым цветом; пользователи не смогут получить доступ к элементу даже при вводе прямого URL-адреса к этому элементу в браузере.

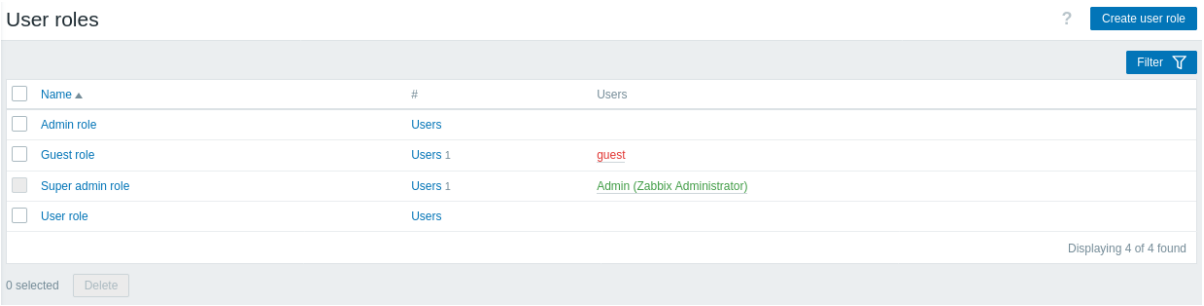
**Note:**
Ограничение доступа к некоторым элементам интерфейса только предотвращает открытие этой страницы — оно не удаляет возможность доступа к базовым данным в других частях интерфейса.

Роли пользователей можно назначать системным пользователям. Каждому пользователю может быть назначена только одна роль.

Роли пользователей по умолчанию

По умолчанию в Zabbix настроены четыре роли пользователей, для которых задан предопределенный набор прав доступа:

- Роль Guest
- Роль User
- Роль Admin
- Роль Super admin

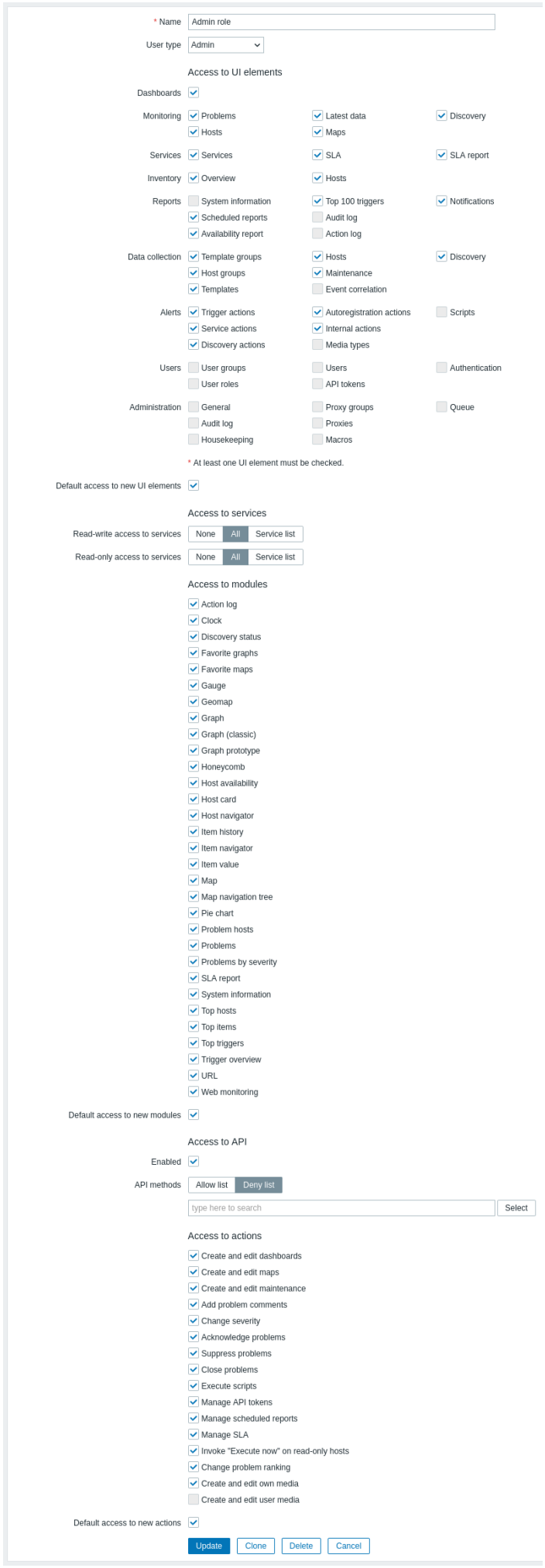



Они основаны на основных типах пользователей в Zabbix. Отображается список всех пользователей, назначенных соответствующей роли. Пользователи, входящие в отключенные группы, отображаются красным цветом. *Роль Guest* — это роль типа пользователя, которая имеет только права на просмотр некоторых разделов веб-интерфейса.

**Note:**
Роль *Super admin* по умолчанию нельзя изменить или удалить, поскольку в Zabbix должен существовать как минимум один пользователь Super admin с неограниченными привилегиями. Пользователи типа *Super admin* могут изменять настройки своей собственной роли, но не тип пользователя.

Конфигурация

Чтобы создать новую роль, нажмите кнопку *Create user role* в правом верхнем углу. Чтобы обновить существующую роль, нажмите на имя роли, чтобы открыть форму конфигурации.


* Name
Admin role
User type
Admin
Access to UI elements
Dashboards
Monitoring
Problems
Latest data
Discovery
Hosts
Maps
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 correlation
Alerts
Trigger actions
Autoregistration actions
Scripts
Service actions
Internal actions
Discovery actions
Media types
Users
User groups
Users
Authentication
User roles
API tokens
Administration
General
Proxy groups
Queue
Audit log
Proxies
Housekeeping
Macros
* At least one UI element must be checked.
Default access to new UI elements
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 history
Item navigator
Item value
Map
Map navigation tree
Pie chart
Problem hosts
Problems
Problems by severity
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
type here to search
Select
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
Default access to new actions
Update
Clone
Delete
Cancel

Отображаются доступные разрешения. Чтобы отозвать определенное разрешение, снимите флажок в соответствующем поле.

Доступные разрешения, а также значения по умолчанию для каждой преднастроенной роли пользователя в Zabbix, описаны ниже.

Права доступа по умолчанию

**Доступ к элементам интерфейса**

Доступ по умолчанию к разделам меню зависит от типа пользователя. См. страницу Permissions для подробностей.

**Доступ к другим параметрам**

| **Параметр** | **Описание** | **Роли пользователей по умолчанию** | | | |
|---|---|---|---|---|---|
| | | *Роль супер-администратора* | *Роль администратора* | *Роль пользователя* | *Роль гостя* |
| *Доступ по умолчанию к новым элементам интерфейса* | Этот параметр определяет, как будет предоставляться доступ к новым разделам меню после обновления Zabbix. Существующие разделы меню модулей остаются без изменений. | Да | Да | Да | Да |
| **Доступ к сервисам** | | | | | |
| Доступ на чтение и запись к сервисам | Выберите доступ на чтение и запись к сервисам:<br>**Нет** - доступа нет<br>**Все** - доступ на чтение и запись ко всем сервисам<br>**Список сервисов** - выбрать сервисы для доступа на чтение и запись<br><br>Доступ на чтение и запись, если он предоставлен, имеет приоритет над настройками доступа только на чтение и динамически наследуется дочерними сервисами. | Все | Все | Нет | Нет |
| Доступ на чтение и запись к сервисам с тегом | Укажите имя тега и, при необходимости, значение, чтобы дополнительно предоставить доступ на чтение и запись к сервисам, соответствующим тегу.<br>Этот параметр доступен, если в параметре *Доступ на чтение и запись к сервисам* выбрано значение 'Список сервисов'.<br>Доступ на чтение и запись, если он предоставлен, имеет приоритет над настройками доступа только на чтение и динамически наследуется дочерними сервисами. | | | | |
| Доступ только на чтение к сервисам | Выберите доступ только на чтение к сервисам:<br>**Нет** - доступа нет<br>**Все** - доступ только на чтение ко всем сервисам<br>**Список сервисов** - выбрать сервисы для доступа только на чтение<br><br>Доступ только на чтение не имеет приоритета над доступом на чтение и запись и динамически наследуется дочерними сервисами. | | | Все | Все |

| | | | | | |
|---|---|---|---|---|---|
| Доступ только на чтение к сервисам с тегом | Укажите имя тега и, при необходимости, значение, чтобы дополнительно предоставить доступ только на чтение к сервисам, соответствующим тегу.<br>Этот параметр доступен, если в параметре *Доступ только на чтение к сервисам* выбрано значение 'Список сервисов'.<br>Доступ только на чтение не имеет приоритета над доступом на чтение и запись и динамически наследуется дочерними сервисами. | | | | |
| **Доступ к модулям** | | | | | |
| <Имя модуля> | Разрешить/запретить доступ к определенному модулю. В этом разделе отображаются только включенные модули. Невозможно предоставить или ограничить доступ к модулю, который в данный момент отключен. | Да | Да | Да | Да |
| *Доступ по умолчанию к новым модулям* | Этот параметр определяет, как будут доступны новые модули и виджеты после обновления Zabbix. Он также применяется к модулям и виджетам, добавленным в разделе *Администрирование* > *Общие* > *Модули*. | | | | |
| **Доступ к API** | | | | | |
| *Включено Методы API* | Включить/отключить доступ к API.<br>Выберите *Список разрешений*, чтобы разрешить, или *Список запретов*, чтобы запретить методы API, указанные в поле поиска. Обратите внимание, что нельзя разрешить одни методы API и запретить другие.<br><br>В поле поиска начните вводить имя метода, затем выберите метод из списка автодополнения.<br>Вы также можете нажать кнопку *Выбрать* и выбрать методы из полного списка, доступного для этого типа пользователя. Обратите внимание, что если какой-либо параметр из блока Доступ к действиям не отмечен, пользователи не смогут использовать методы API, связанные с этим действием.<br><br>Поддерживаются подстановочные знаки.<br>Примеры: `dashboard.*` (все методы API-сервиса 'dashboard.') `*` (любой метод), `*.export` (методы с именем '.export' из всех API-сервисов).<br><br>Если методы не указаны, правило *Список разрешений/Список запретов* будет проигнорировано. | Да | Да | Да | Нет |
| **Доступ к действиям** | | | | | |
| Создание и редактирование панелей | Снятие этого флажка также отменит права на использование методов API `.create`, `.update` и `.delete` для соответствующих элементов. | Да | Да | Да | Нет |
| Создание и редактирование карт | | | | | |
| Создание и редактирование обслуживания | | | | Нет | |

| | | | |
|---|---|---|---|
| Добавление комментариев к проблемам<br>Изменение серьезности<br>Подтверждение проблем<br>Подавление проблем<br>Закрытие проблем | Снятие этого флажка также отменит права на выполнение соответствующего действия через метод API `event.acknowledge`. | | Да |
| Выполнение скриптов | Снятие этого флажка также отменит права на использование метода API `script.execute`. | | |
| Управление токенами API | Снятие этого флажка также отменит права на использование всех методов API `token.*`. | | |
| Управление запланированными отчетами | Снятие этого флажка также отменит права на использование всех методов API `report.*`. | | Нет |
| Управление SLA | Включить/отключить права на управление SLA. | | |
| Вызов ”Выполнить сейчас” на узлах сети только для чтения | Разрешить использовать параметр ”Выполнить сейчас” в последних данных для элементов данных узлов сети только для чтения. | | Да |
| Изменение ранжирования проблемы | Разрешить изменять ранжирование проблемы с причины на симптом и наоборот. | | |
| Создание и редактирование собственных медиа | Разрешить создавать/редактировать собственные медиа. | | |
| Создание и редактирование медиа пользователей | Разрешить создавать/редактировать медиа для пользователей. Этот параметр доступен только пользователям с ролью супер-администратора. | Нет | Нет |
| Доступ по умолчанию к новым действиям | Этот параметр определяет, как будут доступны новые действия после обновления Zabbix. | Да | Да |

См. также:

- Настройка пользователя

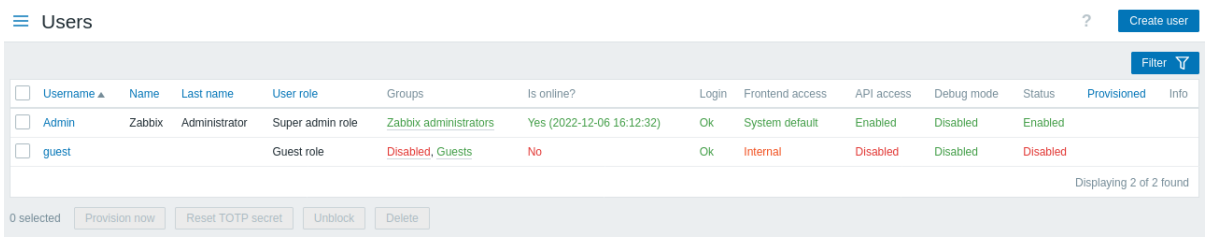
**3 Пользователи**

Обзор

В разделе *Users* → *Users* ведется учет пользователей системы.

Пользователи

Отображается список существующих пользователей с их подробными сведениями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Username* | Имя пользователя для входа в Zabbix. При нажатии на имя пользователя открывается форма настройки. |
| *Name* | Имя пользователя. |
| *Last name* | Фамилия пользователя. |
| *User role* | Отображается роль пользователя. |
| *Groups* | Перечислены группы, членом которых является пользователь. При нажатии на имя группы пользователей открывается форма настройки группы пользователей.<br>Отключенные группы отображаются красным цветом. |
| *Is online?* | Отображается статус пользователя в сети - *Yes* или *No*. В скобках отображается время последней активности пользователя. |
| *Login* | Отображается статус входа пользователя - *Ok* или *Blocked*. Пользователь может быть временно заблокирован при превышении числа неудачных попыток входа, заданного в разделе *Administration → General → Other* (по умолчанию пять). Нажав *Blocked*, вы можете разблокировать пользователя. |
| *Frontend access* | Отображается уровень доступа к веб-интерфейсу - *System default*, *Internal*, *LDAP* или *Disabled* в зависимости от значения, заданного для всей группы пользователей. |
| *API access* | Отображается статус доступа к API - *Enabled* или *Disabled* в зависимости от значения, заданного для роли пользователя. |
| *Debug mode* | Отображается статус режима отладки - *Enabled* или *Disabled* в зависимости от значения, заданного для всей группы пользователей. |
| *Status* | Отображается статус пользователя - *Enabled* или *Disabled* в зависимости от значения, заданного для всей группы пользователей. |
| *Provisioned* | Отображается дата последнего предоставления пользователю доступа.<br>Используется для пользователей, созданных с помощью JIT-provisioning из LDAP/SAML. |
| *Info* | Отображается информация об ошибках.<br>Желтое предупреждение отображается для пользователей без групп пользователей.<br>Красное предупреждение отображается для пользователей без ролей, а также для пользователей без ролей и групп пользователей. |

Чтобы настроить нового пользователя, нажмите кнопку *Create user* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько параметров массового редактирования:

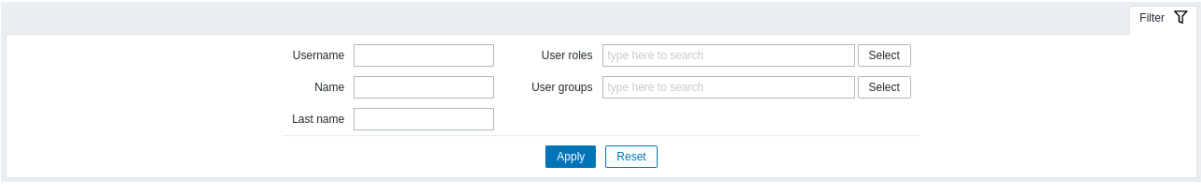
- *Provision now* - обновить информацию о пользователе из LDAP (эта опция доступна только если выбран пользователь LDAP)
- *Reset TOTP secret* - сбросить секреты TOTP пользователя для всех методов TOTP и удалить сеанс пользователя (эта опция доступна только если включена MFA; для пользователей без секретов TOTP их сеанс не будет удален)
- *Unblock* - снова предоставить доступ к системе заблокированным пользователям
- *Delete* - удалить пользователей

Чтобы использовать эти параметры, отметьте флажки напротив соответствующих пользователей, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только тех пользователей, которые вас интересуют. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

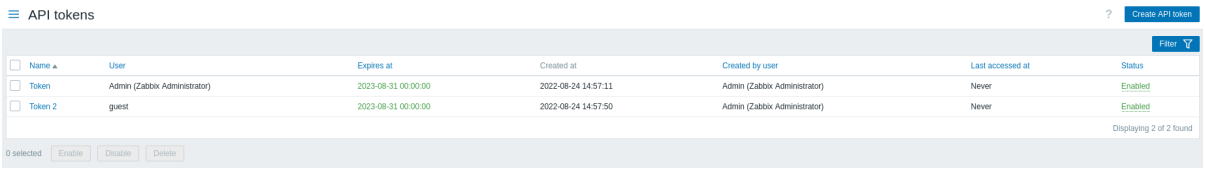
Ссылка *Filter* доступна над списком пользователей. Если вы нажмете на нее, станет доступен фильтр, с помощью которого можно отфильтровать пользователей по имени пользователя, имени, фамилии, роли пользователя и группе пользователей.




**4 API токены**

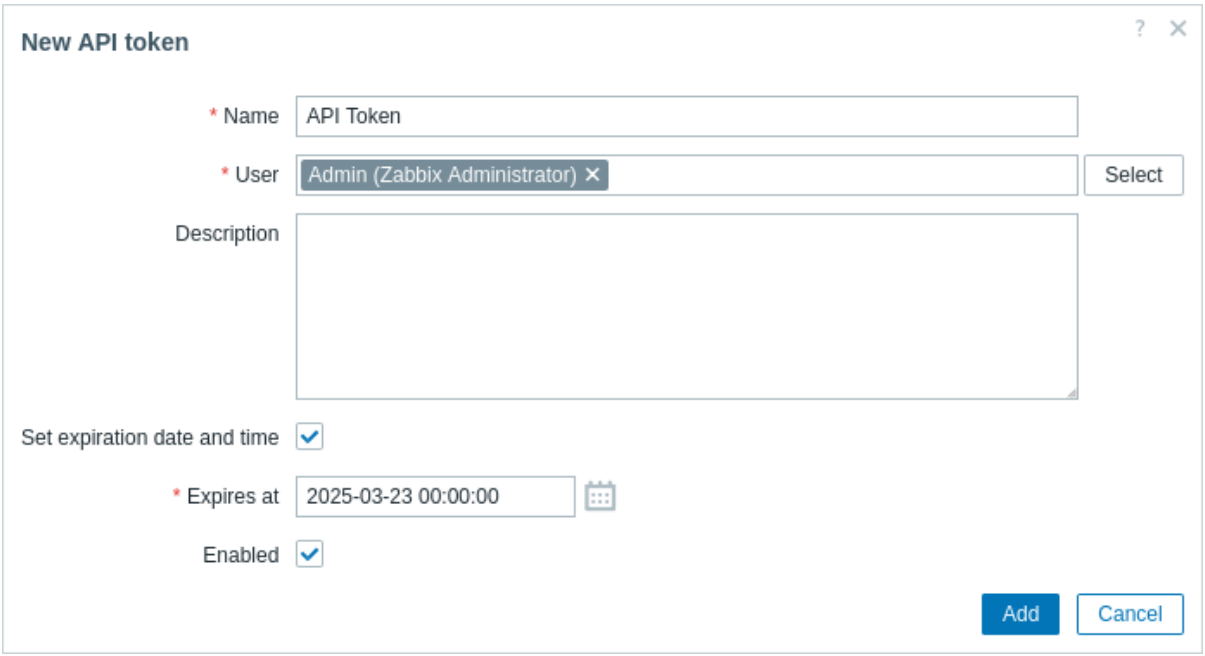
Обзор

Этот раздел позволяет создавать и управлять API-токенами.



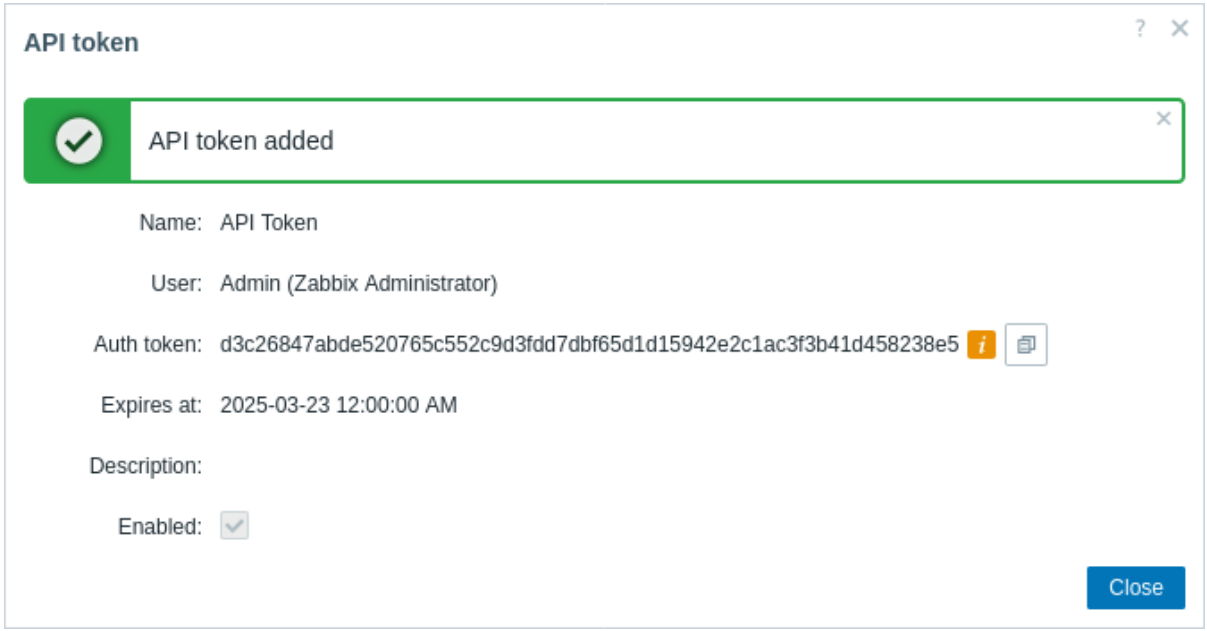

Вы можете фильтровать API-токены по имени, пользователям, которым назначены токены, дате истечения срока действия, пользователям, создавшим токены, или статусу (включен/выключен). Щелкните статус токена в списке, чтобы быстро включить или выключить токен. Вы также можете массово включать или выключать токены, выбрав их в списке, а затем нажав кнопки *Включить/Выключить* под списком.

Чтобы создать новый токен, нажмите кнопку *Создать API-токен* в правом верхнем углу, затем заполните обязательные поля на экране настройки токена:




| Параметр | Описание |
|---|---|
| Имя | Отображаемое имя токена. |
| Пользователь | Пользователь, которому должен быть назначен токен. Чтобы быстро выбрать пользователя, начните вводить имя пользователя, имя или фамилию, затем выберите нужного пользователя из списка автодополнения. Кроме того, вы можете нажать кнопку *Выбрать* и выбрать пользователя из полного списка пользователей. Токен может быть назначен только одному пользователю. |
| Описание | Необязательное описание токена. |
| Установить дату и время истечения срока действия | Снимите этот флажок, если у токена не должно быть даты истечения срока действия. |
| Дата истечения срока действия | Щелкните значок календаря, чтобы выбрать дату истечения срока действия токена, или введите дату вручную в формате YYYY-MM-DD hh:mm:ss |
| Включен | Снимите этот флажок, если нужно создать токен в выключенном состоянии. |

Нажмите *Добавить*, чтобы создать токен.




Скопируйте значение *Auth token* и сохраните его в надежном месте **до закрытия страницы**, затем нажмите *Закрыть*. Токен появится в списке.

**Warning:**
Значение *Auth token* нельзя будет просмотреть снова позже. Оно доступно только сразу после создания токена. Если вы потеряете сохраненный токен, вам придется сгенерировать его заново, и при этом будет создана новая строка авторизации.

Щелкните имя токена, чтобы изменить имя, описание, настройки даты истечения срока действия или статус токена. Обратите внимание, что изменить пользователя, которому назначен токен, невозможно. Нажмите кнопку *Обновить*, чтобы сохранить изменения. Если токен был утерян или скомпрометирован, вы можете нажать кнопку *Перегенерировать*, чтобы создать новое значение токена. Появится диалоговое окно подтверждения с просьбой подтвердить это действие, поскольку после продолжения ранее сгенерированный токен станет недействительным.

Пользователи без доступа к разделу меню *Администрирование* могут просматривать и изменять сведения о назначенных им токенах только в разделе *Профиль пользователя → API-токены* section, если в их правах роли пользователя разрешено *Управление API-токенами*.

**5 Аутентификация**

Обзор

Раздел *Users → Authentication* позволяет указать метод аутентификации пользователей для Zabbix и внутренние требования к паролям.

Доступные методы аутентификации: внутренняя аутентификация, HTTP, LDAP, SAML и MFA-аутентификация.

Аутентификация по умолчанию

По умолчанию Zabbix использует **внутреннюю** аутентификацию Zabbix для всех пользователей.

Можно изменить метод аутентификации по умолчанию на системный LDAP. Для этого перейдите на вкладку *LDAP* и настройте параметры LDAP, затем вернитесь на вкладку *Authentication* и переключите селектор *Default authentication* на LDAP.

Обратите внимание, что метод аутентификации можно тонко настраивать на уровне группы пользователей. Даже если аутентификация LDAP включена глобально, некоторые группы пользователей по-прежнему могут проходить аутентификацию через Zabbix. Для этих групп параметр frontend access должен быть установлен в Internal.

Также можно включить аутентификацию LDAP только для определенных групп пользователей, если глобально используется внутренняя аутентификация. В этом случае можно указать и использовать сведения для аутентификации LDAP для конкретных групп пользователей, для которых параметр frontend access должен быть установлен в LDAP. Если пользователь входит хотя бы в одну группу пользователей с аутентификацией LDAP, он не сможет использовать внутренний метод аутентификации.

Методы аутентификации HTTP, SAML 2.0 и MFA можно использовать дополнительно к методу аутентификации по умолчанию.

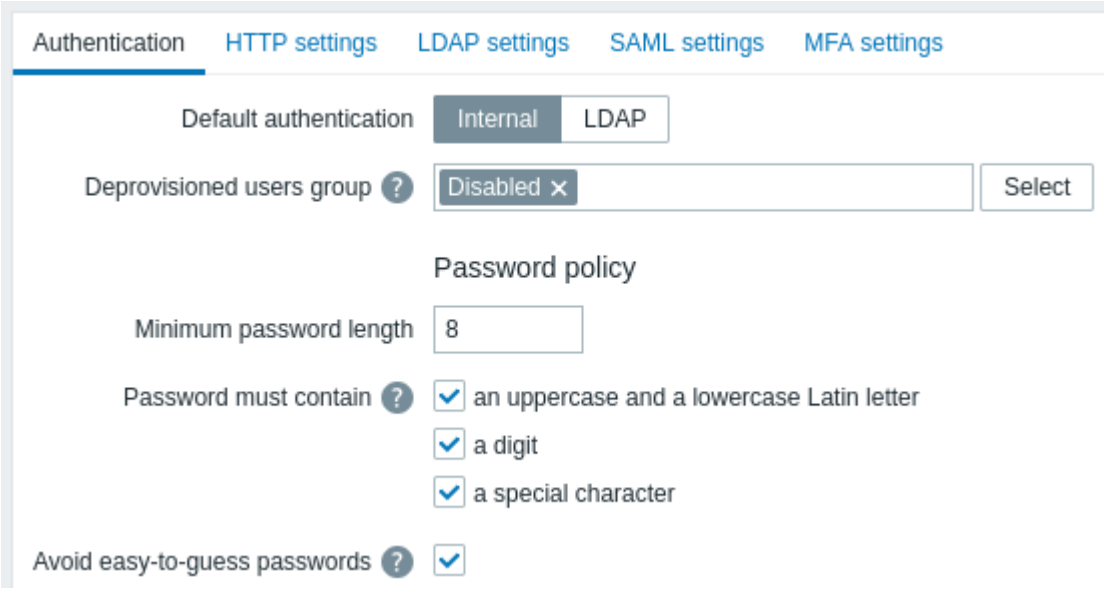
Zabbix поддерживает just-in-time (JIT) provisioning, что позволяет создавать учетные записи пользователей в Zabbix при первом входе внешнего пользователя и выполнять их подготовку. JIT provisioning поддерживается для LDAP и SAML.

См. также:

- HTTP authentication
- LDAP authentication
- SAML authentication
- MFA authentication

Конфигурация

Вкладка *Аутентификация* позволяет задать метод аутентификации по умолчанию, указать группу для деактивированных пользователей и настроить требования к сложности паролей для пользователей Zabbix.




Параметры конфигурации:

| Parameter | Description |
|---|---|
| *Default authentication* | Выберите метод аутентификации по умолчанию для Zabbix - *Internal* или *LDAP*. |
| *Deprovisioned users group* | Укажите группу пользователей для деактивированных пользователей. Этот параметр требуется только для JIT provisioning в отношении пользователей, которые были созданы в Zabbix из систем LDAP или SAML, но больше не нуждаются в provisioning.<br>Необходимо указать отключенную группу пользователей. |
| *Minimum password length* | По умолчанию минимальная длина пароля установлена в 8. Поддерживаемый диапазон: 1-70. Обратите внимание, что пароли длиной более 72 символов будут усечены. |
| *Password must contain* | Отметьте один или несколько флажков, чтобы потребовать использование в пароле указанных символов:<br>- заглавную и строчную латинскую букву<br>- цифру<br>- специальный символ<br><br>Наведите указатель на знак вопроса, чтобы увидеть подсказку со списком символов для каждого варианта. |
| *Avoid easy-to-guess passwords* | Если отмечено, пароль будет проверяться на соответствие следующим требованиям:<br>- не должен содержать имя, фамилию или имя пользователя<br>- не должен быть одним из распространенных или контекстно-зависимых паролей.<br><br>Список распространенных и контекстно-зависимых паролей автоматически формируется на основе списка NCSC ”Top 100k passwords”, списка SecLists ”Top 1M passwords” и списка контекстно-зависимых паролей Zabbix. Внутренним пользователям не будет разрешено задавать пароли, включенные в этот список, поскольку такие пароли считаются слабыми из-за их широкого использования. |

Изменения в требованиях к сложности паролей не повлияют на существующие пароли пользователей, но если существующий пользователь решит изменить пароль, новый пароль должен будет соответствовать текущим требованиям. Подсказка со списком требований будет отображаться рядом с полем *Password* в профиле пользователя и в форме настройки пользователя, доступной из меню *Users* → *Users*.

1 HTTP

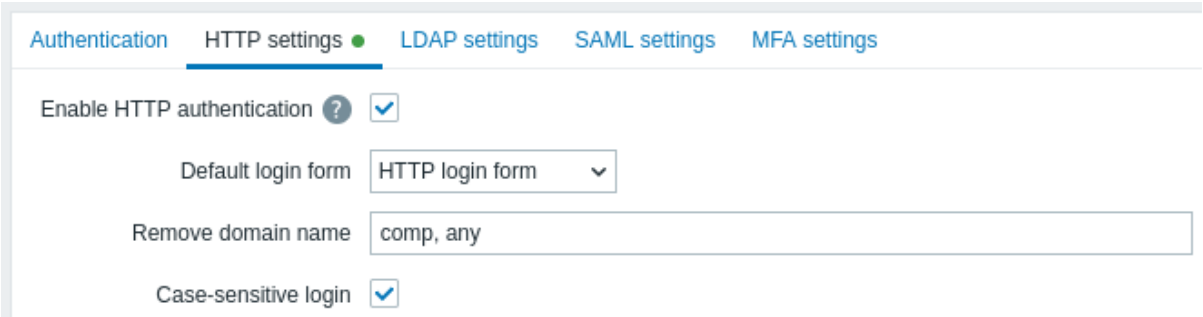
Обзор

Аутентификация на основе HTTP или веб-сервера authentication (например, BasicAuthentication, NTLM/Kerberos) может использоваться для проверки имен пользователей и паролей. Обратите внимание, что пользователь также должен существовать в Zabbix, однако его пароль Zabbix использоваться не будет.

**Attention:**
Будьте внимательны! Перед включением убедитесь, что аутентификация веб-сервера настроена и работает корректно.

Аутентификацию HTTP можно отключить во веб-интерфейсе, настроив соответствующий параметр на вкладке *HTTP settings* в разделе *Users > Authentication*. Когда аутентификация HTTP отключена, вкладка с параметрами аутентификации HTTP не будет отображаться во веб-интерфейсе. Обратите внимание, что переустановка веб-интерфейса (запуск setup.php) сбросит настройки аутентификации, включая конфигурацию аутентификации HTTP.

Конфигурация




Параметры конфигурации:

| Parameter | Description |
|---|---|
| *Enable HTTP authentication* | Установите флажок, чтобы включить HTTP-аутентификацию. При наведении курсора на появится всплывающая подсказка с предупреждением о том, что в случае аутентификации на веб-сервере все пользователи (даже с доступом к веб-интерфейсу, установленным в LDAP/Internal) будут проходить аутентификацию на веб-сервере, а не в Zabbix. |
| *Default login form* | Укажите, куда перенаправлять неаутентифицированных пользователей:<br>**Zabbix login form** - стандартная страница входа Zabbix.<br>**HTTP login form** - страница входа HTTP.<br>Рекомендуется включать аутентификацию на основе веб-сервера только для страницы `index_http.php`. Если для *Default login form* задано значение 'HTTP login page', пользователь будет автоматически входить в систему, если модуль аутентификации веб-сервера установит корректный логин пользователя в переменной `$_SERVER`.<br>Поддерживаемые ключи `$_SERVER`: `PHP_AUTH_USER`, `REMOTE_USER`, `AUTH_USER`. |
| *Remove domain name* | Список доменных имен, разделенных запятыми, которые следует удалять из имени пользователя.<br>Например, `comp,any` - если имя пользователя 'Admin@any', 'comp\Admin', пользователь войдет в систему как 'Admin'; если имя пользователя 'notacompany\Admin', вход будет запрещен. |

| Parameter | Description |
|---|---|
| *Case-sensitive login* | Снимите флажок, чтобы отключить чувствительность к регистру при входе для имен пользователей (по умолчанию включено).<br>Отключение чувствительности к регистру позволяет, например, входить как ”admin”, даже если пользователь Zabbix — ”Admin” или ”ADMIN”.<br>Обратите внимание: если чувствительность к регистру отключена и в Zabbix есть несколько пользователей с похожими именами (например, Admin и admin), вход для этих пользователей всегда будет запрещен со следующим сообщением об ошибке: ”Authentication failed: supplied credentials are not unique.” |

**Note:**
Для внутренних пользователей, которые не могут войти с использованием HTTP-учетных данных (если в качестве значения по умолчанию задана HTTP login form), что приводит к ошибке 401, можно добавить строку `ErrorDocument 401 /index.php?form=default` в директивы базовой аутентификации, чтобы выполнялось перенаправление на обычную форму входа Zabbix.

2 LDAP

Обзор

Внешний LDAP сервис authentication Может быть использован для проверки логинов и паролей пользователей

Аутентификация LDAP в Zabbix работает как минимум с Microsoft Active Directory и OpenLDAP.

Если настроен только вход через LDAP, пользователь также должен существовать в Zabbix, однако его пароль из Zabbix использоваться не будет. Если аутентификация проходит успешно, Zabbix сопоставляет локальное имя пользователя с атрибутом имени пользователя, возвращаемым LDAP.

Подготовка пользователей

Можно настроить JIT (just-in-time) **подготовку пользователей** для пользователей LDAP. В этом случае не требуется, чтобы пользователь уже существовал в Zabbix. Учетная запись пользователя может быть создана при первом входе пользователя в Zabbix.

Когда пользователь LDAP вводит свой логин и пароль LDAP, Zabbix проверяет *default* LDAP-сервер, существует ли этот пользователь. Если пользователь существует и еще не имеет учетной записи в Zabbix, в Zabbix создается новый пользователь, и он может войти в систему.

Пользователь, созданный с помощью JIT-подготовки, связан с LDAP-сервером (каталогом), который был задан как default на момент создания. Изменение default LDAP-сервера позже не изменяет и не обновляет LDAP-сервер, связанный с пользователями, которые уже были подготовлены.

**Attention:**
Если JIT-подготовка включена, в вкладке *Authentication* необходимо указать группу пользователей для неподготовленных пользователей.

JIT-подготовка также позволяет обновлять подготовленные учетные записи пользователей на основе изменений в LDAP. Например, если пользователя переместили из одной LDAP-группы в другую, в Zabbix он также будет перемещен из одной группы в другую; если пользователя удалить из LDAP-группы, он также будет удален из группы в Zabbix и, если не принадлежит ни к какой другой группе, добавлен в группу пользователей для неподготовленных пользователей. Подготовленные учетные записи пользователей обновляются на основе настроенного периода подготовки или при входе пользователя в Zabbix.

Обратите внимание, что фоновая подготовка выполняется веб-интерфейсом Zabbix, когда пользователь взаимодействует с ним или, по крайней мере, имеет открытую страницу веб-интерфейса в браузере. Отдельных фоновых процессов для подготовки пользователей нет.

LDAP поддерживает три способа привязки к каталогу для аутентификации и поиска:

- Анонимная привязка — не указан *Bind DN* / *Bind password*, и LDAP-сервер разрешает анонимные запросы.
- Выделенный пользователь привязки (служебная учетная запись) — в *Bind DN* / *Bind password* задана конкретная учетная запись LDAP, которая используется Zabbix для поиска и подготовки. Это рекомендуемый и наиболее гибкий вариант, поскольку Zabbix может выполнять поиск и фоновую подготовку без учетных данных конечного пользователя.

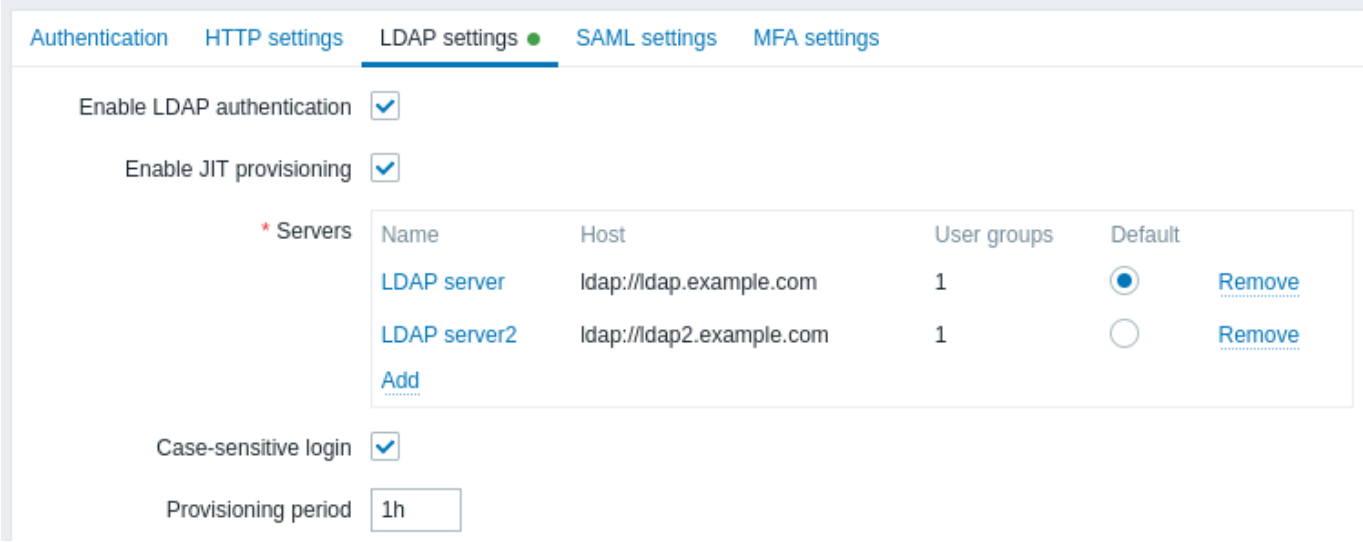
- Прямая привязка пользователя — Zabbix выполняет привязку, используя учетные данные, которые пользователь вводит при входе в систему (не настроены *Bind DN* / *Bind password*); это настраивается путем включения заполнителя, например `uid=%{user}`, в Base DN. В этом режиме Zabbix имеет доступ только к паролю пользователя во время интерактивного входа. В результате действия подготовки, требующие аутентификации в LDAP вне сеанса входа пользователя (например, использование кнопки *Provision now* или запуск фоновой подготовки, когда пользователь не вошел в систему), не могут пройти аутентификацию и, следовательно, не будут работать. Подготовка и обновления при прямой привязке пользователя выполняются только в момент входа пользователя в систему.

Несколько серверов

При необходимости можно определить несколько LDAP-серверов. Например, для аутентификации разных групп пользователей можно использовать разные серверы. После настройки LDAP-серверов в конфигурации группы пользователей становится возможным выбрать нужный LDAP-сервер для соответствующей группы пользователей.

Если пользователь входит в несколько групп пользователей и для них настроено несколько LDAP-серверов, для аутентификации будет использоваться первый сервер в списке LDAP-серверов, отсортированном по имени по возрастанию.

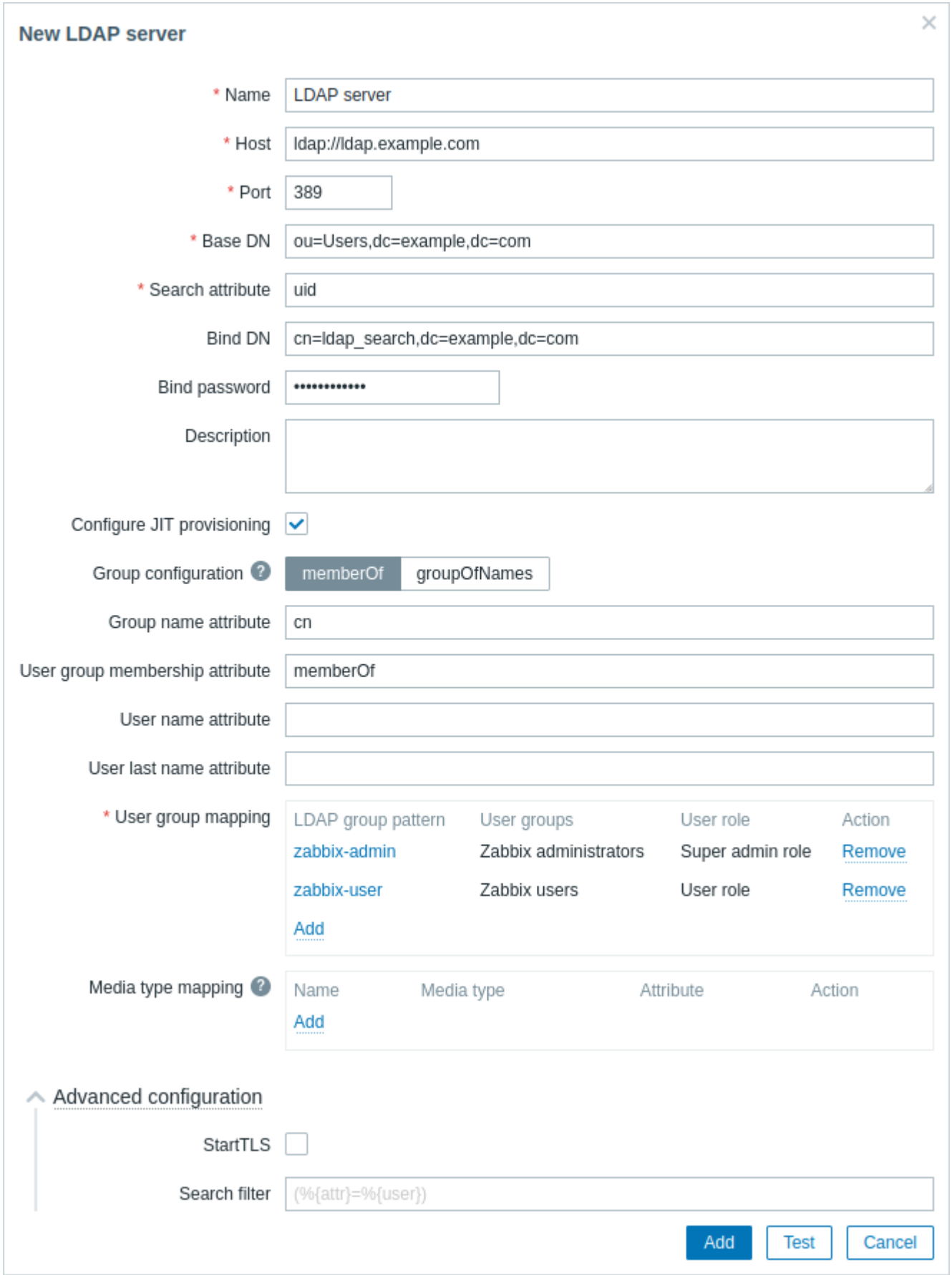
Конфигурация




Параметры конфигурации:

| Parameter | Description |
|---|---|
| *Enable LDAP authentication* | Установите флажок, чтобы включить аутентификацию LDAP. |
| *Enable JIT provisioning* | Установите флажок, чтобы включить JIT provisioning. |
| *Servers* | Нажмите *Add*, чтобы настроить LDAP-сервер (см. ниже Конфигурация LDAP-сервера). |
| *Case-sensitive login* | Снимите флажок, чтобы отключить чувствительность имени пользователя к регистру (по умолчанию включено).<br>Отключение чувствительности имени пользователя к регистру позволяет, например, войти как ”admin”, даже если пользователь Zabbix — ”Admin” или ”ADMIN”.<br>Обратите внимание: если чувствительность имени пользователя к регистру отключена и существует несколько пользователей Zabbix с похожими именами (например, Admin и admin), вход для этих пользователей всегда будет запрещен со следующим сообщением об ошибке: ”Authentication failed: supplied credentials are not unique.” |
| *Provisioning period* | Укажите период provisioning, то есть как часто вошедший в систему пользователь будет provisioned при работе с веб-интерфейсом. |

Конфигурация LDAP-сервера




Параметры конфигурации LDAP-сервера:

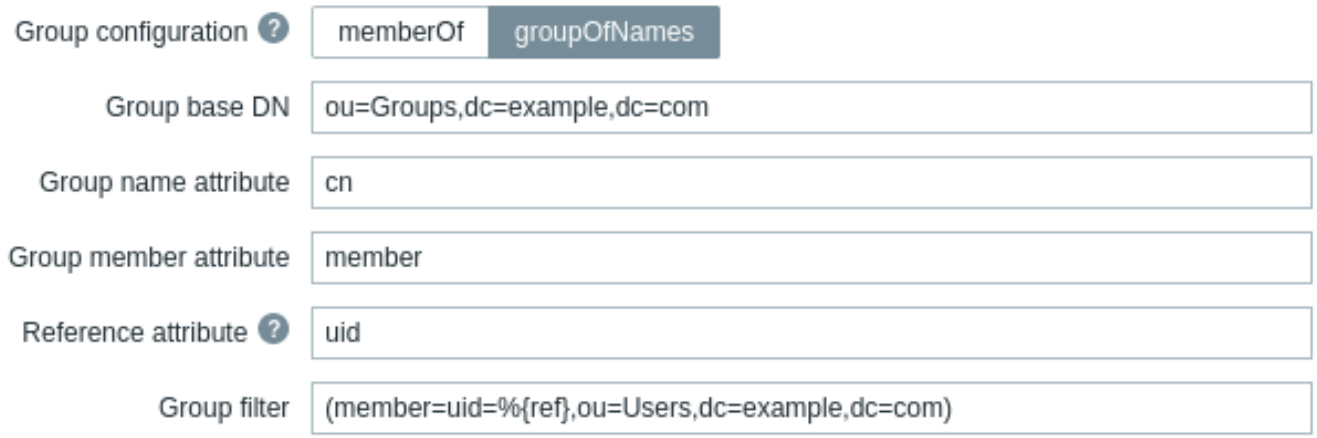
| Parameter | Description |
|---|---|
| *Name* | Имя LDAP-сервера в конфигурации Zabbix. |

| Parameter | Description |
|---|---|
| *Host* | Имя хоста, IP-адрес или URI LDAP-сервера. Примеры: ldap.example.com, 127.0.0.1, ldap://ldap.example.com<br>Для защищенного LDAP-сервера используйте протокол *ldaps* и имя хоста. Пример: ldaps://ldap.example.com<br>Для OpenLDAP 2.x.x и новее можно использовать полный LDAP URI вида ldap://hostname:port или ldaps://hostname:port. |
| *Port* | Порт LDAP-сервера. По умолчанию 389.<br>Для защищенного LDAP-подключения номер порта обычно 636.<br>Не используется при применении полных LDAP URI. |
| *Base DN* | Базовый путь к учетным записям пользователей на LDAP-сервере:<br>ou=Users,ou=system (для OpenLDAP),<br>DC=company,DC=com (для Microsoft Active Directory)<br>uid=%{user},dc=example,dc=com (для прямой привязки пользователя, см. примечание ниже) |
| *Search attribute* | Атрибут учетной записи LDAP, используемый для поиска:<br>uid (для OpenLDAP),<br>sAMAccountName (для Microsoft Active Directory) |
| *Bind DN* | Учетная запись LDAP для привязки и поиска на LDAP-сервере, примеры:<br>uid=ldap_search,ou=system (для OpenLDAP),<br>CN=ldap_search,OU=user_group,DC=company,DC=com (для Microsoft Active Directory)<br>Также поддерживается анонимная привязка. Обратите внимание, что анонимная привязка потенциально открывает конфигурацию домена для неавторизованных пользователей (сведения о пользователях, компьютерах, серверах, группах, службах и т. д.). В целях безопасности отключите анонимные привязки на LDAP-хостах и используйте вместо этого аутентифицированный доступ. |
| *Bind password* | Пароль LDAP-учетной записи для привязки и поиска на LDAP-сервере. |
| *Description* | Описание LDAP-сервера. |
| *Configure JIT provisioning* | Отметьте этот флажок, чтобы показать параметры, связанные с JIT provisioning. |
| *Group configuration* | Выберите метод настройки групп:<br>**memberOf** - путем поиска пользователей и их атрибута членства в группе<br>**groupOfNames** - путем поиска групп через атрибут member<br>Обратите внимание, что memberOf предпочтительнее, так как он работает быстрее; используйте groupOfNames, если ваш LDAP-сервер не поддерживает `memberOf` или требуется фильтрация групп. |
| *Group name attribute* | Укажите атрибут, из которого будет получено имя группы для всех объектов в атрибуте `memberOf` (см. поле *User group membership attribute*)<br>Имя группы необходимо для сопоставления групп пользователей. |
| *User group membership attribute* | Укажите атрибут, содержащий сведения о группах, к которым принадлежит пользователь (например, `memberOf`).<br>Например, атрибут memberOf может содержать такие сведения:<br>`memberOf=cn=zabbix-admin,ou=Groups,dc=example,dc=com`<br>Это поле доступно только для метода memberOf. |
| *User name attribute* | Укажите атрибут, содержащий имя пользователя. |
| *User last name attribute* | Укажите атрибут, содержащий фамилию пользователя. |
| *User group mapping* | Сопоставьте шаблон группы пользователей LDAP с группой пользователей и ролью пользователя Zabbix.<br>Это необходимо, чтобы определить, какую группу пользователей/роль получит подготовленный пользователь в Zabbix.<br>Нажмите *Add*, чтобы добавить сопоставление.<br>Поле *LDAP group pattern* поддерживает подстановочные знаки. Имя группы должно совпадать с существующей группой.<br>Если пользователь LDAP соответствует нескольким группам пользователей Zabbix, он становится участником всех из них.<br>Если пользователь соответствует нескольким ролям пользователей Zabbix, он получит ту, у которой самый высокий уровень прав среди них. |
| *Media type mapping* | Сопоставьте атрибуты LDAP media пользователя (например, email) с пользовательскими media Zabbix для отправки уведомлений (значение атрибута используется как поле media *Send to*). |
| *Advanced configuration* | Нажмите заголовок *Advanced configuration*, чтобы отобразить дополнительные параметры конфигурации (см. ниже). |

| Parameter | Description |
|---|---|
| *StartTLS* | Отметьте флажок, чтобы использовать операцию StartTLS при подключении к LDAP-серверу. Соединение будет разорвано, если сервер не поддерживает StartTLS. StartTLS нельзя использовать с серверами, которые используют протокол *ldaps*. |
| *Search filter* | Определите пользовательскую строку при аутентификации пользователя в LDAP. Поддерживаются следующие заполнители:<br>`%{attr}` - имя атрибута поиска (uid, sAMAccountName)<br>`%{user}` - значение имени пользователя для аутентификации<br>Например, чтобы выполнить поиск с учетом регистра в среде LDAP или Microsoft Active Directory, где поиск обычно не чувствителен к регистру, строку можно определить следующим образом:<br>`(%{attr}:caseExactMatch:=%{user})`.<br>Если фильтр не настроен, LDAP будет использовать значение по умолчанию:<br>`(%{attr}=%{user})`. |

**Note:**
Чтобы настроить LDAP-сервер для **прямой привязки пользователя**, добавьте атрибут uid=%{user} к параметру *Base DN* (например, *uid=%{user},dc=example,dc=com*) и оставьте параметры *BindDN* и *Bind password* пустыми. При аутентификации заполнитель %{user} будет заменен именем пользователя, введенным при входе в систему.<br>
При прямой привязке пользователя Zabbix имеет доступ только к учетным данным пользователя во время интерактивного входа в систему. Поэтому задачи provisioning, которые выполняются без интерактивного входа в систему (например, кнопка *Provision now* или метод API `user.provision`), будут игнорировать пользователей, которые проходят аутентификацию с использованием прямой привязки пользователя, поскольку Zabbix не может выполнить привязку от их имени. Используйте анонимную привязку или отдельного пользователя привязки (служебную учетную запись), если вам нужно, чтобы provisioning и LDAP-поиск работали из веб-интерфейса без необходимости входа конечного пользователя в систему.

Следующие поля относятся только к ”groupOfNames” как к методу *Group configuration*:




| Parameter | Description |
|---|---|
| *Group base DN* | Базовый путь к группам на LDAP-сервере. |
| *Group name attribute* | Укажите атрибут, из которого будет получено имя группы в указанном базовом пути к группам.<br>Имя группы необходимо для сопоставления групп пользователей. |
| *Group member attribute* | Укажите атрибут, содержащий сведения об участниках группы в LDAP (например, `member`). |
| *Reference attribute* | Укажите ссылочный атрибут для фильтра группы (см. поле *Group filter*).<br>Затем используйте `%{ref}` в фильтре группы, чтобы получить значения для указанного здесь атрибута. |
| *Group filter* | Укажите фильтр для получения группы, членом которой является пользователь.<br>Например, `(member=uid=%{ref},ou=Users,dc=example,dc=com)` совпадет с ”User1”, если атрибут member группы равен `uid=User1,ou=Users,dc=example,dc=com`, и вернет группу, членом которой является ”User1”. |

**Warning:**
В случае проблем с сертификатами, чтобы защищенное LDAP-подключение (ldaps) заработало, может потребоваться добавить строку `TLS_REQCERT allow` в файл конфигурации /etc/openldap/ldap.conf. Это может снизить безопасность подключения к LDAP-каталогу.

**Note:**
Рекомендуется создать отдельную LDAP-учетную запись (*Bind DN*) для выполнения привязки и поиска на LDAP-сервере с минимальными привилегиями в LDAP вместо использования реальных учетных записей пользователей (используемых для входа в веб-интерфейс Zabbix).<br> Такой подход обеспечивает более высокий уровень безопасности и не требует изменения *Bind password*, когда пользователь меняет свой пароль на LDAP-сервере.<br> В таблице выше это имя учетной записи *ldap_search*.

Проверка доступа

Кнопка *Test* позволяет проверить доступ пользователя:

| Parameter | Description |
|---|---|
| *Login* | Имя пользователя LDAP для проверки (предзаполнено текущим именем пользователя из веб-интерфейса Zabbix). Это имя пользователя должно существовать на сервере LDAP. Zabbix не активирует аутентификацию LDAP, если не сможет выполнить аутентификацию тестового пользователя. |
| *User password* | Пароль пользователя LDAP для проверки. |

3 SAML

Обзор

SAML 2.0 аутентификация может использоваться для входа в Zabbix.

Если настроен только вход через SAML, то пользователь также должен существовать в Zabbix, однако его пароль Zabbix использоваться не будет. Если аутентификация успешна, то Zabbix сопоставит локальное имя пользователя с атрибутом имени пользователя, возвращенным SAML.

Подготовка пользователей

Можно настроить JIT (just-in-time) **подготовку пользователей** для пользователей SAML. В этом случае не требуется, чтобы пользователь уже существовал в Zabbix. Учетная запись пользователя может быть создана при первом входе пользователя в Zabbix.

**Attention:**
Если включена JIT-подготовка, на вкладке *Authentication* необходимо указать группу пользователей для деактивированных пользователей.

Помимо JIT-подготовки также можно включить и настроить подготовку SCIM (System for Cross-domain Identity Management) - *непрерывное* управление учетными записями пользователей для тех пользователей, которые были созданы с помощью подготовки пользователей. Для аутентификации в Zabbix при подготовке SCIM требуется API token Zabbix (с правами Super admin).

Например, если пользователь перемещается из одной группы SAML в другую, то в Zabbix он также будет перемещен из одной группы в другую; если пользователь удаляется из группы SAML, он также будет удален из группы в Zabbix и, если не принадлежит ни к какой другой группе, добавлен в группу пользователей для деактивированных пользователей.

Если SCIM включен и настроен, пользователь SAML будет подготовлен в момент входа пользователя в Zabbix и будет непрерывно обновляться на основе изменений в SAML. Уже существующие пользователи SAML не будут подготовлены, и обновляться будут только подготовленные пользователи. Обратите внимание, что при подготовке или обновлении пользователя будут добавляться только допустимые медиа.

Если SCIM не включен, пользователь SAML будет подготовлен (а затем обновлен) в момент входа пользователя в Zabbix.

**Note:**
Если включена аутентификация SAML, пользователи смогут выбирать между локальным входом и единым входом через SAML. Если используется JIT-подготовка, то возможен только единый вход.

Настройка поставщика удостоверений

Для работы с Zabbix поставщик удостоверений SAML (onelogin.com, auth0.com, okta.com и т. д.) должен быть настроен следующим образом:

- *Assertion Consumer URL* должен быть установлен в `<path_to_zabbix_ui>/index_sso.php?acs`
- *Single Logout URL* должен быть установлен в `<path_to_zabbix_ui>/index_sso.php?sls`

Примеры `<path_to_zabbix_ui>`: `https://example.com/zabbix/ui`, `http://another.example.com/zabbix`, `http://<any_public_ip_address>/zabbix`

Настройка Zabbix

> **Attention:**
> Необходимо установить php-openssl, если вы хотите использовать аутентификацию SAML во веб-интерфейсе.

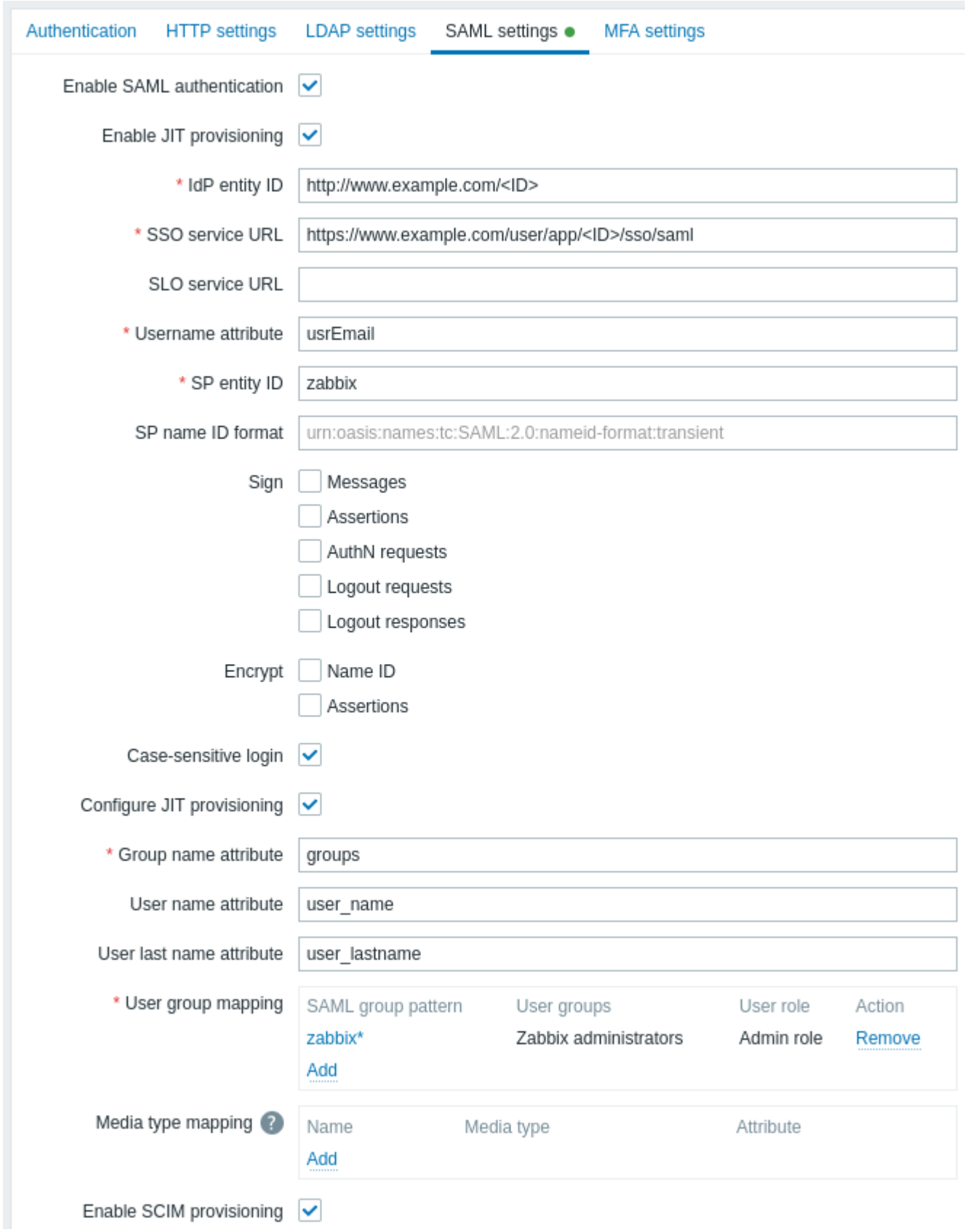
Чтобы использовать аутентификацию SAML, Zabbix следует настроить следующим образом:

1. Закрытый ключ и сертификат должны храниться в *ui/conf/certs/*, если не указаны пользовательские пути в zabbix.conf.php.

По умолчанию Zabbix будет искать файлы в следующих расположениях:

- ui/conf/certs/sp.key - файл закрытого ключа SP
- ui/conf/certs/sp.crt - файл сертификата SP
- ui/conf/certs/idp.crt - файл сертификата IDP

2. Все наиболее важные параметры можно настроить во веб-интерфейсе Zabbix. Однако дополнительные параметры можно указать в файле конфигурации.




Параметры конфигурации, доступные во веб-интерфейсе Zabbix:

| Parameter | Description |
|---|---|
| *Enable SAML authentication* | Отметьте флажок, чтобы включить аутентификацию SAML. |
| *Enable JIT provisioning* | Отметьте флажок, чтобы включить JIT-подготовку пользователей. |
| *IDP entity ID* | Уникальный идентификатор сущности в поставщике удостоверений SAML. |
| *SSO service URL* | URL, на который будут перенаправляться пользователи при входе в систему. |

| Parameter | Description |
|---|---|
| *SLO service URL* | URL, на который будут перенаправляться пользователи при выходе из системы. Если оставить поле пустым, служба SLO использоваться не будет. |
| *Username attribute* | Атрибут SAML, который будет использоваться как имя пользователя при входе в Zabbix. Список поддерживаемых значений определяется поставщиком удостоверений.<br><br>Примеры:<br>uid<br>userprincipalname<br>samaccountname<br>username<br>userusername<br>urn:oid:0.9.2342.19200300.100.1.1<br>urn:oid:1.3.6.1.4.1.5923.1.1.1.13<br>urn:oid:0.9.2342.19200300.100.1.44 |
| *SP entity ID* | Уникальный идентификатор поставщика услуг (если он не совпадает, операция будет отклонена).<br>Можно указать URL или любую строку данных. |
| *SP name ID format* | Запросить в ответе определенный формат Name ID.<br><br>Примеры:<br>urn:oasis:names:tc:SAML:2.0:nameid-format:persistent<br>urn:oasis:names:tc:SAML:2.0:nameid-format:unspecified<br>urn:oasis:names:tc:SAML:2.0:nameid-format:transient |
| *Sign* | Отметьте флажки, чтобы выбрать сущности, для которых должна быть включена подпись SAML:<br>*Messages*<br>*Assertions*<br>*AuthN requests*<br>*Logout requests*<br>*Logout responses* |
| *Encrypt* | Отметьте флажки, чтобы выбрать сущности, для которых должно быть включено шифрование SAML:<br>*Name ID*<br>*Assertions* |
| *Case-sensitive login* | Снимите флажок, чтобы отключить чувствительный к регистру вход для имен пользователей (по умолчанию включено).<br>Отключение чувствительного к регистру входа позволяет, например, входить как ”admin”, даже если пользователь Zabbix — ”Admin” или ”ADMIN”.<br>Обратите внимание: если чувствительный к регистру вход отключен и существуют несколько пользователей Zabbix с похожими именами (например, Admin и admin), вход для этих пользователей всегда будет отклонен со следующим сообщением об ошибке: ”Authentication failed: supplied credentials are not unique.” |
| *Configure JIT provisioning* | Отметьте этот флажок, чтобы показать параметры, связанные с JIT-подготовкой пользователей. |
| *Group name attribute* | Укажите атрибут имени группы для JIT-подготовки пользователей. |
| *User name attribute* | Укажите атрибут имени пользователя для JIT-подготовки пользователей. |
| *User last name attribute* | Укажите атрибут фамилии пользователя для JIT-подготовки пользователей. |
| *User group mapping* | Сопоставьте шаблон группы пользователей SAML с группой пользователей Zabbix и ролью пользователя.<br>Это необходимо, чтобы определить, какую группу пользователей/роль получит подготовленный пользователь в Zabbix.<br>Нажмите *Add*, чтобы добавить сопоставление.<br>Поле *SAML group pattern* поддерживает подстановочные знаки. Имя группы должно совпадать с существующей группой.<br>Если пользователь SAML соответствует нескольким группам пользователей Zabbix, он станет участником всех этих групп.<br>Если пользователь соответствует нескольким ролям пользователей Zabbix, он получит наивысший уровень прав среди них. |
| *Media type mapping* | Сопоставьте медиа-атрибуты SAML пользователя (например, email) с медиа пользователя Zabbix для отправки уведомлений. |

| Parameter | Description |
|---|---|
| *Enable SCIM provisioning* | Отметьте этот флажок, чтобы включить подготовку SCIM 2.0. |

См. примеры настройки поставщиков удостоверений SAML для входа и подготовки пользователей в Zabbix:

- Microsoft Azure AD
- Okta
- Onelogin

Примечания по SCIM provisioning

Для SCIM provisioning укажите путь к веб-интерфейсу Zabbix и добавьте к нему api_scim.php на стороне поставщика удостоверений, то есть:

`https://<path-to-zabbix-ui>/api_scim.php`

Атрибуты пользователя, которые используются в Zabbix (username, user name, user lastname и media attributes), нужно добавить как пользовательские атрибуты и, при необходимости, внешний namespace должен совпадать со схемой пользователя: `urn:ietf:params:scim:schemas:core:2.0:User`.

Дополнительные настройки

Дополнительные параметры SAML можно настроить в файле конфигурации веб-интерфейса Zabbix (*zabbix.conf.php*):

- $SSO['SP_KEY'] = '<путь к файлу закрытого ключа SP>';
- $SSO['SP_CERT'] = '<путь к файлу сертификата SP>';
- $SSO['IDP_CERT'] = '<путь к файлу сертификата IDP>';
- $SSO['SETTINGS']

**Note:**
Массив `$SSO['SETTINGS']` должен иметь ту же структуру, которую ожидает библиотека *SAML PHP Toolkit* (поставляется вместе с Zabbix). Полное описание доступных параметров конфигурации см. в официальной документации библиотеки.

В составе $SSO['SETTINGS'] можно задать только следующие параметры:

- *strict*
- *baseurl*
- *compress*
- *contactPerson*
- *organization*
- *sp* (только параметры, указанные в этом списке)
  - *attributeConsumingService*
  - *x509certNew*
- *idp* (только параметры, указанные в этом списке)
  - *singleLogoutService* (только один параметр)
    - *responseUrl*
  - *certFingerprint*
  - *certFingerprintAlgorithm*
  - *x509certMulti*
- *security* (только параметры, указанные в этом списке)
  - *signMetadata*
  - *wantNameId*
  - *requestedAuthnContext*
  - *requestedAuthnContextComparison*
  - *wantXMLValidation*
  - *relaxDestinationValidation*
  - *destinationStrictlyMatches*
  - *rejectUnsolicitedResponsesWithInResponseTo*
  - *signatureAlgorithm*
  - *digestAlgorithm*
  - *lowercaseUrlencoding*

Все остальные параметры будут взяты из базы данных и не могут быть переопределены. Параметр *debug* будет проигнорирован.

Кроме того, если веб-интерфейс Zabbix находится за прокси или балансировщиком нагрузки, можно использовать пользовательский параметр *use_proxy_headers*:

- *false* (по умолчанию) — игнорировать параметр;
- *true* — использовать HTTP-заголовки X-Forwarded-* для формирования базового URL.

Если для подключения к экземпляру Zabbix используется балансировщик нагрузки, при этом балансировщик использует TLS/SSL, а Zabbix — нет, необходимо указать параметры 'baseurl', 'strict' и 'use_proxy_headers' следующим образом:

```
$SSO['SETTINGS'] = [
    'strict' => false,
    'baseurl' => 'https://zabbix.example.com/zabbix/',
    'use_proxy_headers' => true
];
```

**Пример конфигурации:**

```
$SSO['SETTINGS'] = [
    'security' => [
        'signatureAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha384'
        'digestAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#sha384',
        // ...
    ],
    // ...
];
```

Настройка веб-интерфейса с Kerberos/ADFS

Файл конфигурации веб-интерфейса Zabbix (*zabbix.conf.php*) можно использовать для настройки SSO с аутентификацией Kerberos и ADFS:

```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => [
            'urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos',
        ],
        'requestedAuthnContextComparison' => 'exact'
    ]
];
```

В этом случае в поле *SP name ID* конфигурации SAML установите:

`urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified`

4 MFA

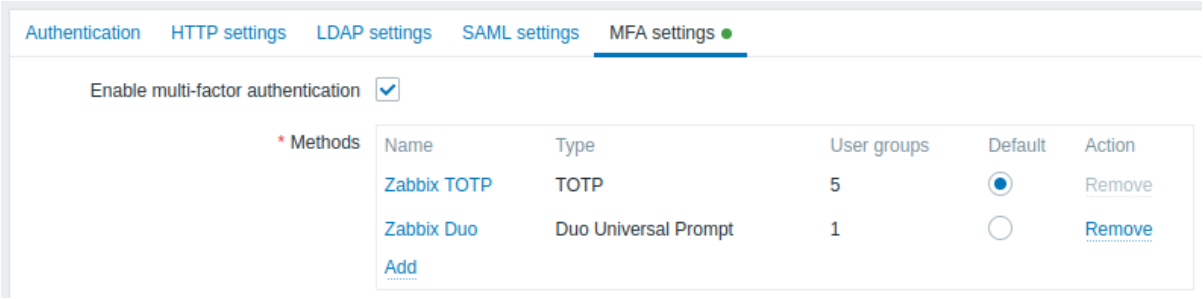
Обзор

Многофакторная аутентификация (MFA) может использоваться для входа в Zabbix, обеспечивая дополнительный уровень безопасности по сравнению с одним только именем пользователя и паролем.

При использовании MFA пользователь должен существовать в Zabbix, должен указать учетные данные Zabbix при входе в систему, а также должен подтвердить свою личность другим способом, обычно с помощью кода, сгенерированного приложением-аутентификатором на телефоне пользователя.

Доступно несколько методов MFA, что позволяет пользователям выбрать вариант, который лучше всего соответствует их требованиям к безопасности и предпочтениям. Эти методы: одноразовый пароль на основе времени (Time-Based One-Time Password, TOTP) и Duo Universal Prompt.

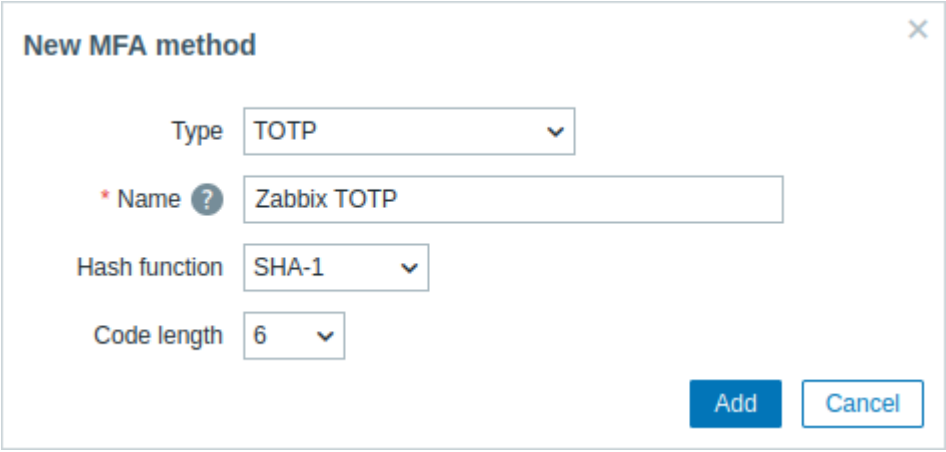
Конфигурация




Параметры конфигурации:

| Parameter | Description |
|---|---|
| *Enable multi-factor authentication* | Установите флажок, чтобы включить многофакторную аутентификацию. |
| *Methods* | Нажмите *Add*, чтобы настроить метод MFA (см. настройка метода ниже). |

Настройка метода




Параметры настройки метода:

| Parameter | Description |
|---|---|
| *Type* | Выберите тип метода MFA:<br>**TOTP** - используйте приложение-аутентификатор для генерации одноразовых паролей на основе времени;<br>**Duo Universal Prompt** - используйте службу аутентификации Duo для обеспечения многофакторной аутентификации. |
| *Name* | Введите имя, которое будет отображаться как имя учетной записи для всех пользователей MFA в приложениях-аутентификаторах (например, ”Zabbix”). |
| *Hash function* | Выберите функцию хеширования (SHA-1, SHA-256 или SHA-512) для генерации кодов TOTP.<br>Этот параметр доступен, если для типа метода MFA установлено значение ”TOTP”.<br>Обратите внимание, что выбор SHA-256 или SHA-512 может значительно ограничить совместимость, поскольку многие приложения в настоящее время не поддерживают эти функции. |
| *Code length* | Выберите длину кода подтверждения (6 или 8).<br>Этот параметр доступен, если для типа метода MFA установлено значение ”TOTP”. |
| *API hostname* | Введите имя хоста API, предоставленное службой аутентификации Duo.<br>Этот параметр доступен, если для типа метода MFA установлено значение ”Duo Universal Prompt”. |
| *Client ID* | Введите идентификатор клиента, предоставленный службой аутентификации Duo.<br>Этот параметр доступен, если для типа метода MFA установлено значение ”Duo Universal Prompt”. |
| *Client secret* | Введите секрет клиента, предоставленный службой аутентификации Duo.<br>Этот параметр доступен, если для типа метода MFA установлено значение ”Duo Universal Prompt”. |

Примеры конфигурации

В этом разделе приведены примеры настройки MFA с использованием Time-Based One-Time Password (TOTP) и Duo Universal Prompt.

TOTP

Для TOTP пользователи должны подтвердить свою личность с помощью приложения-аутентификатора (например, приложения Google Authenticator).

1. Перейдите в настройки MFA в Zabbix в разделе *Users* → *Authentication* и включите многофакторную аутентификацию.

2. Добавьте новый MFA метод со следующей конфигурацией:

   - Type: TOTP
   - Name: Zabbix TOTP
   - Hash function: SHA-1
   - Code length: 6

3. Нажмите кнопку *Add*, а затем кнопку *Update*.

4. Перейдите в *Users* → *User groups* и создайте новую группу пользователей со следующей конфигурацией:

   - Group name: TOTP group
   - Users: Admin
   - Multi-factor authentication: Default (или "Zabbix TOTP", если это не задано как значение по умолчанию)

5. Выйдите из Zabbix и снова войдите, используя свои учетные данные. После успешного входа вам будет предложено зарегистрироваться в MFA, при этом будет отображаться QR-код и секретный ключ.

6. Отсканируйте QR-код или введите секретный ключ в приложение Google Authenticator. Приложение сгенерирует код подтверждения, который нужно ввести для завершения процесса входа.

7. При последующих входах получите код подтверждения в приложении Google Authenticator и введите его при входе.

Duo Universal Prompt

Для Duo Universal Prompt пользователи должны подтверждать свою личность с помощью приложения-аутентификатора Duo Mobile.

**Attention:**
Метод MFA Duo Universal Prompt требует установки расширения php-curl, доступа к Zabbix по HTTPS и разрешения на исходящие подключения к серверам Duo. Кроме того, если вы включили Content Security Policy (CSP) на веб-сервере, обязательно добавьте "duo.com" в директиву CSP в файле конфигурации вашего виртуального хоста.

1. Зарегистрируйте бесплатную учетную запись администратора Duo на Duo Signup.

2. Откройте панель Duo Admin Panel, перейдите в *Applications* → *Protect an Application*, найдите приложение *Web SDK* и нажмите *Protect*.

3. Запишите учетные данные (Client ID, Client secret, API hostname), необходимые для настройки метода MFA в Zabbix.

4. Перейдите в настройки MFA в Zabbix в разделе *Users* → *Authentication* и включите многофакторную аутентификацию.

5. Добавьте новый MFA метод со следующей конфигурацией:

- Type: Duo Universal Prompt
- Name: Zabbix Duo
- API hostname: (use API hostname from Duo)
- Client ID: (use Client ID from Duo)
- Client secret: (use Client secret from Duo)

6. Нажмите кнопку *Add*, а затем кнопку *Update*.

7. Перейдите в *Users* → *User groups* и создайте новую группу пользователей со следующей конфигурацией:

- Group name: Duo group
- Users: Admin
- Multi-factor authentication: Default (or "Zabbix Duo" if it is not set as default)

8. Выйдите из Zabbix и войдите снова, используя свои учетные данные. После успешного входа вам будет предложено зарегистрироваться в MFA, после чего вы будете перенаправлены в Duo. Завершите настройку Duo и подтвердите пользователя с помощью приложения Duo на телефоне, чтобы войти в систему.

9. Для последующих входов используйте соответствующий метод MFA, предоставляемый приложением Duo (например, получение кода подтверждения, ответ на push-уведомления или использование аппаратных ключей), и вводите требуемую информацию во время входа.

## 9. Администрирование

Обзор

Меню Администрирование используется в Zabbix для административных функций. Это меню доступно только пользователям с типом Супер-администратор.

### 1. Общие

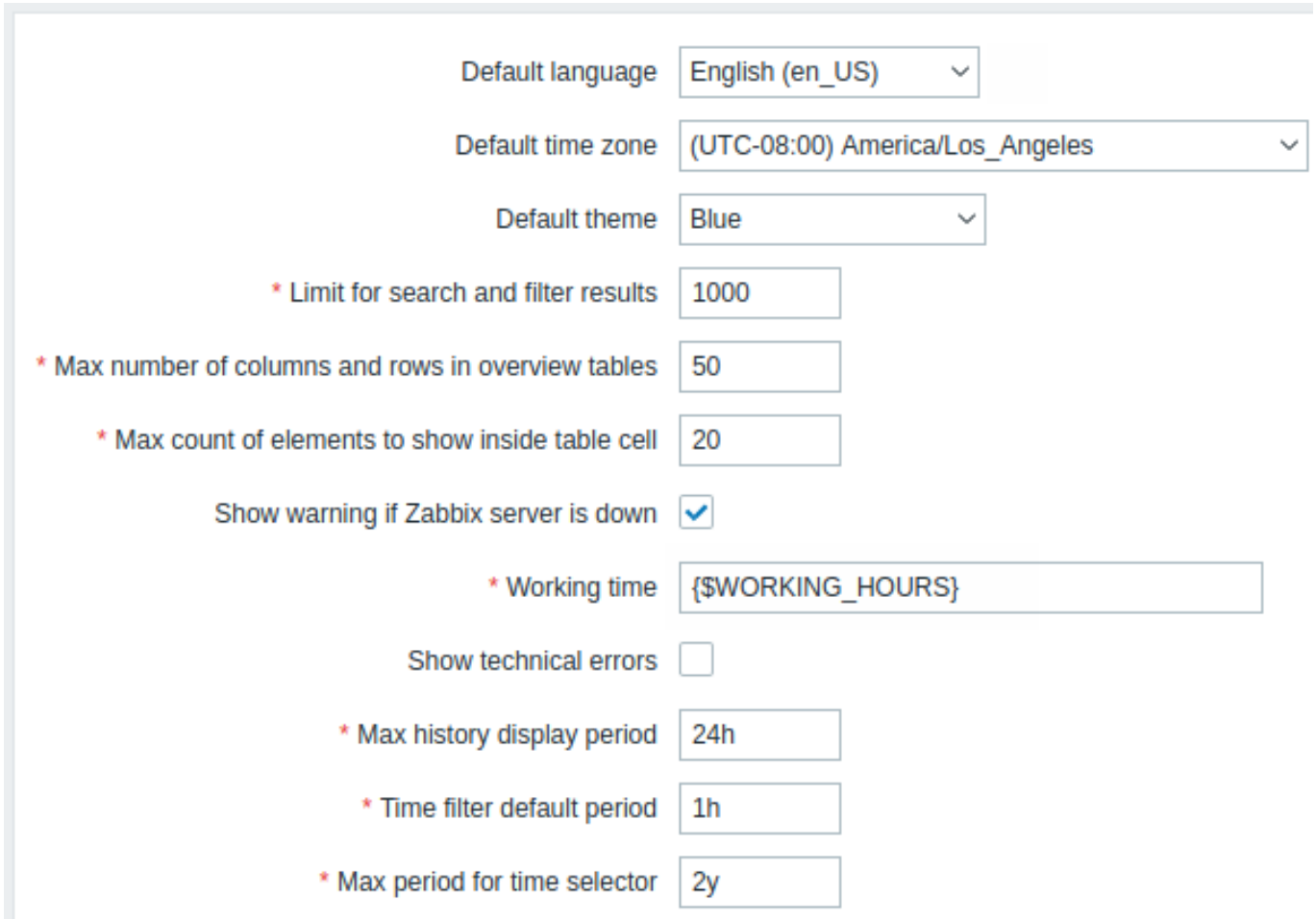
Обзор

Раздел *Administration* > *General* содержит ряд подразделов для настройки значений по умолчанию, связанных с веб-интерфейсом, и для настройки Zabbix.

Список доступных подразделов отображается при нажатии *General* в разделе меню *Administration*. Также можно переключаться между подразделами с помощью выпадающего списка заголовка в левом верхнем углу.

GUI

Этот раздел предоставляет настройку нескольких значений по умолчанию, связанных с веб-интерфейсом.



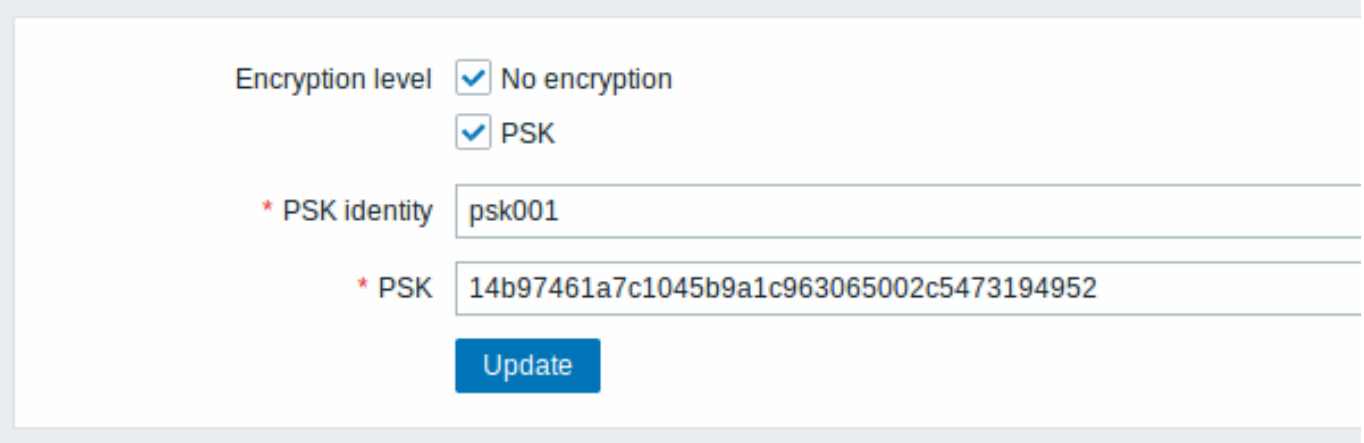

Параметры конфигурации:

| Parameter | Description |
|---|---|
| *Default language* | Язык по умолчанию для пользователей, которые не указали язык в своих профилях, и для гостевых пользователей.<br>Дополнительные сведения см. в разделе Установка дополнительных языков веб-интерфейса. |
| *Default time zone* | Часовой пояс по умолчанию для пользователей, которые не указали часовой пояс в своих профилях, и для гостевых пользователей. |
| *Default theme* | Тема по умолчанию для пользователей, которые не указали тему в своих профилях, и для гостевых пользователей. |
| *Limit for search and filter results* | Максимальное количество элементов (строк), которое будет отображаться в списке веб-интерфейса, например в *Сбор данных* > *Узлы сети*.<br>*Примечание*: если задать, например, значение '50', во всех затронутых списках веб-интерфейса будут отображаться только первые 50 элементов. Если в каком-либо списке содержится более пятидесяти элементов, это будет обозначено знаком '+' в сообщении *"Отображаются элементы с 1 по 50 из найденных **50+**"*. Также, если используется фильтрация и совпадений по-прежнему больше 50, будут отображены только первые 50.<br>Обратите внимание, что увеличение значения этого параметра может привести к снижению производительности и увеличению потребления памяти на стороне веб-сервера. |
| *Max number of columns<br>and rows in overview tables* | Максимальное количество столбцов и строк, отображаемых в виджете панели *Обзор триггеров*. Этот же предел применяется и к столбцам, и к строкам. Если существует больше строк и/или столбцов, чем показано, система выведет внизу таблицы предупреждение: "Не все результаты отображены. Уточните критерии поиска." |
| *Max count of elements<br>to show inside table cell* | Для записей, отображаемых в одной ячейке таблицы, будет показано не более указанного здесь количества. |

| Parameter | Description |
|---|---|
| *Show warning if Zabbix server is down* | Этот параметр включает отображение предупреждающего сообщения в окне браузера, если к серверу Zabbix невозможно подключиться (возможно, он недоступен). Сообщение остается видимым, даже если пользователь прокручивает страницу вниз. При наведении курсора сообщение временно скрывается, чтобы показать содержимое под ним. |
| *Working time* | Этот общесистемный параметр определяет рабочие часы. На графиках рабочее время отображается белым фоном, а нерабочее - серым.<br>Описание формата времени см. на странице Спецификация временного периода.<br>Поддерживаются пользовательские макросы. |
| *Show technical errors* | Если этот параметр включен, все зарегистрированные пользователи смогут видеть технические ошибки (PHP/SQL). Если он выключен, информация доступна только суперадминистраторам Zabbix и пользователям, входящим в группы пользователей с включенным режимом отладки. |
| *Max history display period* | Максимальный период времени, за который отображаются исторические данные в *Мониторинг* > *Последние данные*, сведения о веб-сценарии узла сети в *Мониторинг* > *Узлы сети* и в виджете панели *Топ элементов данных*.<br>Допустимый диапазон: 24 часа (по умолчанию) - 1 неделя. Поддерживаются суффиксы времени, например 1w (одна неделя), 36h (36 часов). |
| *Time filter default period* | Период времени, используемый по умолчанию в графиках и панелях. Допустимый диапазон: 1 минута - 10 лет (по умолчанию: 1 час).<br>Поддерживаются суффиксы времени, например 10m (десять минут), 5w (пять недель).<br>Примечание: когда пользователь изменяет период времени при просмотре графика, этот период сохраняется как пользовательская настройка, заменяя глобальное значение по умолчанию или предыдущий выбор пользователя. |
| *Max period for time selector* | Максимально доступный период времени для графиков и панелей. Пользователи не смогут просматривать более старые данные. Допустимый диапазон: 1 год - 10 лет (по умолчанию: 2 года).<br>Поддерживаются суффиксы времени, например 1y (один год), 365w (365 недель). |

Автосоздание узлов сети

В этом разделе можно настроить уровень шифрования для автосоздания узлов сети активным агентом.




Параметры, отмеченные звездочкой, обязательны.

Параметры конфигурации:

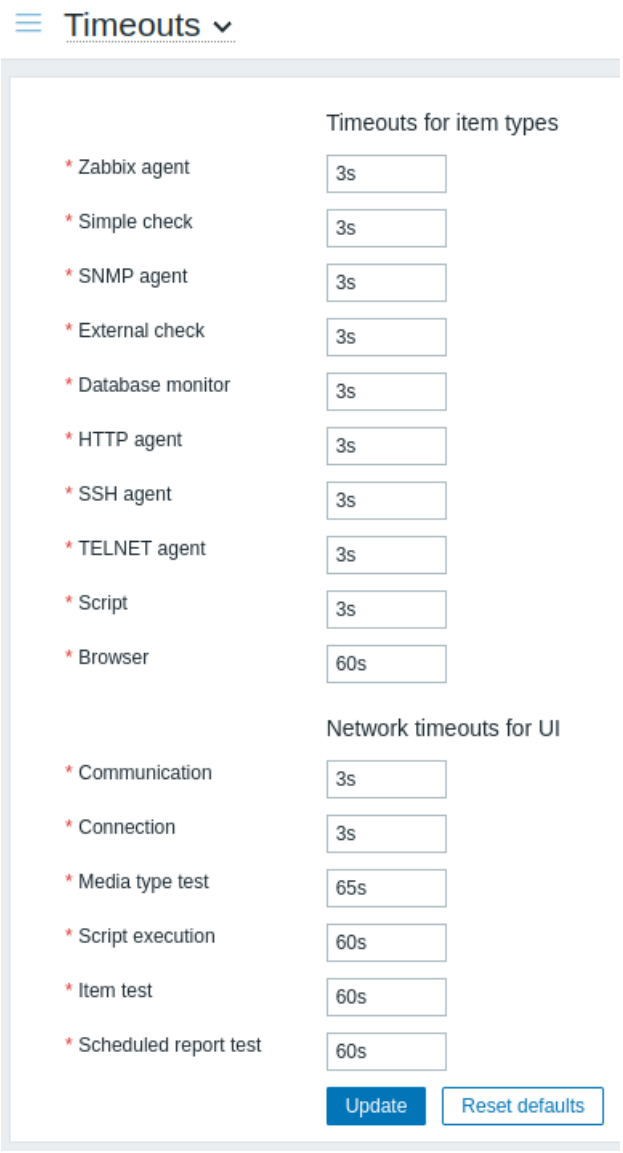
| Parameter | Description |
|---|---|
| *Уровень шифрования* | Выберите один или оба варианта уровня шифрования:<br>**Без шифрования** - разрешены незашифрованные соединения<br>**PSK** - разрешены TLS-зашифрованные соединения с предварительно общим ключом |
| *Идентификатор PSK* | Введите строку идентификатора предварительно общего ключа.<br>Это поле доступно только если в качестве *Уровень шифрования* выбран вариант 'PSK'.<br>Не указывайте конфиденциальную информацию в идентификаторе PSK, так как она передается по сети без шифрования, чтобы сообщить получателю, какой PSK использовать. |

| Parameter | Description |
|---|---|
| *PSK* | Введите предварительно общий ключ (четное количество шестнадцатеричных символов).<br>Максимальная длина: 512 шестнадцатеричных цифр (PSK размером 256 байт), если Zabbix использует библиотеку GnuTLS или OpenSSL, 64 шестнадцатеричные цифры (PSK размером 32 байта), если Zabbix использует библиотеку mbed TLS (PolarSSL).<br>Пример: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952<br>Это поле доступно только если в качестве *Уровень шифрования* выбран вариант 'PSK'. |

См. также: Безопасное автосоздание узлов сети

Таймауты

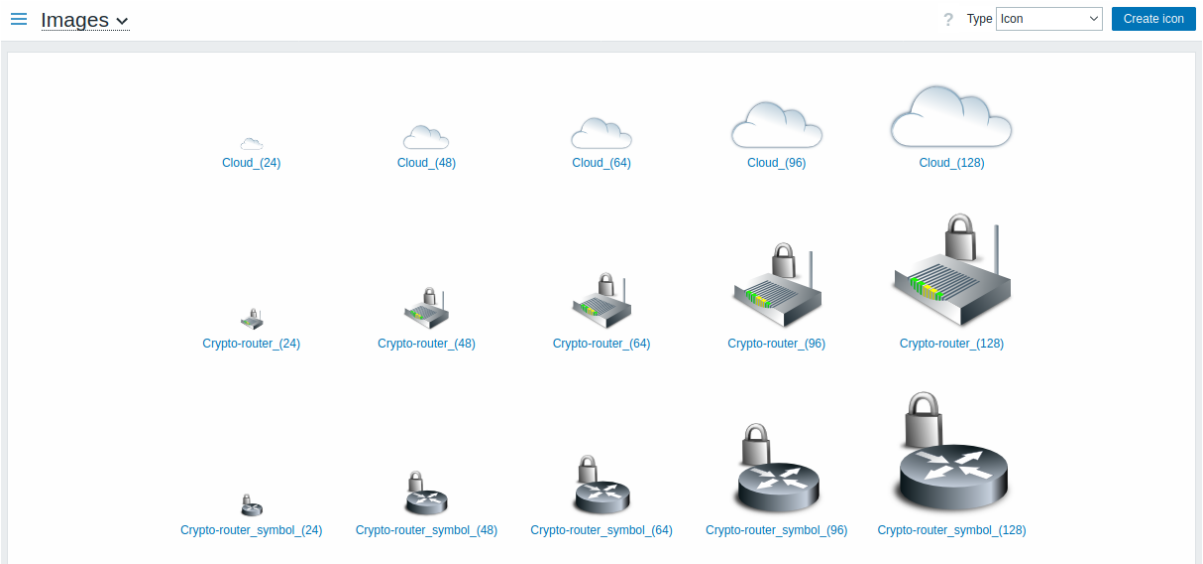
В этом разделе можно задать глобальные таймауты для типов элементов данных и сетевые таймауты. Все поля в этой форме обязательны.



| Параметр | | Описание |
|---|---|---|
| *Таймауты для типов элементов данных* | | Сколько секунд ждать ответа от контролируемого элемента данных (в зависимости от его типа).<br>Допустимый диапазон: 1 - 600 с (по умолчанию: 3 с; по умолчанию для типа элемента данных Browser: 60 с).<br>Поддерживаются суффиксы времени, например 30s, 1m, а также пользовательские макросы.<br><br>Поддерживаемые типы элементов данных:<br>- агент Zabbix (как пассивные, так и активные проверки)<br>- Simple check (кроме элементов данных `icmpping*`, `vmware.*`)<br>- SNMP agent (только для элементов данных SNMP `walk[OID]` и `get[OID]`)<br>- External check<br>- Database monitor<br>- HTTP agent<br>- SSH agent<br>- TELNET agent<br>- Script<br>- Browser<br><br>Обратите внимание: если используется прокси и для него настроены таймауты, параметры таймаутов прокси будут иметь приоритет над глобальными. Если для отдельных элементов данных заданы таймауты, они будут иметь приоритет над настройками прокси и глобальными настройками. |
| *Сетевые таймауты для UI* | | |
| | *Обмен данными* | Сколько секунд ждать перед закрытием неактивного сокета (если соединение с сервером Zabbix уже было установлено, но веб-интерфейс не может завершить операцию чтения/отправки данных в течение этого времени, соединение будет разорвано). Допустимый диапазон: 1 - 300 с (по умолчанию: 3 с). |
| | *Соединение* | Сколько секунд ждать перед прекращением попытки подключения к серверу Zabbix. Допустимый диапазон: 1 - 30 с (по умолчанию: 3 с). |
| | *Тест типа медиа* | Сколько секунд ждать ответа при тестировании типа медиа. Допустимый диапазон: 1 - 300 с (по умолчанию: 65 с). |
| | *Выполнение скрипта* | Сколько секунд ждать ответа при выполнении скрипта. Допустимый диапазон: 1 - 300 с (по умолчанию: 60 с).<br>Этот таймаут действует на всю цепочку выполнения скрипта, длина которой может быть различной. Например, если скрипт выполняется на агенте, это означает полный цикл через сервер (возможно, также через прокси) к агенту и обратно.<br>Для скриптов, выполняемых на агенте, применяется таймаут сервера или прокси.<br>Для скриптов, выполняемых только на активном агенте, вероятно, потребуется увеличить значение таймаута сервера/прокси по умолчанию. Таймаут сервера/прокси должен быть больше частоты обновления активной проверки, иначе таймаут сервера/прокси может быть превышен еще до того, как активный агент получит скрипт.<br>См. также: Таймаут скрипта. |
| | *Тест элемента данных* | Сколько секунд ждать возвращаемых данных при тестировании элемента данных. Допустимый диапазон: 1 - 600 с (по умолчанию: 60 с). |
| | *Тест запланированного отчета* | Сколько секунд ждать возвращаемых данных при тестировании запланированного отчета. Допустимый диапазон: 1 - 300 с (по умолчанию: 60 с). |

Изображения

Раздел Images отображает все изображения, доступные в Zabbix. Изображения хранятся в базе данных.

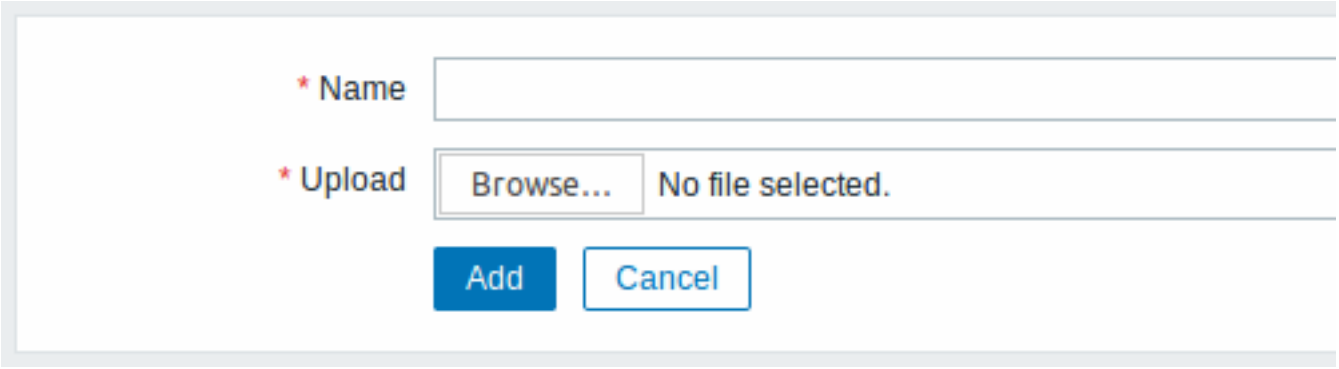



Выпадающий список *Type* позволяет переключаться между значками и фоновыми изображениями:

- Значки используются для отображения элементов карты сети.
- Фоновые изображения используются в качестве фоновых изображений карт сети.

**Добавление изображения**

Вы можете добавить собственное изображение, нажав кнопку *Create icon* или *Create background* в правом верхнем углу.




Атрибуты изображения:

| Parameter | Description |
|---|---|
| *Name* | Уникальное имя изображения. |
| *Upload* | Выберите файл (PNG, JPEG, GIF или WebP) в локальной системе для загрузки в Zabbix. *Note* что может быть возможно загрузить и другие форматы, которые будут преобразованы в PNG во время загрузки. Для обработки изображений используется библиотека GD, поэтому поддерживаемые форматы зависят от версии используемой библиотеки (требуется версия 2.0.28 или выше). |

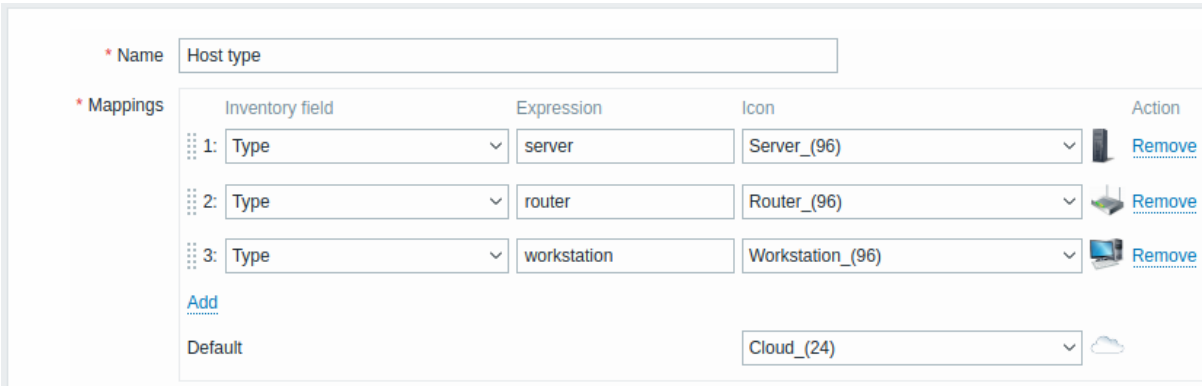
**Note:**
Максимальный размер загружаемого файла ограничен значением ZBX_MAX_IMAGE_SIZE, которое составляет 1024x1024 байта или 1 MB.
Загрузка изображения может завершиться неудачей, если размер изображения близок к 1 MB, а параметр конфигурации MySQL `max_allowed_packet` имеет значение по умолчанию 1MB. В этом случае увеличьте параметр max_allowed_packet.

Сопоставление значков

Этот раздел позволяет создавать сопоставление определенных узлов сети с определенными значками. Для создания сопоставления используется информация из полей инвентаризации узла сети.

Затем эти сопоставления можно использовать в настройке карты сети для автоматического назначения соответствующих значков совпадающим узлам сети.

Чтобы создать новую карту значков, нажмите *Create icon map* в правом верхнем углу.




Параметры конфигурации:

| Параметр | Описание |
|---|---|
| *Name* | Уникальное имя карты значков. |
| *Mappings* | Список сопоставлений. Порядок сопоставлений определяет, какое из них будет иметь приоритет. Вы можете перемещать сопоставления вверх и вниз по списку перетаскиванием. |
| *Inventory field* | Поле инвентаризации узла сети, в котором будет выполняться поиск совпадения. |
| *Expression* | Регулярное выражение, описывающее совпадение. |
| *Icon* | Значок, который будет использоваться, если найдено совпадение с выражением. |
| *Default* | Значок, используемый по умолчанию. |

Регулярные выражения

Этот раздел позволяет создавать пользовательские регулярные выражения, которые можно использовать в нескольких местах веб-интерфейса. Подробности см. в разделе Регулярные выражения.

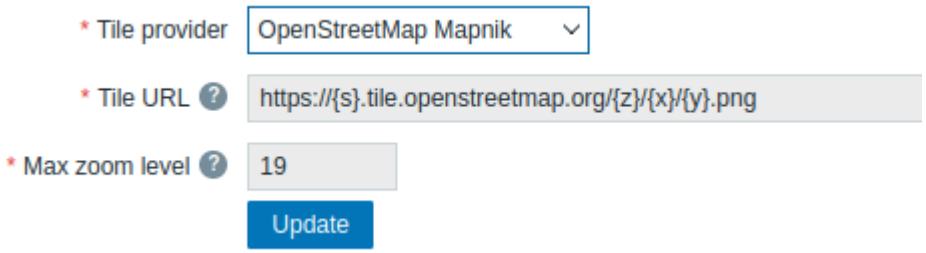
Параметры отображения триггеров

Этот раздел позволяет настраивать, как статус триггера отображается на веб-интерфейсе, а также имена и цвета серьезности триггера.

| Parameter | Description |
|---|---|
| *Использовать пользовательские цвета статуса события* | При включении этого параметра настраиваются цвета для подтвержденных/неподтвержденных проблем. |
| *Неподтвержденные события PROBLEM*, *Подтвержденные события PROBLEM*, *Неподтвержденные события RESOLVED*, *Подтвержденные события RESOLVED* | Введите новый код цвета или щелкните по цвету, чтобы выбрать новый из доступной палитры.<br>Если установлен флажок *мигание*, триггеры будут мигать некоторое время после изменения статуса, чтобы стать более заметными. |
| *Показывать OK-триггеры за* | Период времени для отображения OK-триггеров. Допустимый диапазон: 0 - 24 часа. Поддерживаются суффиксы времени, например 5m, 2h, 1d. |
| *При изменении статуса триггеры мигают в течение* | Длительность мигания триггера. Допустимый диапазон: 0 - 24 часа. Поддерживаются суффиксы времени, например 5m, 2h, 1d. |
| *Не классифицировано*, *Информация*, *Предупреждение*, *Средняя*, *Высокая*, *Катастрофа* | Пользовательские имена и/или цвета серьезности, отображаемые вместо значений по умолчанию системы.<br>Введите новый код цвета или щелкните по цвету, чтобы выбрать новый из доступной палитры.<br><br>Обратите внимание, что пользовательские имена серьезности, введенные здесь, будут использоваться во всех локалях. Если вам нужно перевести их на другие языки для определенных пользователей, см. страницу Настройка серьезностей триггеров. |

Географические карты

Этот раздел позволяет выбрать поставщика сервиса тайлов географической карты и настроить параметры поставщика для виджета панели Geomap. Для визуализации с использованием географических карт Zabbix использует библиотеку интерактивных карт с открытым исходным кодом Leaflet на JavaScript. Обратите внимание, что Zabbix не контролирует качество изображений, предоставляемых сторонними поставщиками тайлов, включая предопределенных поставщиков тайлов.




| Parameter | Description |
|---|---|
| Tile provider | Выберите одного из доступных поставщиков сервиса тайлов или выберите *Other*, чтобы добавить другого поставщика тайлов или самостоятельно размещенные тайлы (см. Использование пользовательского поставщика сервиса тайлов). |
| Tile URL | Шаблон URL (до 2048 символов) для загрузки и отображения слоя тайлов на географических картах.<br>Это поле доступно для редактирования только если для *Tile provider* выбрано *Other*.<br><br>Поддерживаются следующие заполнители:<br>*{s}* представляет один из доступных поддоменов;<br>*{z}* представляет параметр уровня масштабирования в URL;<br>*{x}* и *{y}* представляют координаты тайла;<br>*{r}* можно использовать, чтобы добавить "@2x" к URL для загрузки тайлов retina.<br><br>Пример: `https://{s}.example.com/{z}/{x}/{y}{r}.png` |
| Attribution text | Текст атрибуции поставщика тайлов, который будет отображаться в небольшом текстовом поле на карте. Это поле видно только если для *Tile provider* выбрано *Other*. |

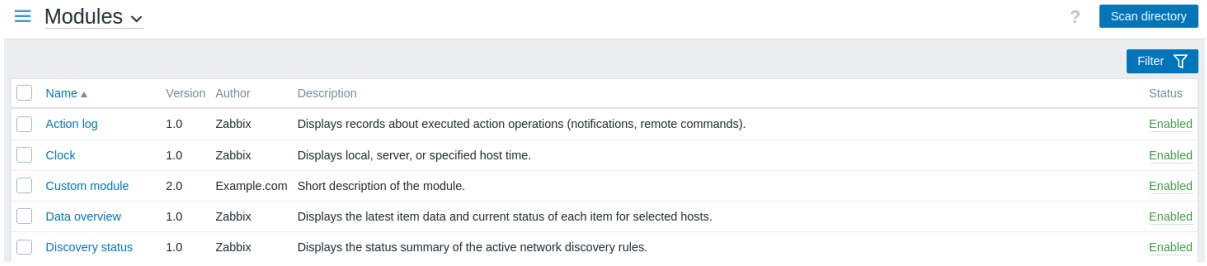
| Parameter | Description |
|---|---|
| Max zoom level | Максимальный уровень масштабирования карты. Это поле доступно для редактирования только если для *Tile provider* выбрано *Other*. |

Использование пользовательского поставщика тайлов

Виджет Geomap может загружать растровые изображения тайлов из пользовательского сервиса поставщика тайлов, размещенного самостоятельно, или из стороннего сервиса поставщика тайлов. Чтобы использовать пользовательский сторонний сервис поставщика тайлов либо локальную папку с тайлами или сервер, выберите *Other* в поле *Tile provider* и укажите пользовательский URL в поле *Tile URL*, используя соответствующие заполнители.

Модули

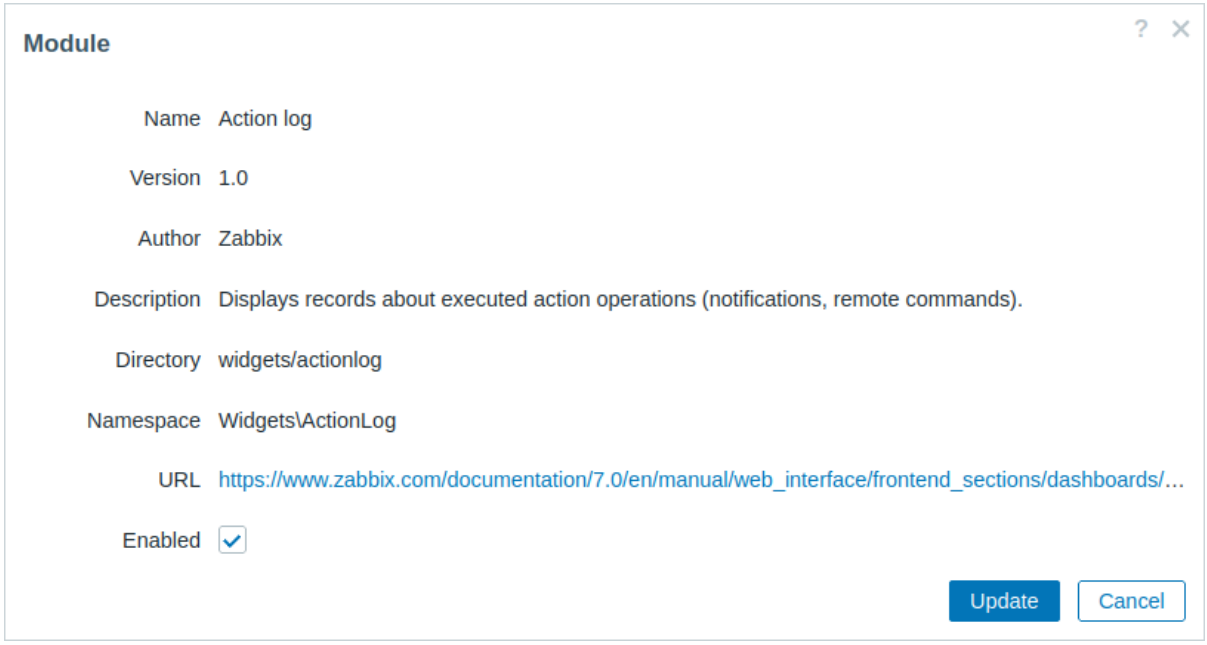
Этот раздел позволяет управлять как пользовательскими, так и встроенными модулями веб-интерфейса.




Нажмите *Scan directory*, чтобы зарегистрировать или отменить регистрацию любых пользовательских модулей. Зарегистрированные модули будут отображаться в списке; незарегистрированные модули будут удалены из списка.

Нажмите на статус модуля в списке, чтобы включить или отключить модуль. Вы также можете массово включать или отключать модули, выбрав их в списке, а затем нажав кнопки *Enable/Disable* под списком.

Нажмите на имя модуля в списке, чтобы просмотреть его сведения во всплывающем окне.

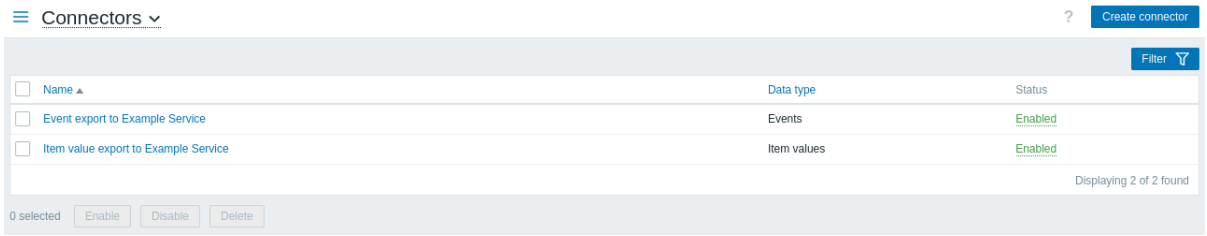



Статус модуля также можно изменить во всплывающем окне со сведениями о модуле; для этого установите или снимите флажок *Enabled*, а затем нажмите *Update*.

Вы можете фильтровать модули по имени или статусу (включен/отключен).

Коннекторы

Этот раздел позволяет настраивать коннекторы для потоковой передачи данных Zabbix во внешние системы по HTTP.




Нажмите *Создать коннектор*, чтобы настроить новый коннектор.

Вы можете фильтровать коннекторы по имени или статусу (включен/выключен). Нажмите на статус коннектора в списке, чтобы включить или выключить коннектор. Вы также можете массово включать или выключать коннекторы, выбрав их в списке, а затем нажав кнопки *Включить/Выключить* под списком.

Другое

Этот раздел позволяет настроить прочие параметры веб-интерфейса.

| Parameter | Description |
|---|---|
| *Frontend URL* | URL (до 2048 символов) для веб-интерфейса Zabbix.<br>Этот параметр используется веб-службой Zabbix для связи с веб-интерфейсом и должен быть указан, чтобы включить запланированные отчеты. |
| *Group for discovered hosts* | Узлы сети, обнаруженные с помощью network discovery и agent autoregistration, будут автоматически помещены в группу узлов сети, выбранную здесь. |

| Parameter | Description |
|---|---|
| *Default host inventory mode* | Режим mode инвентаризации узла сети по умолчанию. Он будет использоваться каждый раз при создании нового узла сети или прототипа узла сети сервером или веб-интерфейсом, если только он не будет переопределен во время обнаружения узла сети/авторегистрации с помощью операции *Set host inventory mode*. |
| *User group for database down message* | Группа пользователей для отправки сообщения тревоги о недоступности базы данных.<br>Сервер Zabbix зависит от доступности внутренней базы данных. Он не может работать без базы данных. Если база данных недоступна, Zabbix может уведомить выбранных пользователей. Уведомления будут отправляться в указанную здесь группу пользователей с использованием включенных элементов пользовательских медиа. Уведомления передаются с использованием следующих media types (если включены): email; SMS; пользовательские скрипты оповещения. Даже если элемент медиа типа вебхук настроен и включен, он не будет вызывать уведомления.<br>Сервер Zabbix не остановится; он будет ждать восстановления базы данных, чтобы продолжить обработку.<br>Уведомление состоит из следующего содержимого:<br>`[MySQL\|PostgreSQL] database <DB Name> [on <DB Host>:<DB Port>] is not available: <error message depending on the type of DBMS (database)>`<br><DB Host> не добавляется в сообщение, если для него задано пустое значение, а <DB Port> не добавляется, если используется значение по умолчанию (”0”). Менеджер оповещений (специальный процесс сервера Zabbix) пытается установить новое соединение с базой данных каждые 10 секунд. Если база данных по-прежнему недоступна, менеджер оповещений повторяет отправку оповещений, но не чаще чем раз в 15 минут. |
| *Log unmatched SNMP traps* | Записывать в журнал SNMP trap, если не найдено соответствующих SNMP-интерфейсов. |

Авторизация

| Параметр | Описание |
|---|---|
| *Попытки входа (Login attempts)* | Количество неуспешных попыток входа, прежде чем возможность входа будет заблокирована. |
| *Интервал блокировки входа в систему (Login blocking interval)* | Период времени, в течение которого вход в систему будет запрещён при превышении лимита *попыток входа*. Допустимый диапазон: 30 - 3600 секунд. Поддерживаются суффиксы времени, например: 90s, 5m, 1h. |

Хранение секретов

| Parameter | Description |
|---|---|
| *Vault provider* | Выберите программное обеспечение для управления секретами для хранения значений пользовательских макросов - *HashiCorp Vault* (по умолчанию) или *CyberArk Vault*. |
| *Resolve secret vault macros by* | Разрешать макросы хранилища секретов с помощью:<br>**сервер Zabbix** - секреты извлекаются сервером Zabbix и при необходимости пересылаются прокси (по умолчанию);<br>**сервер Zabbix и прокси** - секреты извлекаются как сервером Zabbix, так и прокси, что позволяет им разрешать макросы независимо. |

См. также: Хранение секретов.

Безопасность

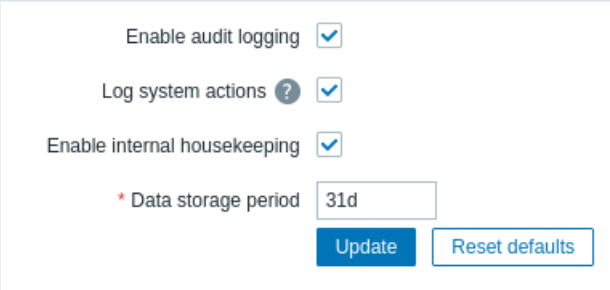
| Параметр | Описание |
|---|---|
| *Валидация схем URI (Validate URI schemes)* | Снимите флажок, чтобы отключить проверку схемы URI (по умолчанию включено).<br>Если включено, можно указать список (через запятую) разрешённых схем URI (по умолчанию: http,https,ftp,file,mailto,tel,ssh). Применяется ко всем полям в веб-интерфейсе, где используются URI (например, URL’ы элементов карт сети). |

| Параметр | Описание |
|---|---|
| *HTTP заголовок X-Frame-Options (Use X-Frame-Options HTTP header)* | Снимите данный флажок, чтобы отключить заголовок X-Frame-options (не рекомендуется).<br>Если отмечено, вы можете указать значение HTTP заголовка X-Frame-options.<br>Поддерживаемые значения:<br>**SAMEORIGIN** (по умолчанию) или **'self'** (должно быть обрамлено апострофами) — страница может отображаться только внутри фрейма с тем же источником, как и сама страница;<br>**DENY** или **'none'** (должно быть обрамлено апострофами) — страница не может отображаться во фрейме, независимо от сайта, пытающегося это сделать;<br>**строка имён хостов через пробел**; добавление **'self'** (должно быть обрамлено апострофами) к списку позволяет странице отображаться во фрейме с тем же источником, как и сама страница.<br>Обратите внимание, что использование **'self'** или **'none'** без апострофов приведёт к тому, что они будут рассматриваться как имена хостов. |
| *Использование атрибута sandbox в iframe (Use iframe sandboxing)* | Снимите данный флажок, чтобы не помещать полученное по URL содержимое в песочницу (не рекомендуется).<br>Если отмечено, вы можете указать исключения песочницы iframe; неуказанные ограничения по-прежнему будут применяться. Если это поле пустое, применяются все ограничения атрибута sandbox.<br>Смотрите описание атрибута `sandbox` [en]для более подробной информации. |

### 2. Журнал аудита

Обзор

В этом разделе можно настроить параметры журнала аудита.




Доступны следующие параметры:

| Параметр | Описание |
|---|---|
| *Активировать журналирование аудита (Enable audit logging)* | Включить (по умолчанию) или отключить ведение журнала аудита. |
| *Журналирование действий системы (Log system actions)* | Включить (по умолчанию) или отключить ведение журнала аудита низкоуровневого обнаружения, сетевого обнаружения и действий авторегистрации, выполняемых сервером (пользователем System). |
| *Активировать внутреннюю очистку истории (Enable internal housekeeping)* | Включить (по умолчанию) или отключить внутреннюю очистку истории для записей журнала аудита. |
| *Период хранения данных (Data storage period)* | Количество дней, в течение которых аудитные записи должны храниться, прежде чем они будут удалены процессом очистки истории.<br>Обязательно, если функция очистки включена.<br>По умолчанию: 31 день. |

Параметры сжатия аудитных записей доступны в разделе *Администрирование → Очистка истории (Administration → Housekeeping)*, в блоке *Сжатие истории, тенденций и журнала аудита (History, trends and audit log compression)*, который становится видимым при использовании TimescaleDB.

### 3. Очистка истории

Обзор

Раздел *Administration* > *Housekeeping* позволяет настроить внутреннюю очистку Zabbix, которая автоматически удаляет устаревшие данные или данные, удаленные пользователем, из базы данных. Это предотвращает бесконечный рост базы данных, который приводил бы к ненужному расходу дискового пространства и снижению производительности базы данных.

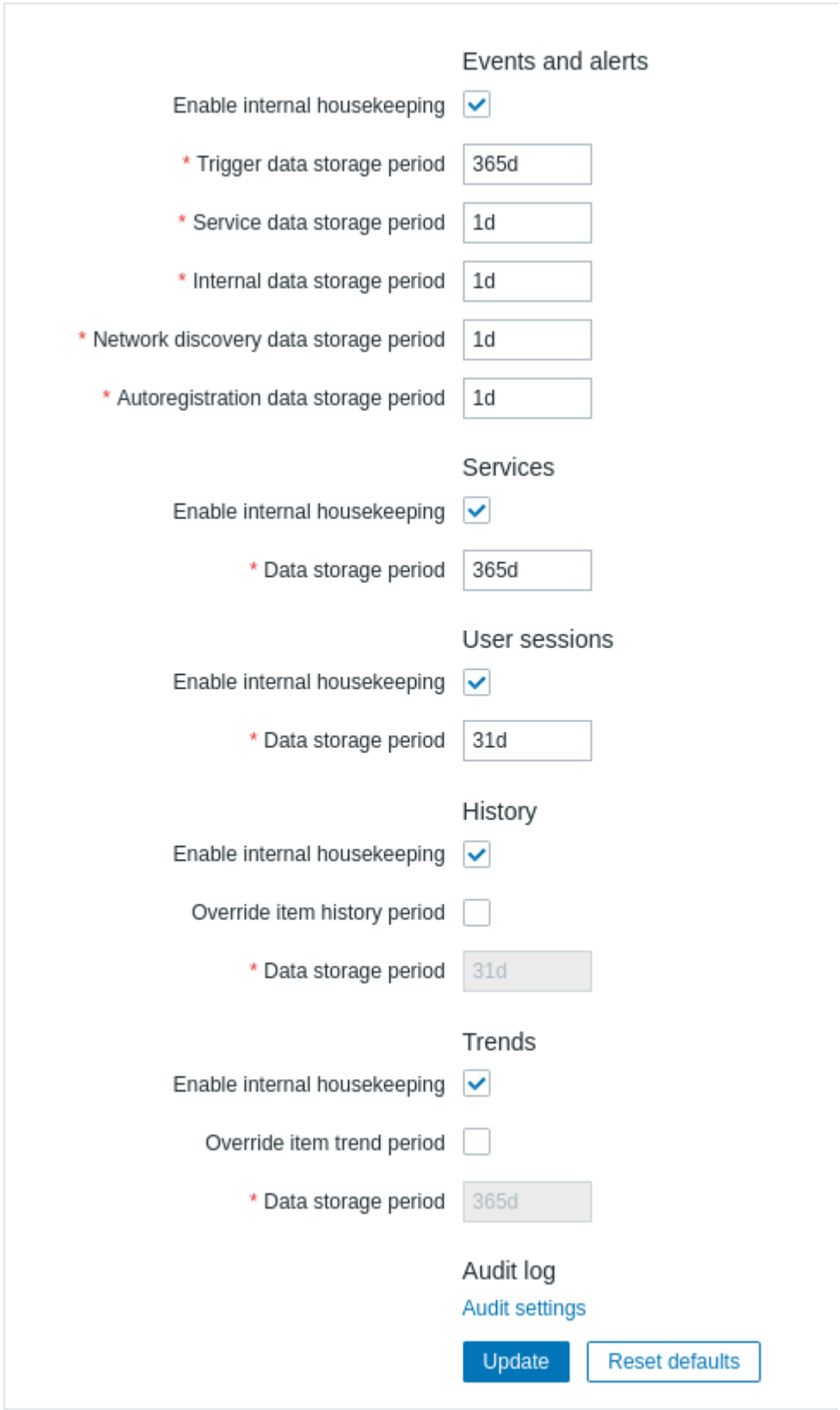
Очистку можно включить и настроить для следующих типов данных:

- События и оповещения от триггеров, служб, внутренних данных, сетевого обнаружения и авто-регистрации
- Службы
- Сеансы пользователей
- История и тренды для элементов данных

> **Note:**
> Очистка журнала аудита настраивается отдельно.

Конфигурация

Доступны следующие параметры:




| Parameter | Description |
|---|---|
| *Enable internal housekeeping* | Включить или отключить внутреннюю очистку (по умолчанию включена).<br>Если включено, процесс housekeeper сервера Zabbix автоматически удаляет из базы данных данные, срок хранения которых превышает *Data storage period*. |

| Parameter | Description |
|---|---|
| *Data storage period* | Укажите, как долго хранить данные до их удаления housekeeper.<br>Обязательно, если включена внутренняя очистка.<br>Диапазон: 1 day (1 hour for history) - 25 years; или ”0”. Поддерживаются суффиксы времени (например, 1d, 1w).<br><br>Для *Events and alerts* срок хранения данных задается отдельно для триггеров, services, внутренних данных, network discovery и autoregistration.<br>Housekeeper удаляет только те события, которые не связаны с проблемами. Например, событие problem/recovery старше *Data storage period* не будет удалено, если оно связано с нерешенной проблемой. Когда housekeeper удаляет устаревшие сущности, он сначала удаляет проблемы, затем события.<br>Обратите внимание, что при удалении элемента данных/триггера его проблемы также удаляются; однако связанные события problem/recovery остаются в базе данных, пока housekeeper не удалит их. |
| *Override item history period* | Для *History* и *Trends* сроки хранения данных также определяют, как долго данные остаются видимыми в *Monitoring* > *Latest data*, даже если внутренняя очистка отключена.<br>Если включено, срок хранения history/trend, указанный в конфигурации элемента данных, переопределяется параметром *Data storage period* (за исключением элементов данных с включенной опцией *Do not store*).<br><br>Эту опцию можно использовать даже при отключенной внутренней очистке, если используется внешний housekeeper. |
| *Override item trend period* | |

Если используется **TimescaleDB**, становится доступен раздел *History, trends and audit log compression*.

**Note:**
Для TimescaleDB включите *Override item history period*, *Override item trend period* и *Enable internal housekeeping* для history и trends, чтобы в полной мере использовать автоматическое разбиение на разделы. Если эти параметры отключены, данные, хранящиеся в таблицах history и trends, по-прежнему будут разбиваться на разделы, но housekeeper не будет удалять устаревшие разделы, и будут отображаться предупреждения о конфигурации. Когда удаление устаревших разделов включено, сервер Zabbix и веб-интерфейс больше не будут отслеживать удаленные элементы данных, а history для этих элементов будет очищена при удалении устаревшего раздела.

Кнопка *Reset defaults* позволяет отменить все внесенные изменения.

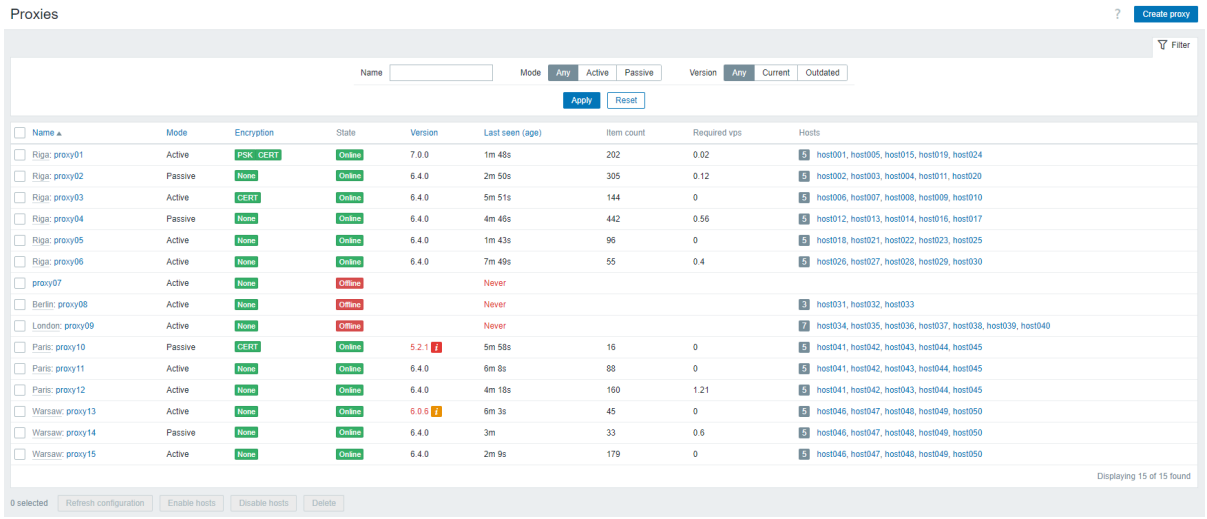
#### 4. Прокси

Обзор

В разделе *Администрирование → Прокси (Administration → Proxies)* можно в веб-интерфейсе Zabbix настроить прокси-серверы для распределённого мониторинга.

Прокси

Отображается список существующих прокси с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя прокси.<br>Щелчок по имени прокси открывает форму конфигурации.<br>Если прокси принадлежит группе прокси, имя группы отображается перед именем прокси в виде серой ссылки. Щелчок по имени группы открывает форму конфигурации. |
| *Mode* | Режим прокси - *Active* или *Passive*. |
| *Encryption* | Состояние шифрования для соединений от прокси:<br>**None** - без шифрования;<br>**PSK** - с использованием предварительно общего ключа;<br>**Cert** - с использованием сертификата. |
| *State* | Состояние прокси:<br>**Unknown** - прокси был создан, когда сервер Zabbix был недоступен, или сервер еще не обновил состояние;<br>**Online** - прокси связывался с сервером Zabbix (пассивный прокси ответил на запрос сервера; активный прокси отправил запрос) в течение периода отказоустойчивости;<br>**Offline** - прокси не связывался с сервером Zabbix в течение периода отказоустойчивости. |
| *Version* | Версия прокси (трехзначный номер версии). Если прокси устарел или не поддерживается, номер версии выделяется (красным), а также отображается значок информационного статуса (желтый или красный). Наведите указатель на значок для получения подробностей. |
| *Last seen (age)* | Время, когда сервер в последний раз видел прокси. |
| *Item count* | Количество включенных элементов данных на включенных узлах сети, назначенных прокси. |
| *Required vps* | Требуемая производительность прокси (количество значений, которые необходимо собирать в секунду). |
| *Hosts* | Количество включенных узлов сети, назначенных прокси, и список узлов сети, контролируемых прокси.<br>Щелчок по имени узла сети открывает форму конфигурации узла сети. |

Чтобы настроить новый прокси, нажмите кнопку *Create proxy* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

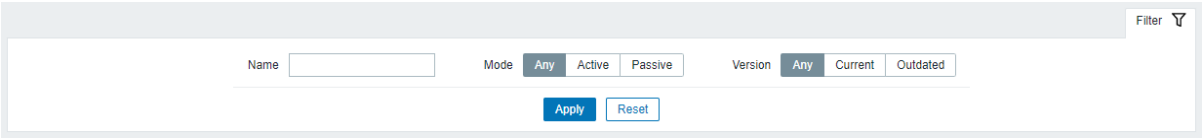
- *Обновить конфигурацию* - обновить конфигурацию прокси;
- *Включить узлы сети* - изменить статус узлов сети, контролируемых прокси, на *Контролируется*;
- *Отключить узлы сети* - изменить статус узлов сети, контролируемых прокси, на *Не контролируется*;
- *Удалить* - удалить прокси.

Чтобы использовать эти параметры, отметьте флажки перед соответствующими прокси, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только интересующие вас прокси. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

Ссылка *Filter* доступна над списком прокси. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать прокси по имени, режиму и версии. Обратите внимание, что параметр фильтра *Outdated* отображает как устаревшие (частично поддерживаемые), так и неподдерживаемые прокси.




### 5. Группы прокси

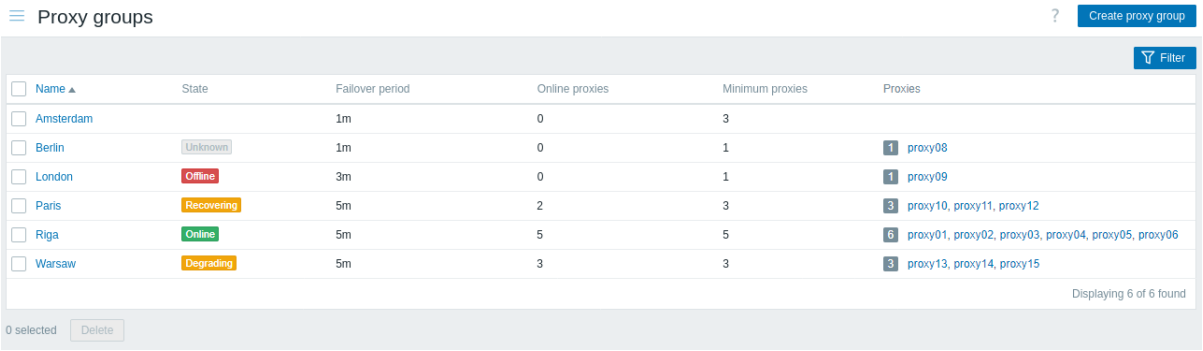
Обзор

Группы прокси можно настроить в разделе *Администрирование → Группы прокси (Administration → Proxy groups)*.

Группы прокси используются при балансировке нагрузки прокси-серверов с автоматическим распределением узлов сети между прокси-серверами и отказоустойчивостью между прокси-серверами.

Группы прокси

Отображается список существующих групп прокси с их подробностями.




Отображаемые данные:

| Column | Description |
|---|---|
| *Name* | Имя группы прокси. Щелчок по имени группы прокси открывает форму настройки группы прокси. |
| *State* | Состояние группы прокси (отображается, если группа содержит хотя бы один прокси):<br>**Unknown** - группа была создана, когда сервер Zabbix был недоступен, или сервер еще не обновил состояние;<br>**Online** - минимальное число прокси связалось с сервером Zabbix в течение периода failover;<br>**Degrading** - группа вскоре перейдет в состояние offline, поскольку число online-прокси опустилось ниже порога *Minimum proxies*;<br>**Offline** - с сервером Zabbix в течение периода failover связалось меньше прокси, чем минимально требуется;<br>**Recovering** - группа вскоре снова перейдет в состояние online, поскольку число online-прокси достигло порога *Minimum proxies*.<br>Состояние группы прокси можно отслеживать с помощью элементов данных zabbix[proxy group,<name>,state] и zabbix[proxy group,discovery]. |
| *Failover period* | Период, в течение которого прокси в группе прокси должен связаться с сервером Zabbix, чтобы считаться online. |
| *Online proxies* | Количество online-прокси (отображается красным, если значение ниже минимального для группы). |
| *Minimum proxies* | Минимальное количество online-прокси, необходимое для поддержания группы прокси в состоянии online. |
| *Proxies* | Количество прокси в группе и список прокси в группе со ссылками на форму настройки прокси.<br>Максимальное число отображаемых прокси ограничено значением *Max count of elements to show inside table cell*. |

Чтобы настроить новую группу прокси, нажмите кнопку *Create proxy groups* в правом верхнем углу.

Параметры массового редактирования

Кнопки под списком предлагают несколько вариантов массового редактирования:

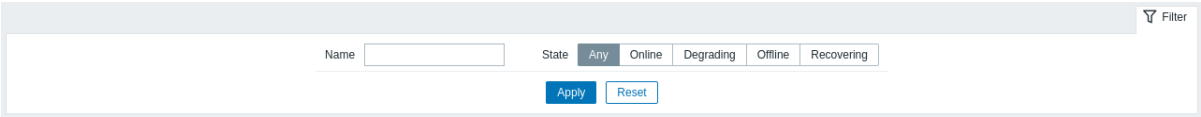
- *Удалить* - удалить группы прокси.

Чтобы использовать эти параметры, отметьте флажки перед соответствующими группами прокси, затем нажмите нужную кнопку.

Использование фильтра

Вы можете использовать фильтр, чтобы отображать только те группы прокси, которые вам интересны. Для повышения производительности поиска данные ищутся с неразрешенными макросами.

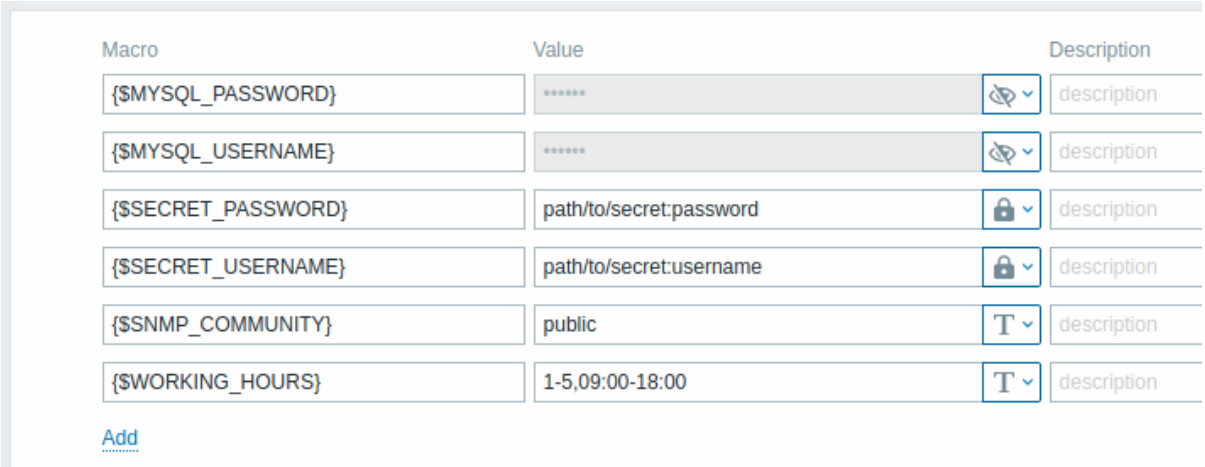
Ссылка *Filter* доступна над списком групп прокси. Если вы нажмете ее, станет доступен фильтр, с помощью которого можно фильтровать группы прокси по имени и статусу.




### 6. Макросы

Обзор

Эта страница позволяет определить общесистемные пользовательские макросы в виде пар имя-значение. Обратите внимание, что значения макросов могут храниться в виде обычного текста, скрытого текста либо секрета Хранилища. Также поддерживается добавление описания.




### 7. Очередь

Обзор

В разделе *Administration* → *Queue* отображаются элементы данных, ожидающие обновления.

В идеале при открытии этого раздела все должно быть ”green”, то есть в очереди не должно быть элементов данных. Если все элементы данных обновляются без задержки, ожидающих нет. Однако из-за недостаточной производительности сервера некоторые элементы данных могут задерживаться, и эта информация отображается в данном разделе. Подробнее см. раздел Queue.

**Note:**
Прокси Zabbix не учитывает периоды обслуживания; подробности см. в разделе Calculation of queues during maintenance.

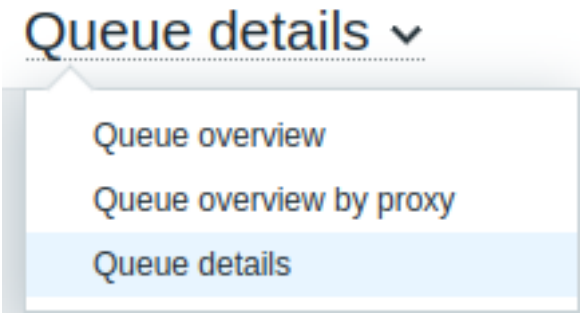
**Note:**
Очередь доступна только если сервер Zabbix запущен. Элементы данных не учитываются в очереди, если интерфейс элемента данных становится недоступен из-за проблем с подключением или если агент работает некорректно.

Раздел *Administration* → *Queue* содержит следующие страницы:

- Queue overview — отображает очередь по типу элемента данных;
- Queue overview by proxy — отображает очередь по прокси;
- Queue details — отображает список задержанных элементов данных.

Список доступных страниц появляется при нажатии *Queue* в разделе меню *Administration*. Также можно переключаться между страницами с помощью выпадающего списка заголовка в левом верхнем углу.

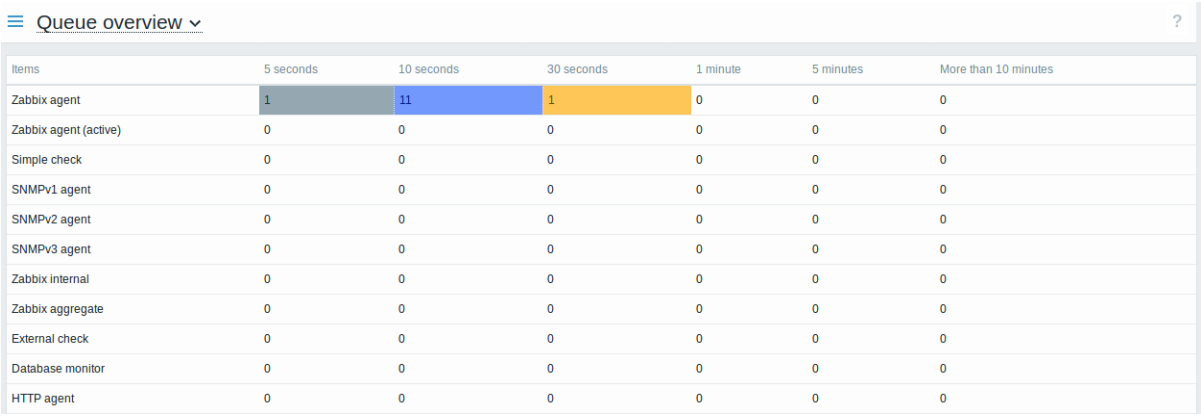
Меню третьего уровня.




Выпадающий список заголовка.

Обзор по типу элементов данных

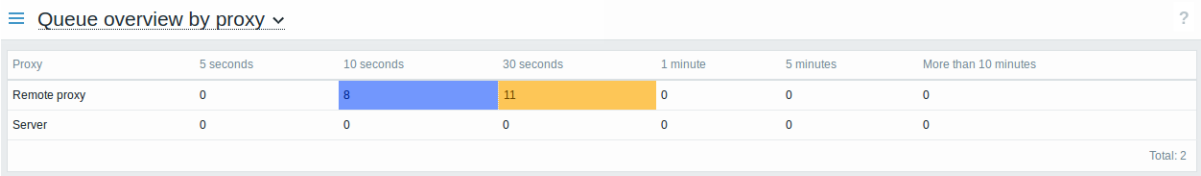
На этом экране можно легко определить, связана ли проблема с одним или несколькими типами элементов данных.




Каждая строка содержит тип элемента данных. Каждый столбец отображает количество ожидающих элементов данных: ожидающих 5—10 секунд / 10—30 секунд / 30—60 секунд / 1—5 минут / 5—10 минут и более 10 минут соответственно.

Обзор по прокси

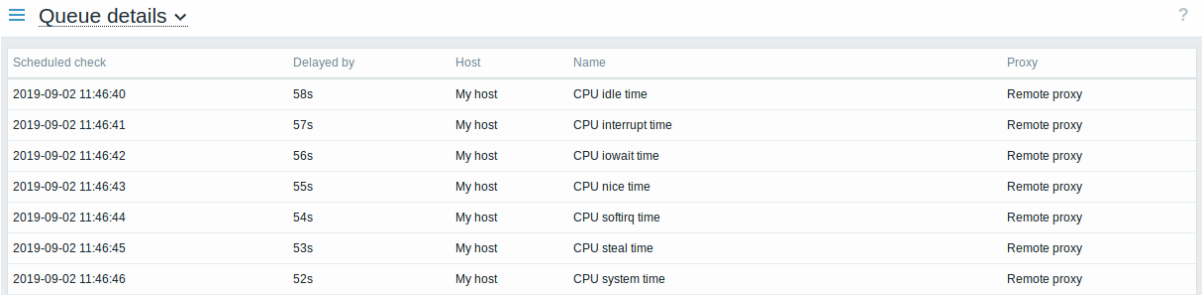
На этом экране можно легко определить, связана ли проблема с одним из прокси или с сервером.




Каждая строка содержит один прокси, и последняя строка в списке — сервер. Каждый столбец отображает количество ожидающих элементов данных: ожидающих 5—10 секунд / 10—30 секунд / 30—60 секунд / 1—5 минут / 5—10 минут и более 10 минут соответственно.

Список ожидающих элементов данных

На этом экране перечислены все ожидающие элементы данных.




Отображаемые данные:

| Столбец | Описание |
|---|---|
| *Запланированная проверка (Scheduled check)* | Время, когда проверка должна была выполниться. |
| *Задерживается на (Delayed by)* | Длительность задержки. |
| *Узел сети (Host)* | Узел сети, которому принадлежит элемент данных. |
| *Имя (Name)* | Имя ожидающего элемента данных. |
| *Прокси (Proxy)* | Имя прокси-сервера, если узел сети наблюдается через прокси. |

Возможные сообщения об ошибках

Вы можете столкнуться с ситуацией, когда никакие данные не отображаются и появляется следующее сообщение об ошибке:

Сообщение об ошибке в этом случае такое:

.
```
(Cannot display item queue. Permission denied)
```

Такое случается, когда параметры конфигурации PHP в файле *zabbix.conf.php* — `$ZBX_SERVER` либо как `$ZBX_SERVER`, так и `$ZBX_SERVER_PORT` — указывают на существующий Zabbix сервер, который использует другую базу данных.

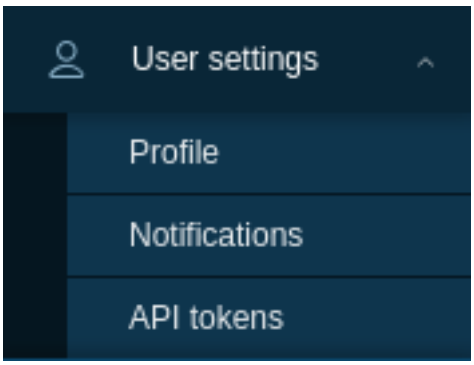
**3 Настройки пользователя**

Обзор

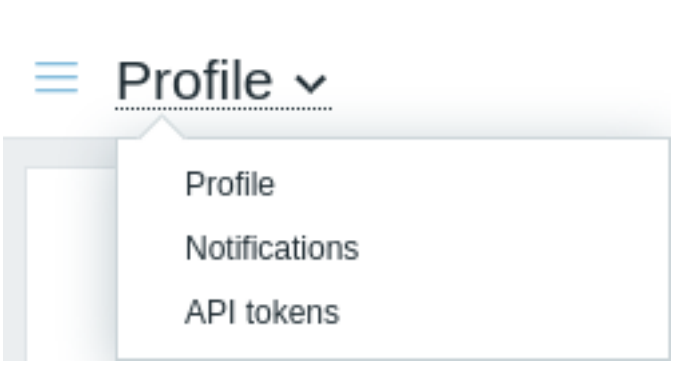
В зависимости от прав роли пользователя, раздел *Настройки пользователя* может содержать следующие страницы:

- *Профиль* или *Профиль пользователя* — для настройки некоторых функций веб-интерфейса Zabbix;
- *Уведомления* — для настройки уведомлений текущего пользователя;
- *Токены API* — для управления токенами API, назначенными текущему пользователю.

Список доступных страниц отображается при нажатии на значок пользователя в нижней части меню Zabbix (недоступно для пользователя *guest*). Также можно переключаться между страницами с помощью выпадающего списка заголовка в левом верхнем углу.



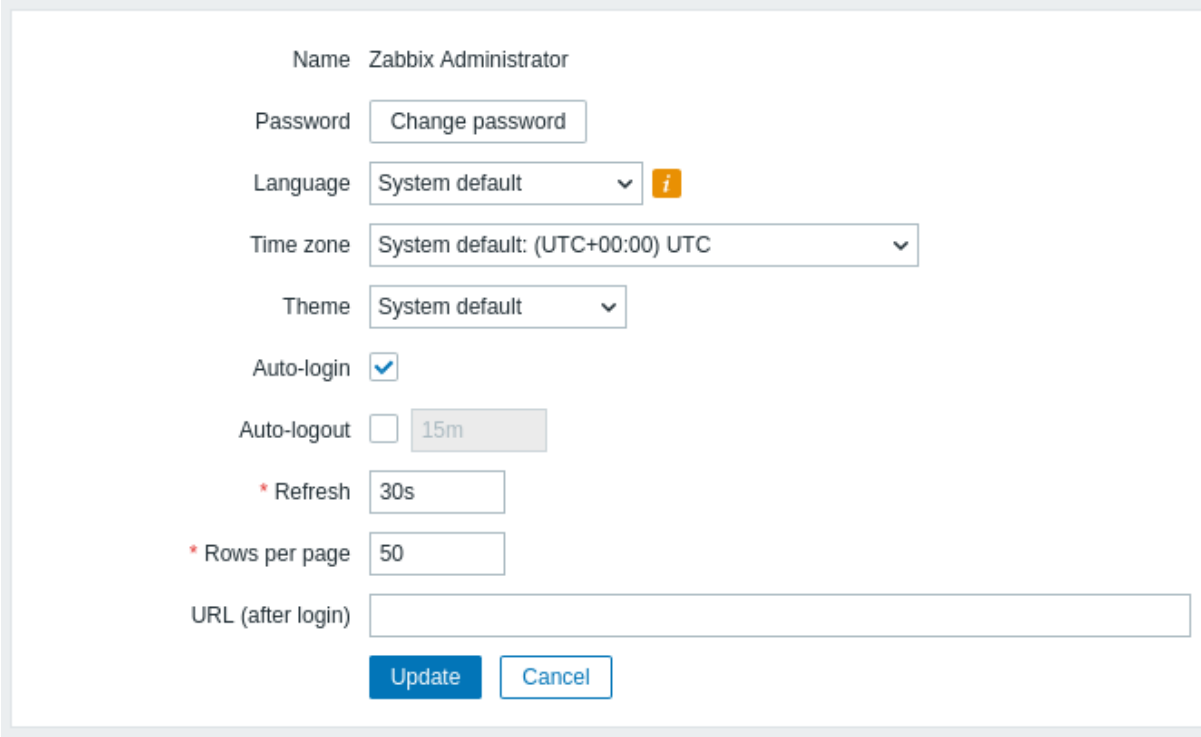

Меню третьего уровня.




Выпадающий список заголовка.

Профиль пользователя

Раздел *Профиль пользователя* предоставляет параметры для настройки языка интерфейса, цветовой темы, количества строк, отображаемых в списках, и т. д. Изменения, внесенные здесь, будут применяться только к текущему пользователю.



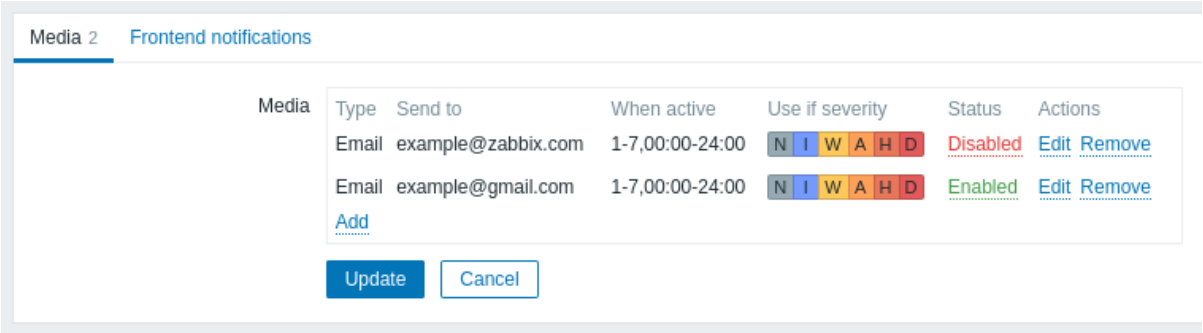

| Parameter | Description |
|---|---|
| *Name* | Отображаются имя и фамилия пользователя.<br>Если имя и фамилия отсутствуют, отображается имя пользователя. |
| *Password* | Нажмите кнопку *Change password*, чтобы открыть три поля: *Old password*, *New password*, *New password (once again)*.<br>После успешной смены пароля пользователь будет выведен из всех активных сеансов.<br>Обратите внимание, что пароль можно изменить только для пользователей, использующих внутреннюю аутентификацию Zabbix. |
| *Language* | Выберите нужный язык интерфейса или **System default**, чтобы использовать системные настройки по умолчанию.<br>При выборе *English (en_US)* в веб-интерфейсе также будет включен формат даты и времени США.<br>Дополнительные сведения см. в разделе Установка дополнительных языков веб-интерфейса. |
| *Time zone* | Выберите часовой пояс, чтобы переопределить глобальный часовой пояс на уровне пользователя, или выберите **System default**, чтобы использовать глобальные настройки часового пояса. |

| Parameter | Description |
|---|---|
| *Theme* | Выберите цветовую тему специально для своего профиля:<br>**System default** - использовать системные настройки по умолчанию;<br>**Blue** - стандартная синяя тема;<br>**Dark** - альтернативная темная тема;<br>**High-contrast light** - светлая тема с высокой контрастностью;<br>**High-contrast dark** - темная тема с высокой контрастностью. |
| *Auto-login* | Установите этот флажок, чтобы Zabbix запоминал вас и выполнял автоматический вход в течение 30 дней.<br>При входе с включенной опцией *Remember for 30 days*:<br>- Таймер автоматического выхода сбрасывается (сеанс сохраняется до окончания 30-дневного периода).<br>- Автоматический вход будет включен при последующих посещениях в течение 30 дней.<br>При входе без *Remember for 30 days*:<br>- Параметр автоматического входа очищается.<br>- Автоматический выход выполняется в соответствии с настроенным тайм-аутом, как обычно.<br>Для этого используются файлы cookie браузера. |
| *Auto-logout* | Если этот флажок установлен, выход из системы будет выполнен автоматически по истечении заданного количества секунд (минимум 90 секунд, максимум 1 день).<br>Обратите внимание: если вход выполнен с включенной опцией *Remember for 30 days*, таймер автоматического выхода переопределяется, и сеанс остается активным в течение всего периода.<br>Поддерживаются суффиксы времени, например: 90s, 5m, 2h, 1d.<br>Обратите внимание, что этот параметр не будет работать в следующих случаях:<br>* Когда страницы меню *Monitoring* выполняют фоновое обновление информации. Если оставить открытыми страницы, которые обновляют данные через определенный интервал времени (панели мониторинга, графики, последние данные и т. д.), срок действия сеанса соответственно продлевается, что отключает функцию автоматического выхода.<br>* Если вход выполнен с отмеченной опцией *Remember me for 30 days*.<br>* Когда аутентификация выполняется через поставщика удостоверений SSO (IdP), автоматический выход завершает только сеанс Zabbix. Сеанс SSO с IdP остается активным, поэтому пользователь может снова войти в систему без ввода пароля.<br>*Auto-logout* также может принимать значение ”0”, что означает отключение функции автоматического выхода после обновления настроек профиля. |
| *Refresh* | Укажите, как часто будет обновляться информация на страницах меню *Monitoring* (минимум 0 секунд, максимум 1 час).<br>Поддерживаются суффиксы времени, например: 30s, 90s, 1m, 1h. |
| *Rows per page* | Укажите, сколько строк будет отображаться на странице в списках. Меньшее количество строк (и меньшее количество записей для отображения) обеспечивает более быструю загрузку. |
| *URL (after login)* | Укажите конкретный URL, который будет отображаться после входа в систему. Вместо значения по умолчанию *Dashboards* это может быть, например, URL раздела *Monitoring* > *Triggers*. |

Уведомления

Раздел *Уведомления* предоставляет параметры для настройки уведомлений текущего пользователя.

Вкладка *Медиа* позволяет указать сведения о медиа для пользователя, такие как типы медиа и адреса, которые следует использовать, а также когда их использовать для доставки уведомлений.



Если тип медиа был отключен:

- после имени отображается желтый значок информации;
- в столбце *Статус* отображается "Disabled".

**Note:**
Права пользователя на изменение сведений о медиа для себя могут быть предоставлены или отозваны на основе его роли пользователя (см. параметр *Create and edit own media*). Права пользователя super admin на изменение сведений о медиа для других также могут быть предоставлены или отозваны на основе его роли пользователя (см. параметр *Create and edit user media*)

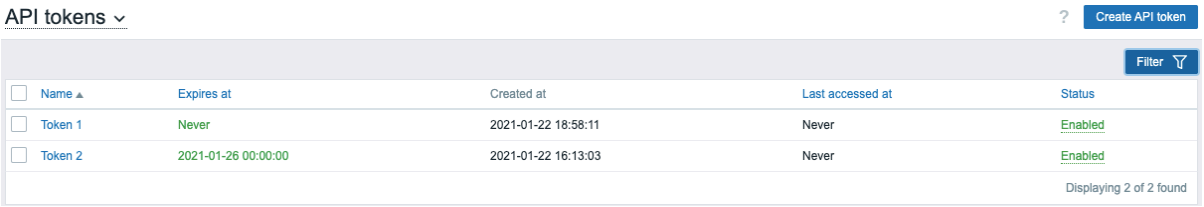
Обратите внимание, что для provisioned users:

- provisioned user media нельзя удалить;
- provisioned user media можно отключать и включать;
- такие поля provisioned user media, как *When active*, *Use if severity* и *Enabled*, можно редактировать вручную;
- для provisioned users можно вручную добавлять дополнительные user media (например, дополнительные адреса электронной почты);
- user media, добавленные вручную, можно удалять.

Вкладка *Frontend notifications* позволяет настроить глобальные уведомления.

API tokens

Раздел *API tokens* позволяет просматривать токены, назначенные пользователю, редактировать сведения о токене и создавать новые токены. Этот раздел доступен пользователю только в том случае, если действие *Manage API tokens* разрешено в настройках роли пользователя.




Вы можете фильтровать API tokens по имени, дате истечения срока действия или статусу (*Enabled/Disabled*). Щелкните статус токена в списке, чтобы быстро включить или отключить токен. Вы также можете включать или отключать несколько токенов одновременно, выбрав их в списке, а затем нажав кнопки *Enable/Disable* под списком.

**Attention:**
Пользователи не могут просматривать значение *Auth token* токенов, назначенных им в Zabbix. Значение *Auth token* отображается только один раз - сразу после создания токена. Если токен был утерян, его необходимо сгенерировать заново.

### 1 Глобальные оповещения

Обзор

Глобальные уведомления предоставляют способ отображать проблемы в реальном времени непосредственно на вашем текущем экране в веб-интерфейсе Zabbix.

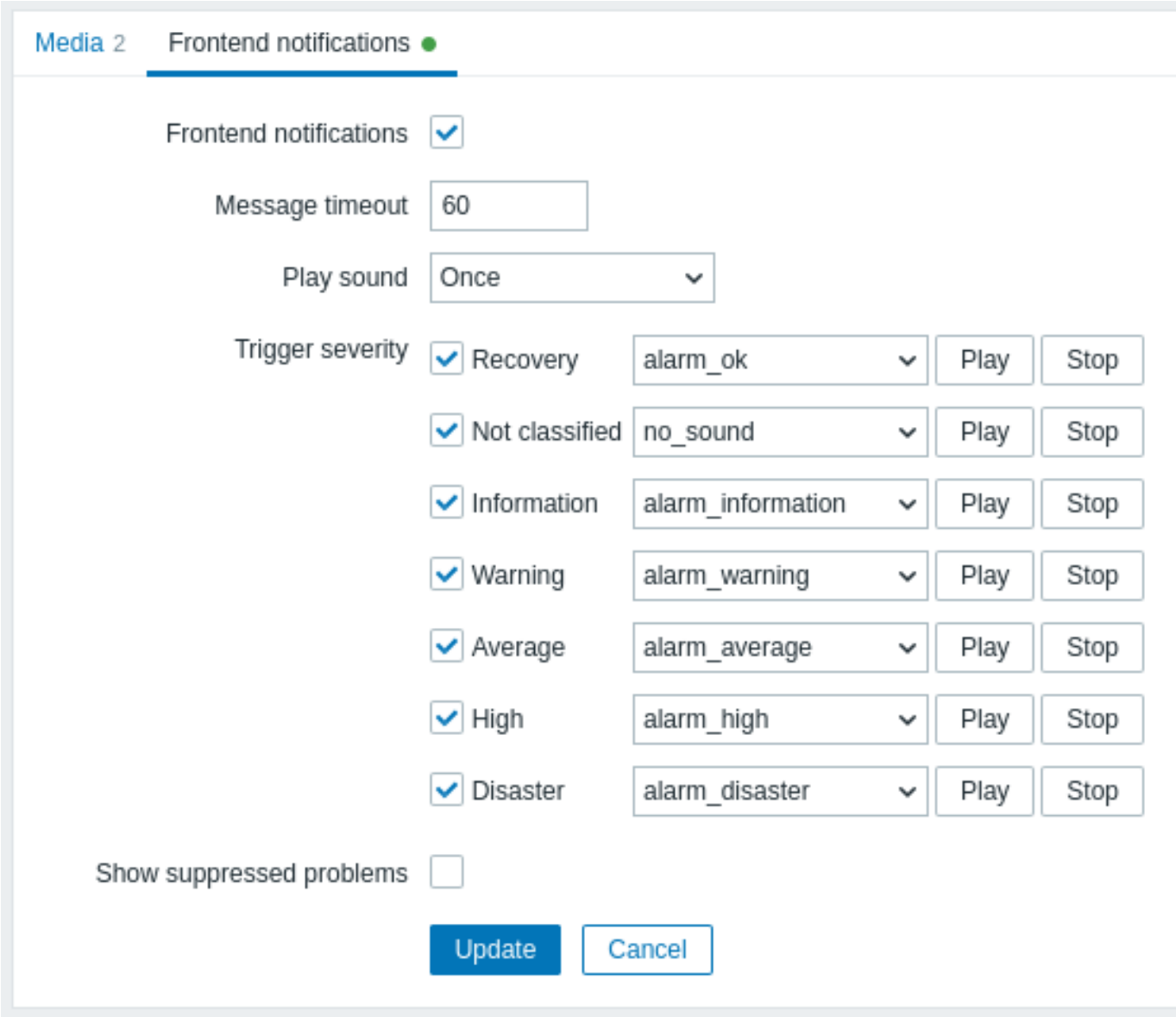
Без глобальных уведомлений при работе вне разделов *Проблемы* или *Дашборд* вы не получали бы никакой информации о текущих проблемах. Глобальные уведомления гарантируют, что эта информация будет отображаться независимо от того, где вы находитесь в веб-интерфейсе Zabbix.

Глобальные уведомления включают как отображение сообщения, так и воспроизведение звука.

**Attention:**
Автовоспроизведение звуков может быть отключено (по умолчанию) в последних версиях браузеров.
В таких случаях необходимо включить этот параметр вручную.

Конфигурация

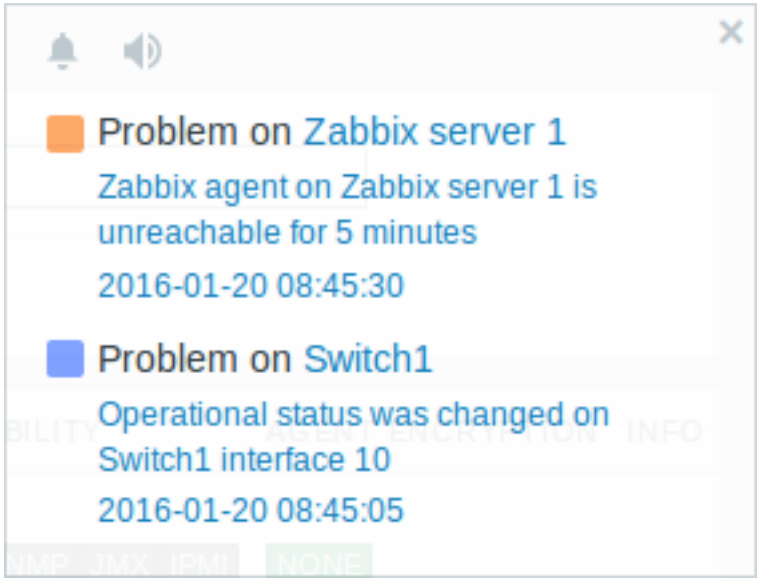
Глобальные уведомления можно включить для каждого пользователя на вкладке **Frontend notifications** в разделе Notifications.




| Parameter | Description |
|---|---|
| *Frontend notifications* | Отметьте флажок, чтобы включить глобальные уведомления. |
| *Message timeout* | Укажите длительность, в течение которой будет отображаться сообщение. По умолчанию сообщения остаются на экране 60 секунд.<br>Поддерживаются суффиксы времени, например: 30s, 5m, 2h, 1d. |
| *Play sound* | Укажите длительность, в течение которой будет воспроизводиться звук.<br>**Once** — звук воспроизводится один раз полностью;<br>**10 seconds** — звук повторяется в течение 10 секунд;<br>**Message timeout** — звук повторяется, пока сообщение отображается. |
| *Trigger severity* | Укажите уровни серьезности триггеров, для которых будут активироваться глобальные уведомления и звуки. Также можно выбрать звуки, соответствующие различным уровням серьезности.<br>Если ни один уровень серьезности не отмечен, сообщения отображаться не будут.<br>Кроме того, сообщения о восстановлении будут отображаться только для отмеченных уровней серьезности. Например, если отмечены *Recovery* и *Disaster*, глобальные уведомления будут отображаться для проблем и восстановлений триггеров с уровнем серьезности *Disaster*. |
| *Show suppressed problems* | Отметьте флажок, чтобы отображать уведомления о проблемах, которые в противном случае были бы подавлены (не показаны) из-за обслуживания узла сети. |

Глобальные сообщения

По мере поступления сообщения отображаются в плавающем разделе в правой части экрана. Вы можете свободно перемещать этот раздел, перетаскивая его заголовок.




Для этого раздела доступны следующие элементы управления:

- Кнопка **Snooze** отключает текущий активный звук тревоги;
- Кнопка **Mute/Unmute** переключает между воспроизведением и полным отключением звуков тревоги.

**2 Звук в браузерах**

Обзор

Звук используется в глобальных уведомлениях.

Чтобы звуки воспроизводились во веб-интерфейсе Zabbix, в профиле пользователя на вкладке *Уведомления веб-интерфейса* должны быть включены *Уведомления веб-интерфейса*, при этом должны быть отмечены все уровни серьезности триггеров. Кроме того, звук должен быть включен во всплывающем окне глобальных уведомлений.

Если по какой-либо причине на устройстве невозможно воспроизведение звука, кнопка во всплывающем окне глобальных уведомлений будет постоянно находиться в состоянии ”без звука”, а при наведении на кнопку будет отображаться сообщение ”Cannot support notification audio for this device”.

Звуки, включая стандартные аудиоклипы, поддерживаются только в формате MP3.

Звуки веб-интерфейса Zabbix были успешно протестированы в последних версиях браузеров Firefox и Opera в Linux, а также в браузерах Chrome, Firefox, Microsoft Edge и Opera в Windows.

**Attention:**
Автовоспроизведение звуков в последних версиях браузеров может быть отключено по умолчанию. В таких случаях эту настройку необходимо включить вручную.

**4 Глобальный поиск**

В веб-интерфейсе Zabbix можно выполнять поиск по узлам сети, группам узлов сети, шаблонам и группам шаблонов.

Поле ввода поиска находится под логотипом Zabbix в меню. Поиск можно запустить, нажав *Enter* или щелкнув значок поиска .

Если есть узел сети, в названии которого в любой части содержится введенная строка, появится выпадающий список со всеми такими узлами сети (при этом совпадающая часть будет выделена оранжевым цветом). В выпадающем списке также будет показан узел сети, если его видимое имя совпадает с техническим именем, введенным в качестве строки поиска; такой узел сети будет отображен, но без какого-либо выделения.

Поисковые атрибуты

Узлы сети можно искать по следующим свойствам:

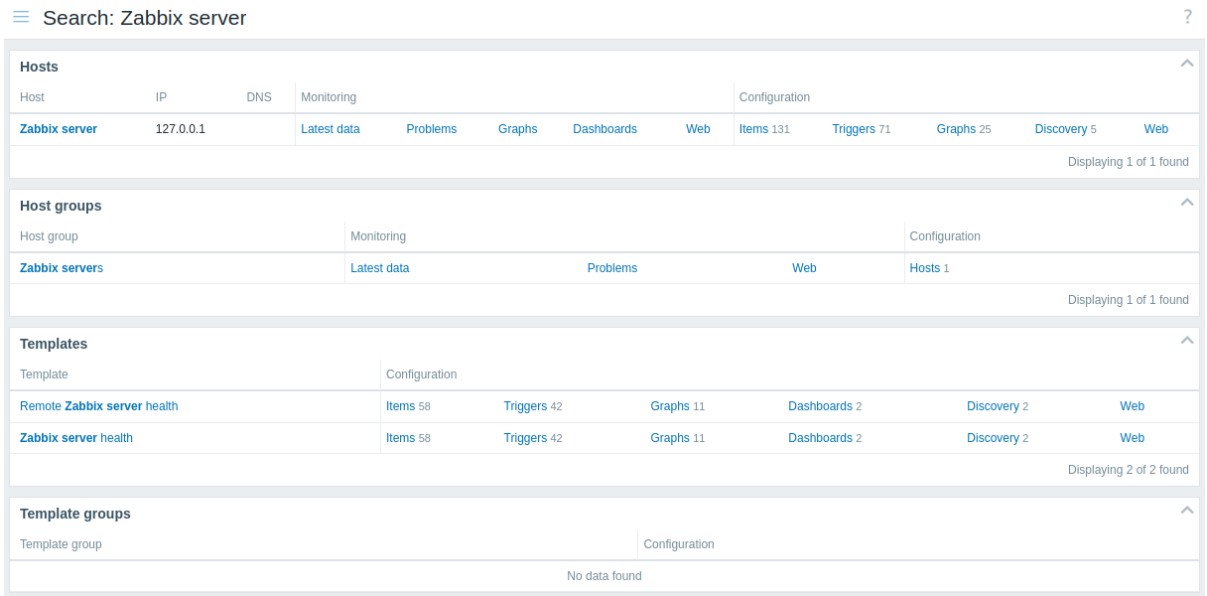
- Имя узла сети
- Отображаемое имя
- IP-адрес
- DNS-имя

Шаблоны можно искать по имени или отображаемому имени. Если вы ищете по имени, которое отличается от отображаемого имени (шаблона/узла сети), в результатах поиска оно отображается под отображаемым именем в скобках.

Группы узлов сети и шаблонов можно искать по имени. Указание родительской группы неявно выбирает все вложенные группы.

Результаты поиска

Результаты поиска состоят из четырех отдельных блоков для узлов сети, групп узлов сети, шаблонов и групп шаблонов.




Каждый отдельный блок можно свернуть или развернуть. Количество записей отображается в нижней части каждого блока, например, *Отображается 13 из 13 найденных*. Если записей нет, количество записей не отображается. Общее количество записей, отображаемых в одном блоке, ограничено 100.

Каждая запись содержит ссылки на данные мониторинга и конфигурации. См. полный список ссылок.

Для всех данных конфигурации (таких как элементы данных, триггеры, графики) количество найденных сущностей отображается числом рядом с названием сущности, серым цветом. **Обратите внимание**, что если сущностей нет, число не отображается.

Включенные узлы сети отображаются синим цветом, отключенные узлы сети — красным.

Доступные ссылки

Для каждой записи доступны следующие ссылки:

- Узлы сети
  - Мониторинг
    - Последние данные
    - Проблемы
    - Графики
    - Панели узла сети
    - Веб-сценарии
  - Конфигурация
    - Элементы данных
    - Триггеры
    - Графики
    - Правила обнаружения
    - Веб-сценарии
- Группы узлов сети
  - Мониторинг
    - Последние данные
    - Проблемы
    - Веб-сценарии
  - Конфигурация
    - Узлы сети
- Шаблоны
  - Конфигурация
    - Элементы данных
    - Триггеры
    - Графики
    - Панели шаблона
    - Правила обнаружения
    - Веб-сценарии
- Группы шаблонов
  - Конфигурация
    - Шаблоны

**5 Режим обслуживания веб-интерфейса**

Обзор

Можно временно отключить веб-интерфейс Zabbix, чтобы ограничить доступ. Это полезно для защиты базы данных Zabbix от изменений, инициированных пользователями, и для сохранения ее целостности.

Пока веб-интерфейс Zabbix находится в режиме обслуживания, вы можете безопасно остановить базу данных и выполнить задачи по обслуживанию.

Пользователи с указанных IP-адресов смогут нормально взаимодействовать с веб-интерфейсом во время режима обслуживания.

Конфигурация

Чтобы включить режим обслуживания, откройте файл `maintenance.inc.php` (расположенный в `/conf` каталога HTML-документов Zabbix на веб-сервере) и раскомментируйте следующие строки:

```
//           .
define('ZBX_DENY_GUI_ACCESS', 1);

//     IP-    ,                        -      (       ).
$ZBX_GUI_ACCESS_IP_RANGE = array('127.0.0.1');
```

```
//      ,                          (       ).
$ZBX_GUI_ACCESS_MESSAGE = '                        MySQL   15:00.            ...';
```

**Note:**
В большинстве случаев файл `maintenance.inc.php` находится в `/conf` каталога HTML-документов Zabbix на веб-сервере.
Однако некоторые операционные системы и веб-серверы могут использовать другое расположение.
Например, расположение для:

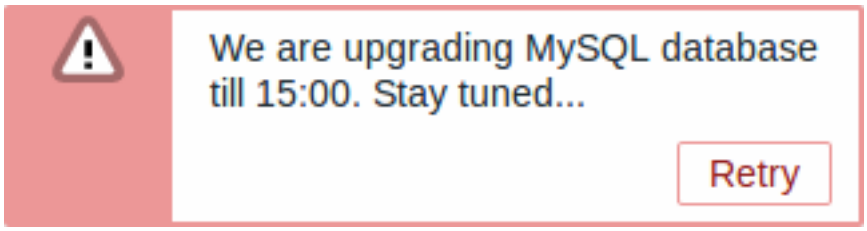
- SUSE и RedHat — `/etc/zabbix/web/maintenance.inc.php`.
- систем на базе Debian — `/usr/share/zabbix/conf/`.

См. также Копирование файлов PHP.

| Parameter | Details |
|---|---|
| **ZBX_DENY_GUI_ACCESS** | Если задано любое значение, будет включен режим обслуживания.<br>Чтобы отключить режим обслуживания, закомментируйте строку или удалите ее. |
| **ZBX_GUI_ACCESS_IP_RANGE** | Массив IP-адресов, которым разрешено подключаться к веб-интерфейсу (необязательно).<br>Например:<br>`array('192.168.1.1', '192.168.1.2')` |
| **ZBX_GUI_ACCESS_MESSAGE** | Сообщение для информирования пользователей об обслуживании (необязательно).<br>Если не задано, будет использовано сообщение по умолчанию *'Zabbix is under maintenance'*. |

Отображение

Пользователи увидят следующий экран при попытке доступа к веб-интерфейсу Zabbix в режиме обслуживания. Экран обновляется каждые 30 секунд, чтобы после завершения обслуживания вернуться в нормальное состояние без вмешательства пользователя.




IP-адреса, определенные в *ZBX_GUI_ACCESS_IP_RANGE*, смогут получать доступ к веб-интерфейсу как обычно.

**6 Параметры страницы**

Обзор

Большинство страниц веб-интерфейса Zabbix поддерживают различные параметры HTTP GET, которые управляют тем, что будет отображаться. Их можно передавать, указывая пары параметр=значение после URL, отделяя их от URL вопросительным знаком (?) и друг от друга амперсандом (&).

Мониторинг > Проблемы

Поддерживаются следующие параметры:

| Parameter | Description | Example |
|---|---|---|
| *show* | Параметр фильтра *Показать*.<br><br>Возможные значения:<br>1 - недавние проблемы;<br>2 - все;<br>3 - в состоянии проблемы. | show=1 |
| *name* | Параметр фильтра *Проблема*: строка произвольного формата. | name=Zabbix agent |

| Parameter | Description | Example |
|---|---|---|
| *severities* | Параметр фильтра *Серьезность*: массив выбранных уровней серьезности в формате `severities[*]=*` (замените * на уровень серьезности).<br><br>Возможные значения:<br>0 - не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа. | severities[3]=3 |
| *inventory* | Параметр фильтра *Инвентаризация узла сети*: массив полей инвентаризации `[field],[value]` | inventory[0][field]=type&invento |
| *evaltype* | Параметр фильтра *Теги*: метод оценки тегов.<br><br>Возможные значения:<br>0 - и/или;<br>2 - или. | evaltype=0 |
| *tags* | Параметр фильтра *Теги*: массив заданных тегов `[tag], [operator], [value]`<br><br>Возможные значения operator:<br>0 - содержит;<br>1 - равно;<br>2 - не содержит;<br>3 - не равно;<br>4 - существует;<br>5 - не существует. | tags[0][tag]=target&tags[0][ope |
| *show_tags* | Параметр фильтра *Показывать теги*.<br><br>Возможные значения:<br>0 - нет;<br>1 - один;<br>2 - два;<br>3 - три. | show_tags=3 |
| *tag_name_format* | Параметр фильтра *Имя тега*.<br><br>Возможные значения:<br>0 - полное имя;<br>1 - сокращенное имя;<br>2 - нет. | tag_name_format=1 |
| *tag_priority* | Параметр фильтра *Приоритет отображения тегов*: строка приоритетов отображения тегов, разделенных запятыми | tag_priority=customer, target |
| *show_suppressed* | Параметр фильтра *Показывать подавленные проблемы*.<br><br>Возможные значения:<br>1 - показывать;<br>2 - не показывать. | show_suppressed=1 |
| *acknowledgement_status* | Параметр фильтра *Статус подтверждения*.<br><br>Возможные значения:<br>0 - все;<br>1 - неподтвержденные;<br>2 - подтвержденные. | acknowledgement_status=0 |
| *acknowledged_by_me* | Параметр фильтра *Мною*; поддерживается только с `acknowledgement_status=2`.<br><br>Возможные значения:<br>0 - отключено;<br>1 - включено. | acknowledged_by_me=1 |

| Parameter | Description | Example |
|---|---|---|
| *compact_view* | Параметр фильтра *Компактный вид*.<br><br>Возможные значения:<br>0 - отключено;<br>1 - включено. | compact_view=1 |
| *highlight_row* | Параметр фильтра *Выделять всю строку*.<br><br>Возможные значения:<br>0 - отключено;<br>1 - включено.<br><br>Для версий Zabbix ранее 7.4.3 этот параметр поддерживается только при `compact_view=1`. | highlight_row=1 |
| *filter_name* | Параметр свойств фильтра *Имя*: строка произвольного формата | filter_name=Databases |
| *filter_show_counter* | Параметр свойств фильтра *Показывать число записей*.<br><br>Возможные значения:<br>0 - отключено;<br>1 - включено. | filter_show_counter=1 |
| *filter_custom_time* | Параметр свойств фильтра *Задать произвольный период времени*.<br><br>Возможные значения:<br>0 - отключено;<br>1 - включено. | filter_custom_time=1 |
| *sort* | Столбец, по которому выполняется сортировка.<br><br>Возможные значения:<br>clock - сортировать по столбцу *Время*;<br>host - сортировать по столбцу *Узел сети*;<br>severity - сортировать по столбцу *Серьезность*;<br>name - сортировать по столбцу *Проблема*. | sort=clock |
| *sortorder* | Порядок сортировки результатов.<br><br>Возможные значения:<br>DESC - по убыванию;<br>ASC - по возрастанию. | sortorder=DESC |
| *age_state* | Параметр фильтра *Меньше чем возраст*; поддерживается только с `show=3`.<br><br>Возможные значения:<br>0 - отключить параметр age;<br>1 - включить параметр age. | age_state=1 |
| *age* | Параметр фильтра *Меньше чем возраст*: целое число, количество дней; поддерживается только с `age_state=1` и `show=3`. | age=7 |
| *groupids* | Параметр фильтра *Группы узлов сети*: массив ID групп узлов сети. | groupids[]=4 |
| *hostids* | Параметр фильтра *Узлы сети*: массив ID узлов сети | hostids[]=10084 |
| *triggerids* | Параметр фильтра *Триггеры*: массив ID триггеров | triggerids[]=22382 |
| *show_timeline* | Параметр фильтра *Показывать временную шкалу*; не поддерживается с `compact_view=1`.<br><br>Возможные значения:<br>0 - не показывать;<br>1 - показывать. | show_timeline=1 |
| *details* | Параметр фильтра *Показывать подробности*.<br><br>Возможные значения:<br>0 - не показывать;<br>1 - показывать. | details=1 |
| *from* | Начало диапазона дат, может быть относительным (например, now-1m); поддерживается только с `filter_custom_time=1` | from=now-2h |

| Parameter | Description | Example |
|---|---|---|
| *to* | Конец диапазона дат, может быть относительным (например, now-1m); поддерживается только с `filter_custom_time=1` | to=now |

См. также: параметры фильтра страницы Проблемы.

Режим киоска

Режим киоска на поддерживаемых страницах веб-интерфейса можно активировать с помощью параметров URL. Например, на панелях мониторинга:

- `/zabbix.php?action=dashboard.view&kiosk=1` - активировать режим киоска
- `/zabbix.php?action=dashboard.view&kiosk=0` - активировать обычный режим

Слайд-шоу

На панели можно активировать слайд-шоу:

- `/zabbix.php?action=dashboard.view&slideshow=1` - активировать слайд-шоу

**7 Определения**

Обзор

Хотя многие параметры веб-интерфейса можно настроить, используя сам веб-интерфейс, однако, некоторые настройки в настоящее время возможно изменить только через файл определений.

Этим файлом является `defines.inc.php` расположенный в папке /include раздела документов Zabbix HTML.

Параметры

Параметры в этом файле, которые могут представлять интерес для пользователей:

- ZBX_MIN_PERIOD

Минимальный период графика, в секундах. По умолчанию — одна минута.

- GRAPH_YAXIS_SIDE_DEFAULT

Расположение оси Y по умолчанию в простых графиках и значение по умолчанию для выпадающего списка при добавлении элементов данных в пользовательские графики. Возможные значения: 0 - слева, 1 - справа.

По умолчанию: 0

- ZBX_SESSION_NAME

Строка, используемая в качестве имени cookie сеанса веб-интерфейса Zabbix.

По умолчанию: zbx_sessionid

- ZBX_DATA_CACHE_TTL

Время ожидания TTL в секундах, используемое для инвалидирования кэша данных ответа Vault. Установите 0, чтобы отключить кэширование ответов Vault.

По умолчанию: 60

- SUBFILTER_VALUES_PER_GROUP

Количество значений подфильтра на группу (например, в подфильтре последних данных).

По умолчанию: 1000

- ZBX_MAX_WIDGET_LINES

Максимальное количество строк виджета для отображения.

По умолчанию: 1000

### 8 Создание своей темы

Обзор

По умолчанию Zabbix поставляется с некоторым количеством предустановленных тем. Вы можете выполнить процедуру приведенную здесь, шаг за шагом, для того, чтобы создать свою собственную тему. Не стесняйтесь поделиться результатами своей работы с сообществом Zabbix, если создали что-то действительно приятное.

Шаг 1

Для того чтобы создать свою тему, вам необходимо создать CSS файл и сохранить его в `assets/styles/` (назвав его, например, *custom-theme.css*) Вы можете скопировать эти файлы из другой темы и создать свою тему на основе этой темы или начать с нуля.

Шаг 2

Добавьте свою тему в список тем, возвращаемых методом APP::getThemes(). Вы можете сделать это, переопределив метод ZBase::getThemes() в классе приложения. Это можно сделать, добавив следующий код перед закрывающими фигурными скобками в *include/classes/core/APP.php*:

public static function getThemes() { return array_merge(parent::getThemes(), [ 'custom-theme' => _('Custom theme') ]); }

**Attention:**
Обратите внимание, что имя, указанное в первой пара кавычек должна соответствовать имени файла темы без расширения.

Чтобы добавить несколько тем, просто перечислите их под первой темой, например:

public static function getThemes() { return array_merge(parent::getThemes(), [ 'custom-theme' => *('Custom theme'), 'anoth-ertheme'* => ('Another theme'), 'onemoretheme' => _('One more theme') ]); }

Обратите внимание, что в каждой строке, кроме последней, должна быть завершающая запятая.

**Note:**
Чтобы изменить цвета графика, необходимо добавить запись в таблицу базы данных *graph_theme*.

Шаг 3

Активируйте новую тему.

Во внешнем интерфейсе Zabbix вы можете установить эту тему как тему по умолчанию. или изменить тему в профиле пользователя.

Наслаждайтесь новым внешним видом!

### 9 Режим отладки

Обзор

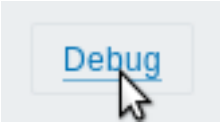
Режим отладки можно использовать для диагностики проблем с производительностью страниц веб-интерфейса Zabbix.

Конфигурация

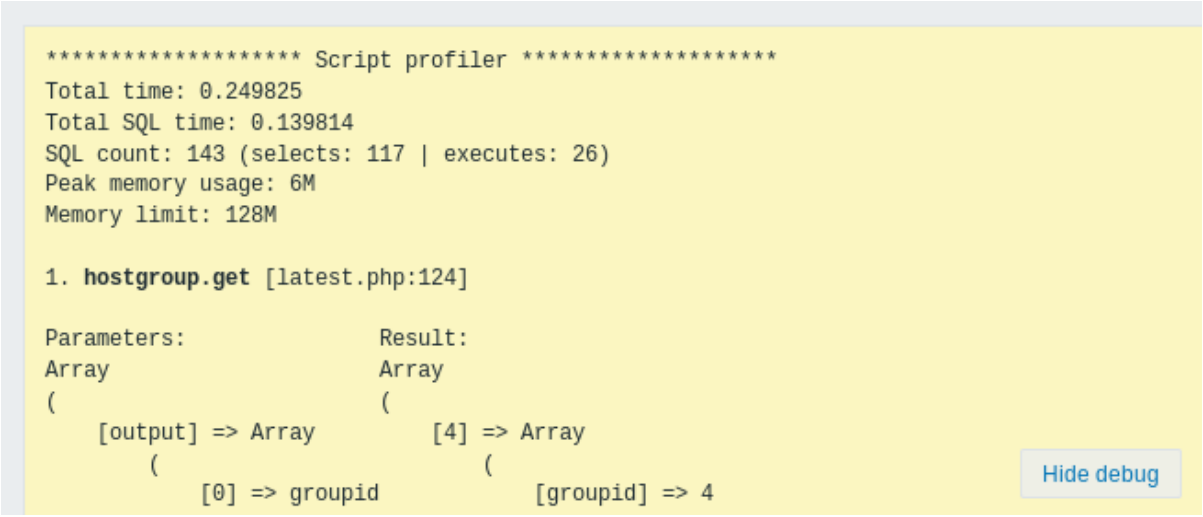
Режим отладки можно включить для отдельных пользователей, которые входят в группу пользователей:

- при настройке группы пользователей;
- при просмотре настроенных групп пользователей.

Когда для группы пользователей включен *Режим отладки*, ее пользователи увидят кнопку *Debug* в правом нижнем углу окна браузера:



При нажатии кнопки *Debug* под содержимым страницы откроется новое окно, содержащее SQL-статистику страницы, а также список вызовов API и отдельных SQL-операторов:




В случае проблем с производительностью страницы это окно можно использовать для поиска первопричины проблемы.

**Warning:**
Включенный *Режим отладки* негативно влияет на производительность веб-интерфейса.

**10 Файлы cookie используемые Zabbix**

Обзор

На этой странице приведен список cookie, используемых Zabbix.

| Name | Description | Values | Expires/Max-Age | HttpOnly[1] | Secure[2] |
|---|---|---|---|---|---|
| ZBX_SESSION_NAME | Данные сеанса веб-интерфейса Zabbix, хранящиеся в виде JSON, закодированного с помощью base64 | | Сеанс (истекает при завершении сеанса просмотра) | + | + (только если на веб-сервере включен HTTPS) |
| tab | Номер активной вкладки; этот cookie используется только на страницах с несколькими вкладками (например, на странице настройки *Host*, *Trigger* или *Action*) и создается, когда пользователь переходит с основной вкладки на другую вкладку (например, *Tags* или *Dependencies*).<br><br>0 используется для основной вкладки. | Пример: 1 | Сеанс (истекает при завершении сеанса просмотра) | - | - |
| browserwarning_ignore | Сигнализирует игнорировать предупреждение об использовании устаревшего браузера. | yes | Сеанс (истекает при завершении сеанса просмотра) | - | - |
| system-message-ok | Сообщение, которое отображается сразу после перезагрузки страницы. | Сообщение в виде обычного текста | Сеанс (истекает при завершении сеанса просмотра) или сразу после перезагрузки страницы | + | - |

| Name | Description | Values | Expires/Max-Age | HttpOnly[1] | Secure[2] |
|---|---|---|---|---|---|
| system-message-error | Сообщение об ошибке, которое отображается сразу после перезагрузки страницы. | Сообщение в виде обычного текста | Сеанс (истекает при завершении сеанса просмотра) или сразу после перезагрузки страницы | + | - |

**Note:**
Принудительное включение флага 'HttpOnly' для cookie Zabbix с помощью директивы веб-сервера не поддерживается.

Примечания

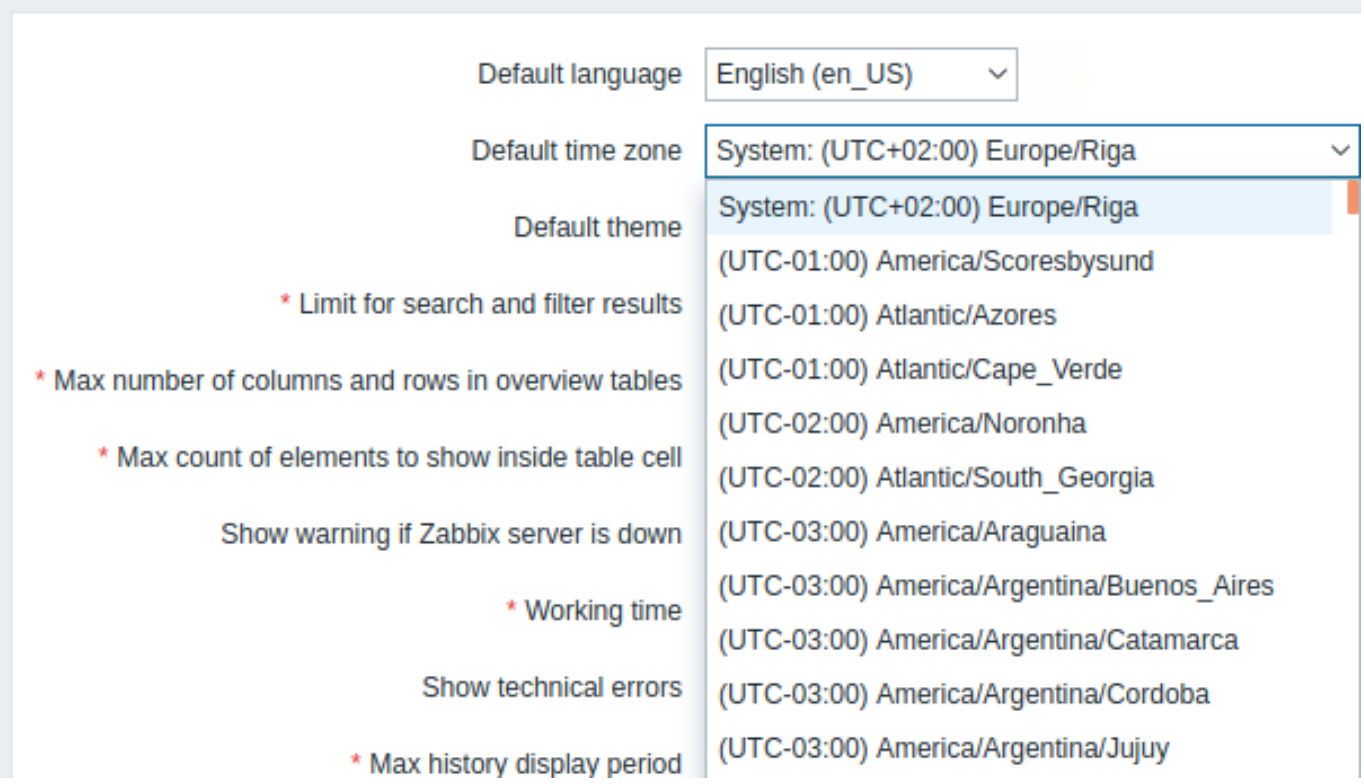
[1] Если `HttpOnly` имеет значение 'true', cookie будет доступен только через протокол HTTP. Это означает, что cookie не будет доступен для языков сценариев, таких как JavaScript. Эта настройка может эффективно помочь снизить риск кражи учетных данных через атаки XSS (хотя она поддерживается не всеми браузерами).

[2] `Secure` указывает, что cookie должен передаваться только по защищенному HTTPS-соединению от клиента. Если установлено значение 'true', cookie будет установлен только при наличии защищенного соединения.

### 11 Часовые пояса

Обзор

Часовой пояс веб-интерфейса можно задать глобально в веб-интерфейсе и настроить для отдельных пользователей.




Если выбран *System*, для веб-интерфейса будет использоваться часовой пояс веб-сервера (включая значение `date.timezone` в php.ini, если оно задано), а сервер Zabbix будет использовать часовой пояс той машины, на которой он запущен.

**Note:**
Сервер Zabbix будет использовать указанный глобальный/пользовательский часовой пояс только при подстановке макросов в уведомлениях (например, {EVENT.TIME} может подставляться в разный часовой пояс для каждого пользователя) и для ограничения по времени, когда отправляются уведомления (см. параметр "When active" в настройках медиа пользователя).

Выбор часового пояса не влияет на формат даты и времени в веб-интерфейсе. Вместо этого можно изменить язык интерфейса (либо при установке, либо в настройках пользователя) — выбор *English (en_US)* также включит формат даты и времени США в веб-интерфейсе.

Конфигурация

Глобальный часовой пояс:

- может быть задан вручную при установке веб-интерфейса
- может быть изменен в *Администрирование* → *Общие* → *GUI*

Часовой пояс на уровне пользователя:

- может быть задан при настройке/обновлении пользователя
- может быть задан каждым пользователем в его профиле пользователя

**См. также:** Согласование часовых поясов при использовании интервалов планирования.

### 12 Сброс пароля

**Обзор** В этом разделе описаны шаги по сбросу паролей пользователей в Zabbix.

**Шаги** Обратитесь к своему администратору Zabbix, если вы забыли свой пароль Zabbix и не можете войти в систему.

Пользователь с ролью супер администратора может изменить пароли для всех пользователей в форме настроек пользователя.

Если пользователь с ролью супер администратора забыл свой пароль и не может войти в систему, необходимо выполнить следующий SQL-запрос, чтобы сбросить пароль к паролю по умолчанию для пользователя с ролью супер администратора (замените 'Admin' соответствующим именем пользователя с ролью супер администратора):

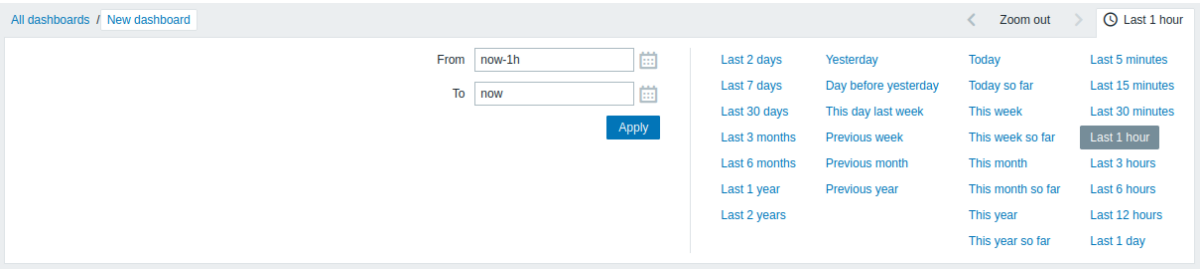
```
UPDATE users SET passwd = '$2a$10$ZXIvHAEP2ZM.dLXTm6uPHOMVlARXX7cqjbhM6Fn0cANzkCQBWpMrS' WHERE username =
```

После выполнения этого запроса пароль пользователя будет установлен в значение *zabbix*. Обязательно измените пароль по умолчанию при первом входе в систему.

### 13. Селектор периода времени

Обзор

Селектор *Time period* позволяет выбрать часто используемые периоды одним щелчком мыши. Его можно развернуть или свернуть, нажав вкладку *Time period* в правом верхнем углу.




Такие варианты, как *Today*, *This week* и т. д., отображают весь период, включая часы/дни в будущем. Такие варианты, как *Today so far*, *This week so far* и т. д., отображают только прошедшие часы.

После выбора периода его можно сдвигать вперед и назад по времени, нажимая кнопки со стрелками . Кнопка *Zoom out* позволяет уменьшить масштаб периода на 50% в каждом направлении.

**Note:**
Для графиков (кроме тех, у которых временной период *Custom*), отображаемый временной период также можно выбрать, выделив область на графике левой кнопкой мыши. После того как вы отпустите левую кнопку мыши, график приблизится к выделенной области. Уменьшить масштаб также можно двойным щелчком по графику.

Поля *From/To* отображают выбранный период либо в абсолютном синтаксисе времени (в формате `Y-m-d H:i:s`), либо в относительном синтаксисе времени. Относительный временной период может содержать одну или несколько математических операций (- или +), например `now-1d` или `now-1d-2h+5m`.

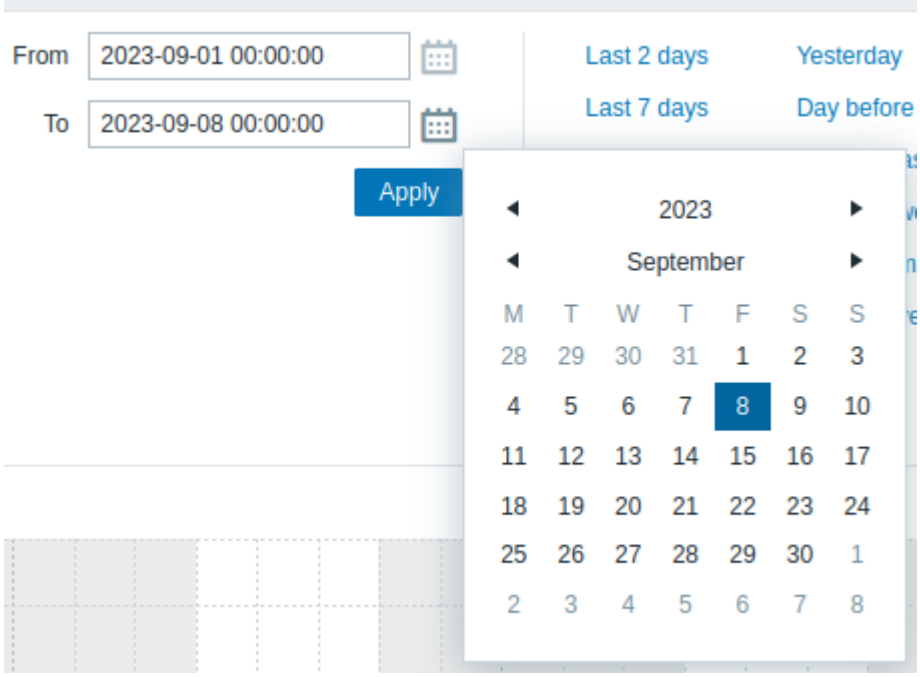
Поддерживаются следующие сокращения относительного времени:

- `now`
- `s` (секунды)
- `m` (минуты)
- `h` (часы)
- `d` (дни)
- `w` (недели)
- `M` (месяцы)
- `y` (годы)

В селекторе *Time period* поддерживается точность (например, `/M` в `now-1d/M`). Подробности о точности:

| Точность | From | To |
|---|---|---|
| `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` | Понедельник недели 00:00:00 | Воскресенье недели 23:59:59 |
| `M` | Первый день месяца 00:00:00 | Последний день месяца 23:59:59 |
| `y` | 1 января года 00:00:00 | 31 декабря года 23:59:59 |

Также можно выбрать временной период с помощью *Date picker*. Чтобы открыть его, нажмите значок календаря рядом с полями *From/To*.




**Note:**
В окне выбора даты можно перемещаться между годом/месяцем/датой с помощью Tab, Shift+Tab и клавиш со стрелками на клавиатуре. Нажатие Enter подтверждает выбор.

Примеры

| От | До | Выбранный период |
|---|---|---|
| `now/d` | `now/d` | 00:00 — 23:59 сегодня |
| `now/d` | `now/d+1d` | 00:00 сегодня — 23:59 завтра |
| `now/w` | `now/w` | Понедельник 00:00:00 — Воскресенье 23:59:59 текущей недели |
| `now-1y/w` | `now-1y/w` | Неделя с понедельника 00:00:00 по воскресенье 23:59:59 годом ранее |

**Attention:**
Использование «now/M+1M» для параметра *До (To)* может добавить 31 день, что может привести к смещению даты на 1-3 дня в зависимости от количества дней в месяце. Например, если использовать в январе, результатом может быть 02 марта вместо ожидаемого 28 февраля. Чтобы избежать этой проблемы, используйте «now/M-3d+1M/M», что точно подстраивается под длину месяца. Аналогично, если настраиваете *От (From)* для перехода назад, используйте «now/M+3d-1M/M».

## 19. Наилучшие практики

**Обзор** В этом разделе описаны рекомендации по настройке Zabbix.

Хотя эти рекомендации не являются обязательными для работы Zabbix, их внедрение настоятельно рекомендуется для обеспечения оптимального и безопасного использования.

### 1. Наилучшие практики по безопасности

Обзор

В этом разделе содержатся рекомендации по настройке Zabbix безопасным образом.

Рекомендации в этом разделе не требуются для работы Zabbix, но рекомендуются для лучшей безопасности системы.

Кодировка UTF-8

UTF-8 является единственной кодировкой, которая поддерживается Zabbix. Известно, что она работает без каких-либо проблем с безопасностью. Пользователи должны знать, что существуют известные проблемы с безопасностью при использовании некоторых других кодировок.

Пути инсталлятора Windows

При использовании инсталлятора Windows рекомендуется использовать предоставляемые инсталлятором пути по умолчанию. Использование пользовательских путей без надлежащим образом назначенных прав может скомпрометировать безопасность вашей инсталляции.

Макросы в определяемых пользователем глобальных скриптах

Для повышения безопасности рекомендуется в определяемых пользователем глобальных скриптах использовать функции макросов вместо обычных макросов, поскольку макросы не экранируются автоматически.

Рекомендации по безопасности Zabbix и база данных CVE

См. *Рекомендации по безопасности Zabbix и базу данных CVE*.

Шаблон электронных писем в формате HTML

При создании или редактировании шаблонов сообщений, используемых для электронных писем в формате HTML, всегда заключайте каждый макрос в функцию макросов `htmlencode()`. Например:

```
<b>          :</b> {{EVENT.TIME}.htmlencode()} {{EVENT.DATE}.htmlencode()}    <br><b>       :</b>
{{EVENT.NAME}.htmlencode()}<br><b>      :</b> {{HOST.NAME}.htmlencode()}
<br><b>     :</b>
{{EVENT.SEVERITY}.htmlencode()}<br><b>           :</b> {{EVENT.OPDATA}.htmlencode()}<br><b>
```

Применение `htmlencode()` обеспечивает экранирование любых HTML-символов в значениях макросов, что предотвращает несанкционированную вставку HTML в уведомление (например, вставку атакующим в оповещение зловредной/фишинговой ссылки).

Обратите внимание: email сообщения в формате HTML по умолчанию, предоставленные компанией Zabbix, уже применяют `htmlencode()` к макросам. Эти рекомендации применимы при редактировании существующих шаблонов или создании новых — убедитесь, что макросы закодированы, прежде чем использовать шаблон для отправки электронных писем в формате HTML.

### 1. Контроль доступа

Обзор

В этом разделе приведены рекомендации по настройке контроля доступа безопасным образом.

Принцип наименьших привилегий

Учётные записи пользователей всегда должны иметь как можно меньше привилегий. Это означает, что учётные записи пользователей веб-интерфейса Zabbix, пользователей баз данных или пользователей, от имени которых запускаются процессы сервера/прокси/агента Zabbix, должны иметь только те привилегии, которые необходимы для выполнения предназначенных функций.

> **Attention:**
> Предоставление избыточных привилегий пользователю «zabbix» позволит ему получать доступ к файлам конфигурации и выполнять операции, которые могут поставить под угрозу безопасность инфраструктуры.

При настройке прав учётной записи пользователя следует учитывать типы пользователей веб-интерфейса Zabbix. Обратите внимание, что, хотя тип пользователя *Администратор* имеет меньше прав, чем тип пользователя *Супер-администратор*, он, тем не менее, может управлять конфигурацией и выполнять пользовательские скрипты.

> **Note:**
> Некоторая информация доступна даже непривилегированным пользователям. Например, хотя раздел *Оповещения → Скрипты* доступен только для пользователей с ролью *Супер-администратор*, скрипты также можно получить посредством Zabbix API. В этом случае ограничение прав доступа к скриптам и исключение конфиденциальной информации из скриптов (например, учётных данных) может помочь избежать раскрытия конфиденциальной информации, доступной в глобальных скриптах.

Защищённый пользователь для Zabbix агента

По умолчанию процессы Zabbix сервера, прокси и агента (или агента 2) используют одного пользователя `zabbix`. Чтобы предотвратить доступ Zabbix агента/агента 2 (работающего на том же компьютере, что и сервер/прокси) к конфиденциальным данным в конфигурации сервера/прокси (например, к учётным данным базы данных), агент следует запускать под другим пользователем:

Для Zabbix агента:

1. Создайте защищённую группу и пользователя (например, `zabbix-agent`).
2. Укажите этого пользователя в параметре User файла конфигурации агента.
3. Перезапустите агента, чтобы сбросить привилегии на нового пользователя.

Для Zabbix агента 2, настройки должны применяться на уровне службы [en], так как файл конфигурации агента 2 не поддерживает параметр `User`. Пример можно посмотреть в ZBX-26442 [en].

Отзыв доступа на запись к конфигурации SSL (Windows)

Если вы скомпилировали агент Zabbix в Windows, где OpenSSL находится в незащищённом каталоге (например, `C:\zabbix`, `c:\openssl-64bit`, `C:\OpenSSL-Win64-111-static` или `C:\dev\openssl`), обязательно отмените доступ на запись в этот каталог для пользователей, не являющихся администраторами. В противном случае агент будет загружать настройки SSL из пути, который могут изменить непривилегированные пользователи, что приведёт к потенциальной уязвимости безопасности.

Усиление безопасности компонентов Zabbix

Некоторые функции можно отключить, чтобы повысить безопасность компонентов Zabbix:

- Глобальное выполнение скриптов на сервере Zabbix можно отключить, установив `EnableGlobalScripts=0` в конфигурации сервера.
- Глобальное выполнение скриптов на прокси Zabbix отключено по умолчанию (можно включить, установив `EnableRemoteCommands=1` в конфигурации прокси).
- Глобальное выполнение скриптов на агентах Zabbix отключено по умолчанию (можно включить, добавив параметр `AllowKey=system.run[<command>,*]` для каждой разрешенной команды в конфигурации агента).
- HTTP-аутентификацию пользователя можно отключить, установив `$ALLOW_HTTP_AUTH=false` в файле конфигурации веб-интерфейса (`zabbix.conf.php`). Обратите внимание, что переустановка веб-интерфейса (запуск `setup.php`) удалит этот параметр.

Доступ к UNC-путям в Windows через Zabbix агент

В элементах данных вроде `vfs.file.*`, `vfs.dir.*` и `perf_counter*` Zabbix агенты в Windows используют UNC-пути (ресурсы SMB наподобие `\\ \ \ .txt`). В некоторых контекстах это может представлять угрозу безопасности.

Когда Windows запрашивает доступ к UNC-пути, выполняется попытка аутентификации на том сервере. Это означает, что вредоносный запрос к Zabbix агенту может раскрыть NTLM-хеш серверу отправителя запроса. Если необходимо, пользователи могут предотвратить это с помощью параметров конфигурации AllowKey и DenyKey.

#### 1 Защита MySQL/MariaDB

Обзор

В этом разделе приведены рекомендации по обеспечению безопасности базы данных MySQL/MariaDB.

> **Note:**
> Для базовой настройки см. инструкции по умолчанию для создания базы данных MySQL/MariaDB, которые включают создание пользователя 'zabbix' с полными привилегиями в базе данных Zabbix. Этот пользователь является владельцем базы данных и также имеет необходимые привилегии для изменения структуры базы данных при обновлении Zabbix.

Для повышения безопасности рекомендуется создавать дополнительные роли и пользователей базы данных с минимальными привилегиями. Эти роли и пользователи должны быть настроены в соответствии с принципом наименьших привилегий, то есть они должны иметь только те привилегии, которые необходимы для выполнения предполагаемых функций.

Создание ролей пользователей

Создайте следующие роли с соответствующими привилегиями:

- **zbx_srv** - роль для запуска сервера Zabbix и прокси:

```
CREATE ROLE 'zbx_srv';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_srv';
FLUSH PRIVILEGES;
```

- **zbx_web** - роль для запуска веб-интерфейса Zabbix и API:

```
CREATE ROLE 'zbx_web';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_web';
FLUSH PRIVILEGES;
```

- **zbx_bckp** - роль для резервного копирования таблиц:

```
CREATE ROLE 'zbx_bckp';
GRANT LOCK TABLES, TRIGGER, SELECT ON zabbix.* TO 'zbx_bckp';
GRANT process ON *.* TO 'zbx_bckp';
FLUSH PRIVILEGES;
```

> **Note:**
> Восстановление и обновление таблиц должны выполняться владельцем базы данных.

- **zbx_part** - роль с сокращенным набором привилегий для секционирования базы данных; обратите внимание, что эту роль можно создать только после создания базы данных, так как она предоставляет привилегии для конкретных таблиц базы данных:

```
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;
```

После создания ролей их можно назначать пользователям.

Назначение ролей пользователей

Чтобы назначить созданные роли пользователей, создайте пользователей и назначьте им соответствующие роли. При необходимости замените <user>, <host>, <role> и <password>.

```
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;
```

Например, чтобы создать и назначить роль для запуска сервера Zabbix и прокси:

```
CREATE USER 'usr_srv'@'localhost' IDENTIFIED BY 'password';
GRANT 'zbx_srv' TO 'usr_srv'@'localhost';
SET DEFAULT ROLE ALL TO 'usr_srv'@'localhost';
FLUSH PRIVILEGES;
```

#### 2 Защита PostgreSQL/TimescaleDB

Обзор

В этом разделе приведены рекомендации по обеспечению безопасности базы данных PostgreSQL.

> **Note:**
> Для базовой настройки см. инструкции по умолчанию для создания базы данных PostgreSQL, которые включают создание пользователя 'zabbix' с полными привилегиями в базе данных Zabbix. Этот пользователь является владельцем базы данных и также имеет необходимые привилегии для изменения структуры базы данных при обновлении Zabbix.

Для повышения безопасности рекомендуется настроить безопасный способ использования схемы, а также создать дополнительные роли и пользователей базы данных с минимальными привилегиями. Эти роли и пользователи должны быть настроены в соответствии с принципом наименьших привилегий, то есть они должны иметь только те привилегии, которые необходимы для выполнения предполагаемых функций.

Настройка базы данных

Создайте пользователя, который будет владельцем базы данных, и создайте базу данных Zabbix; владелец базы данных — это пользователь, указанный при создании базы данных:

```
createuser -U postgres -h localhost --pwprompt usr_owner
createdb -U postgres -h localhost -O usr_owner -E Unicode -T template0 zabbix
```

> **Note:**
> Чистая установка или обновление базы данных должны выполняться владельцем базы данных. Это связано с тем, что право удалять объект базы данных или изменять его определение является привилегией, присущей владельцу базы данных, и не может быть предоставлено или отозвано.

> **Attention:**
> Следующие команды на этой странице должны выполняться только при подключении к PostgreSQL именно к базе данных zabbix.

Создайте схему zabbix и назначьте владельца базы данных (usr_owner) владельцем этой схемы:

```
CREATE SCHEMA zabbix AUTHORIZATION usr_owner;
```

Настройте безопасный шаблон использования схемы:

```
REVOKE CREATE ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON DATABASE zabbix FROM PUBLIC;
--       : search_path                "zabbix":
ALTER ROLE ALL IN DATABASE zabbix SET search_path = "zabbix";
```

После настройки базы данных перейдите к созданию ролей пользователей.

Создание ролей пользователей

Создайте следующие роли с соответствующими привилегиями:

- **zbx_srv** - роль для запуска Zabbix server и 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** - роль для запуска веб-интерфейса Zabbix и 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** - роль для резервного копирования таблиц:

```
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:**
Восстановление таблиц возможно только владельцем базы данных.

После создания ролей их можно назначать пользователям.

Назначение ролей пользователей

Чтобы назначить созданные роли пользователей, создайте пользователей и назначьте им соответствующие роли. При необходимости замените <user>, <role> и <password>.

```
CREATE USER <user> WITH ENCRYPTED password '<password>';
GRANT <role> TO <user>;
```

Например, чтобы создать и назначить роль для запуска Zabbix server и proxy:

```
CREATE USER usr_srv WITH ENCRYPTED password 'password';
GRANT zbx_srv TO usr_srv;
```

Разделение базы данных с помощью TimescaleDB

Разделение базы данных упрощается с помощью TimescaleDB. Чтобы использовать TimescaleDB, сервер Zabbix требует прав владельца базы данных.

Если схема PostgreSQL zabbix уже создана в базе данных zabbix, вы можете включить TimescaleDB с помощью следующей команды:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb WITH SCHEMA zabbix CASCADE;" | sudo -u postgres psql zabb
```

#### 2 Криптография

Обзор

В этом разделе приведены рекомендации по безопасной настройке криптографии.

Настройка SSL для веб-интерфейса Zabbix

На системах на базе RHEL установите пакет mod_ssl:

```
dnf install mod_ssl
```

Создайте каталог для SSL-ключей:

```
mkdir -p /etc/httpd/ssl/private
chmod 700 /etc/httpd/ssl/private
```

Создайте SSL-сертификат:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/httpd/ssl/private/apache-selfsigned.key -
```

Заполните запросы соответствующим образом. Самая важная строка — та, которая запрашивает Common Name. Вы должны указать доменное имя, которое хотите связать с вашим сервером. Если у вас нет доменного имени, вместо него можно указать публичный IP-адрес.

```
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 []:
```

Отредактируйте файл конфигурации Apache SSL (`/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
```

Перезапустите службу Apache, чтобы применить изменения:

```
systemctl restart httpd.service
```

**3. Веб-сервер**

Обзор

В этом разделе приведены рекомендации по настройке веб-сервера безопасным образом.

Принудительное перенаправление корневого URL на использование SSL

На системах на основе RHEL, добавьте виртуальный хост в файл конфигурации Apache (`/etc/httpd/conf/httpd.conf`) и настройте постоянную переадресацию для корневого каталога на Zabbix SSL URL. Обратите внимание, что `example.com` следует заменить на фактическое имя сервера.

```
####         :

<VirtualHost *:*>
    ServerName example.com
    Redirect permanent / https://example.com
</VirtualHost>
```

Перезапустите службу Apache, чтобы применить изменения:

```
systemctl restart httpd.service
```

Включение строгой транспортной безопасности HTTP (HSTS) на веб-сервере

Чтобы защитить веб-интерфейс Zabbix от атак, понижающих версию протокола, мы рекомендуем включить HSTS политику на веб-сервере.

Чтобы включить HSTS политику для веб-интерфейса вашего Zabbix в конфигурации Apache, выполните седующие шаги:

1. Найдите файл конфигурации виртуальных хостов:

- `/etc/httpd/conf/httpd.conf` на системах на базе RHEL
- `/etc/apache2/sites-available/000-default.conf` на Debian/Ubuntu

2. Добавьте следующую директиву к файл конфигурации ваших виртуальных хостов:

```
<VirtualHost *:*>
    Header set Strict-Transport-Security "max-age=31536000"
</VirtualHost>
```

3. Перезапустите службу Apache, чтобы применить изменения:

```
####              RHEL:
systemctl restart httpd.service
```

```
####   Debian/Ubuntu:
systemctl restart apache2.service
```

Принудительное применение безопасных и SameSite сеансовых файлов cookie в Zabbix

При настройке Zabbix важно принудительно применять атрибуты secure и SameSite для сеансовых файлов cookie для повышения безопасности и предотвращения атак с подделкой межсайтовых запросов (CSRF). Однако, принудительное применение `SameSite=Strict` может вызвать проблемы в определённых сценариях, такие как:

- Виджеты URL панели отображают «пользователь не вошел в систему» (”user not logged in”) при встраивании фреймов (iframes) того же домена.
- Пользователи, получающие доступ к панели через HTTP вместо HTTPS, могут столкнуться с проблемами входа.
- Невозможность поделиться URL-адресами к определённым разделам меню Zabbix или хостам.

Чтобы смягчить эти проблемы, пользователи должны иметь возможность настроить политику SameSite.

1. Безопасные файлы cookie

Установка флага `secure` гарантирует, что файлы cookie будут передаваться только по HTTPS, предотвращая раскрытие информации при незашифрованных соединениях.

Чтобы включить безопасные файлы cookie в Zabbix, добавьте или измените следующую настройку в конфигурации веб-сервера:

Для Apache:

```
Header always edit Set-Cookie ^(.*)$ $1;Secure
```

Для Nginx:

```
proxy_cookie_path / "/; Secure";
```

Убедитесь, что ваш веб-интерфейс Zabbix доступен через HTTPS; в противном случае файлы cookie с флагом `Secure` не будут отправляться.

2. Настройка атрибута SameSite

Настройки веб-сервера также могут принудительно применять атрибут SameSite:

Для Apache:

```
<IfModule mod_headers.c>
Header onsuccess edit Set-Cookie (.*) "$1; SameSite=Strict"
</IfModule>
```

Для Nginx (версия 1.19.3+):

```
proxy_cookie_flags ~ samesite=Strict; #       «~»    «zbx_session»
```

Включение политики безопасности контента (CSP) на веб-сервере

Чтобы защитить интерфейс Zabbix от межсайтового скриптинга (Cross Site Scripting, XSS), внедрения данных (data injection) и подобных атак, мы рекомендуем включить политику безопасности контента (Content Security Policy, CSP) на веб-сервере. Для этого настройте веб-сервер на возврат HTTP-заголовка [en].

**Attention:**
Следующая конфигурация заголовка CSP предназначена только для установки веб-интерфейса Zabbix по умолчанию и для случаев, когда весь контент происходит из домена сайта (за исключением поддоменов). Другая конфигурация заголовка CSP может потребоваться, если вы, например, настраиваете виджет *URL* для отображения контента из поддоменов сайта или внешних доменов, переключаете с *OpenStreetMap* на другой картографический движок или добавляете внешние CSS или виджеты. Если вы используете метод многофакторной аутентификации Duo Universal Prompt, обязательно добавьте «duo.com» в директиву CSP в файле конфигурации вашего виртуального хоста.

Чтобы включить CSP для вашего веб-интерфейса Zabbix в конфигурации Apache, выполните следующие действия:

1. Найдите ваш файл конфигурации виртуальных хостов:

- `/etc/httpd/conf/httpd.conf` на системах на базе RHEL
- `/etc/apache2/sites-available/000-default.conf` на Debian/Ubuntu

2. Добавьте следующую директиву в ваш файл конфигурации виртуальных хостов:

```
<VirtualHost *:*>
    Header set Content-Security-Policy: "default-src 'self' *.openstreetmap.org; script-src 'self' 'unsafe
</VirtualHost>
```

3. Перезапустите службу Apache, чтобы применить изменения:

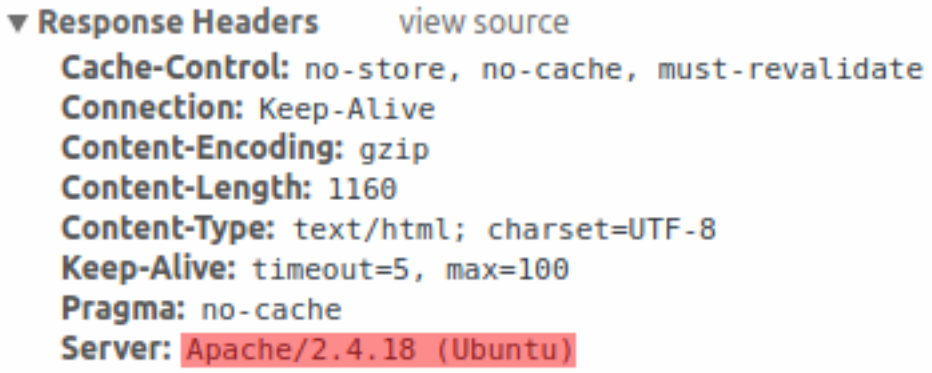
```
####              RHEL:
systemctl restart httpd.service

####   Debian/Ubuntu
systemctl restart apache2.service
```

Отключение отображения информации о веб-сервере

Чтобы улучшить безопасность, рекомендуется отключить все подписи веб-сервера.

To improve security, disabling all web server signatures is recommended.

По умолчанию веб-сервер раскрывает подпись программного обеспечения:




Эту подпись можно отключить, добавив следующие параметры в файл конфигурации Apache:

```
ServerSignature Off
ServerTokens Prod
```

Подпись PHP (HTTP заголовок X-Powered-By ) можно отключить, изменив файл конфигурации `php.ini` (подпись отключена по умолчанию):

```
expose_php = Off
```

Чтобы изменения файлов конфигурации вступили в силу, необходимо перезапустить веб-сервер.

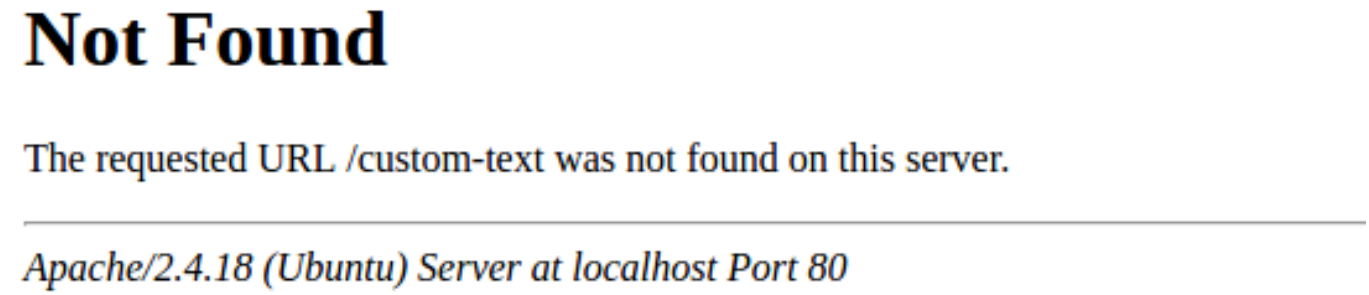
Дополнительного уровня безопасности можно достичь, используя *mod_security* (пакет *libapache2-mod-security2*) с Apache. mod_security позволяет полностью удалить подпись сервера вместо удаления лишь версии из подписи сервера. После установки *mod_security*, подпись можно изменить на любое значение, исправив ”SecServerSignature” на любое желаемое значение.

Пожалуйста, обратитесь к документации по вашему веб-серверу для того, чтобы узнать, как удалять/изменять подписи к программному обеспечению.

Отключение страниц ошибок веб-сервера по умолчанию

Чтобы избежать раскрытия информации, рекомендуется отключить страницы ошибок по умолчанию.

По умолчанию веб-сервер использует встроенные страницы ошибок:



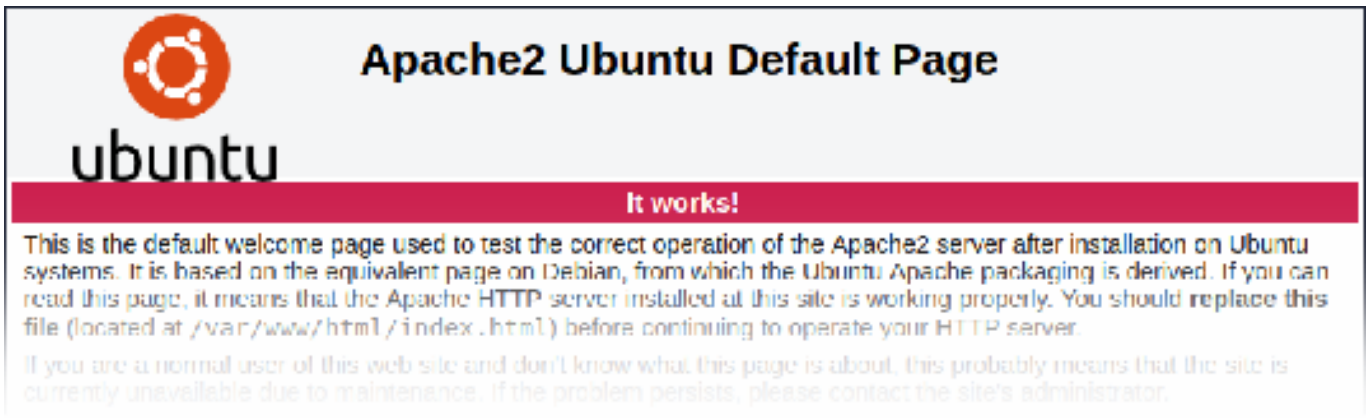

Эти страницы ошибок по умолчанию следует заменить/удалить. Например, для веб-сервера Apache можно использовать директиву ”ErrorDocument” для определения пользовательской страницы ошибок/текста.

Пожалуйста, обратитесь к документации по вашему веб-серверу для того, чтобы узнать как заменять/удалять используемые по умолчанию страницы ошибок.

Удаление тестовой страницы веб-сервера

Чтобы избежать раскрытия информации, рекомендуется удалить тестовую страницу веб-сервера.

По умолчанию, корневой каталог веб-сервера Apache содержит тестовую страницу с именем `index.html`:




Пожалуйста, обратитесь к документации вашего веб-сервера, чтобы узнать, как удалить тестовые страницы по умолчанию.

Настройка заголовка HTTP ответа X-Frame-Options

По умолчанию, в конфигурации Zabbix параметр *Использовать HTTP заголовок X-Frame-Options* имеет значение `SAMEORIGIN`. Это означает, что содержимое может быть загружено только во фрейм с таким же источником, как и у самой страницы.

Элементы веб-интерфейса Zabbix, которые извлекают содержимое с внешних URL-адресов (а именно, виджет панели URL), отображают полученное содержимое в так называемой песочнице (sandbox), при этом все ограничения песочницы включены.

Такие настройки улучшают безопасность веб-интерфейса Zabbix и обеспечивают защиту от XSS и кликджекинг атак. Пользователи *Супер-администраторы* могут изменить параметры *Использовать sandbox атрибут в iframe* and *Использовать HTTP заголовок X-Frame-Options* по мере необходимости. Пожалуйста, тщательно взвесьте риски и преимущества перед изменением настроек по умолчанию. Отключать полностью песочницу iframe или HTTP-заголовок X-Frame-Options не рекомендуется.

Скрытие файла со списком распространённых паролей

Для повышения сложности атак методом перебора пароля, рекомендуется ограничить доступ к файлу `ui/data/top_passwords.tx`
Данном файл содержит список самых распространённых и контекстно-зависимых паролей и не даёт пользователям устанавливать такие пароли в случае, если параметр *Избегать легко угадываемых паролей* включён в политике паролей.

Чтобы ограничить доступ к файлу `top_passwords.txt`, измените конфигурацию веб-сервера.

В Apache доступ к файлам можно ограничить с помощью файла `.htaccess`:

```
<Files "top_passwords.txt">
    Order Allow,Deny
    Deny from all
</Files>
```

В NGINX доступ к файлам можно ограничить с помощью директивы `location`:

```
location = /data/top_passwords.txt {
    deny all;
    return 404;
}
```

**2 Лучшие практики настройки**

Обзор

В этом разделе описаны лучшие практики настройки Zabbix для достижения оптимальной производительности и удобства использования. Рекомендации основаны на советах разработчиков Zabbix и практическом опыте тренеров и инженеров поддержки Zabbix.

Каждая установка Zabbix уникальна, и некоторые из этих рекомендаций могут не подходить для вашей конкретной конфигурации. Однако рекомендуется по возможности придерживаться этих рекомендаций, чтобы избежать распространенных потенциальных проблем.

**Узлы сети и элементы данных** Определение узла сети

Узел сети в Zabbix — это не физическая машина или устройство, а логическая сущность. В целях мониторинга вы можете создавать отдельные узлы сети для базы данных или, например, виртуальной машины. Либо вы можете создать общий узел сети *ноутбук Джона* и отслеживать все метрики в рамках этого узла сети.

Лучшая практика — создавать отдельный узел сети для каждого независимого экземпляра, такого как виртуальная машина, база данных, контейнер или сетевой коммутатор. Используя такой подход, вы сможете:

1. Избежать загромождения данных мониторинга за счет отдельных элементов данных, триггеров и уведомлений о проблемах для каждого узла сети.
2. Точно настраивать уровни доступа пользователей. Вы можете настроить роли пользователей, чтобы предоставлять доступ только к просмотру и/или настройке определенных узлов сети. См. также принцип наименьших привилегий.

Узлы сети с дублирующимися элементами данных

Если у вас есть несколько похожих узлов сети, например *Network switch 1* и *Network switch 2*, Zabbix предоставляет несколько способов быстро создать узел сети заново. Вы можете просто клонировать узел сети со всеми его метриками, нажав кнопку *Clone*, но в этом случае для последующего обновления элемента данных вам придется делать это вручную на каждом узле сети.

Лучшей практикой является создание шаблона со всеми необходимыми метриками, например *Network switch template*. Затем сгруппируйте похожие узлы сети в группу узлов сети, в приведенном выше примере это может быть *Network switches*. Теперь в разделе *Data Collection -> Hosts* вы можете отфильтровать все узлы сети по группе узлов сети и использовать кнопку *Mass update*, чтобы привязать шаблон ко всем вашим сетевым коммутаторам.

Зависимые элементы данных

Чтобы минимизировать количество запросов к целевому объекту, Zabbix позволяет создавать мастер-элементы и зависимые элементы данных. В этом случае мастер-элемент собирает большой набор информации одним запросом. Затем зависимые элементы данных можно настроить так, чтобы они извлекали конкретные фрагменты данных из этого набора с помощью предварительной обработки и сохраняли их как отдельные метрики.

Например, мастер-элемент может собирать ответ JSON или XML, содержащий несколько метрик, либо выполнять запрос к базе данных, который возвращает несколько столбцов данных (например, число открытых соединений, прерванных соединений, максимально допустимое количество одновременных соединений и общее накопленное число соединений с момента запуска), а зависимые элементы данных будут разбирать и сохранять каждое требуемое значение отдельно.

Лучшая практика для такой конфигурации — удалять историю мастер-элемента сразу после сбора и сохранять только данные зависимых элементов данных.

Серверы и прокси

Если все узлы сети находятся в одной локальной сети с сервером Zabbix и нет требований к масштабируемости или производительности, прокси может не понадобиться. В более крупных или сложных средах прямого мониторинга узлов сети с помощью сервера Zabbix может быть недостаточно. Добавление прокси и назначение части узлов сети этому прокси позволяет более равномерно распределить нагрузку.

Лучшей практикой является добавление прокси Zabbix, когда:

1. Вы отслеживаете несколько узлов сети, используя различные методы сбора метрик, за межсетевым экраном. Прокси будет собирать данные с узлов сети и пересылать их на сервер Zabbix, уменьшая необходимость открывать несколько портов на межсетевом экране.
2. Вы отслеживаете удаленные площадки, филиалы и/или сети. В случае нарушения сетевого соединения между сервером Zabbix и вашими удаленными площадками прокси Zabbix, развернутые в удаленных местах, продолжат сбор данных и отправят собранные данные обратно на сервер Zabbix, как только сетевое соединение будет восстановлено.
3. У вас крупномасштабное развертывание, и вы хотите снизить нагрузку на сервер Zabbix и повысить производительность. Определение крупномасштабного развертывания очень широкое и зависит не только от количества узлов сети, но и от количества значений, собираемых в секунду.

Секретные макросы

Вы можете использовать пользовательские макросы secret либо как секретный текст, либо как секретные макросы хранилища.

Для повышения безопасности при использовании секретных макросов хранилища рекомендуется настроить получение значений макросов сервером Zabbix и прокси Zabbix независимо друг от друга. По умолчанию значения секретных макросов получают сервер Zabbix, а затем они передаются прокси Zabbix.

## 20. API

**Обзор** Zabbix API позволяет программно получать и изменять конфигурацию Zabbix, а также предоставляет доступ к историческим данным. Она широко используется для:

- создания новых приложений для работы с Zabbix;
- интеграции Zabbix в стороннее программное обеспечение;
- автоматизации рутинных задач.

Zabbix API — это API на основе HTTP, и оно поставляется как часть веб-интерфейса. Оно использует протокол JSON-RPC 2.0, что означает две вещи:

- API состоит из набора отдельных методов;
- запросы и ответы между клиентами и API кодируются с использованием формата JSON.

Дополнительные сведения о протоколе и JSON см. в спецификации JSON-RPC 2.0 и на главной странице формата JSON.

Дополнительные сведения об интеграции функциональности Zabbix в ваши приложения Python см. в библиотеке Python для Zabbix.

> **Note:**
> Доступ пользователей в Zabbix, включая как конфигурацию, так и исторические данные, зависит от типа пользователя, назначенной роли пользователя и групп пользователей.

**Структура** API состоит из ряда методов, которые условно сгруппированы в отдельные API. Каждый метод выполняет одну отдельную задачу. Например, метод `host.create` относится к API *узла сети* и используется для создания новых узлов сети. Исторически для API иногда используется название «классы».

> **Note:**
> Большинство API содержит по крайней мере четыре метода: `get`, `create`, `update` и `delete` для получения, создания, обновления и удаления данных соответственно, однако некоторые API могут предоставлять совершенно другой набор методов.

**Выполнение запросов** После настройки веб-интерфейса вы можете использовать удаленные HTTP-запросы для вызова API. Для этого нужно отправлять HTTP POST-запросы к файлу `api_jsonrpc.php`, расположенному в каталоге веб-интерфейса. Например, если ваш веб-интерфейс Zabbix установлен в `https://example.com/zabbix`, HTTP-запрос для вызова метода `apiinfo.version` может выглядеть так:

```
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}'
```

Запрос должен содержать заголовок `Content-Type` со значением `application/json-rpc`, `application/json` или `application/jsonrequest`.

Объект запроса должен содержать следующие свойства:

- `jsonrpc` - версия протокола JSON-RPC, используемая API (Zabbix API реализует версию JSON-RPC 2.0);
- `method` - вызываемый метод API;
- `params` - параметры, которые будут переданы методу API;
- `id` - произвольный идентификатор запроса (если не указан, API рассматривает запрос как уведомление).

Если запрос корректен, ответ, возвращаемый API, должен выглядеть так:

```
{
    "jsonrpc": "2.0",
    "result": "7.4.0",
    "id": 1
}
```

Объект ответа, в свою очередь, содержит следующие свойства:

- `jsonrpc` - версия протокола JSON-RPC;
- `result` - данные, возвращенные методом;
- `id` - идентификатор соответствующего запроса.

**Пример рабочего процесса** В следующем разделе приведено несколько примеров использования более подробно.

**Аутентификация** Для доступа к любым данным в Zabbix необходимо:

- либо использовать существующий токен API (созданный в веб-интерфейсе Zabbix или с помощью Token API);
- либо использовать токен аутентификации, полученный методом user.login.

Например, если вы хотите получить новый токен аутентификации, войдя в систему как стандартный пользователь *Admin*, запрос JSON будет наподобие следующего:

```
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}
```

Если вы правильно указали учётные данные, ответ API должен содержать токен аутентификации пользователя:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

**Методы авторизации** Через заголовок "Authorization"

Все запросы к API требуют аутентификации или API-токена. Вы можете передать учетные данные, используя заголовок Authorization в запросе:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer 0424bd59b807674191e7d77572075f33'
```

> **Attention:**
> Если у вас возникают проблемы с аутентификацией, см. переадресацию заголовка Authorization.

API Zabbix принимает заголовки без учета регистра (например, `authorization`, `Authorization` и `AUTHORIZATION` обрабатываются одинаково).

Заголовок Authorization поддерживается в междоменных запросах (CORS).

По cookie-файлам Zabbix

Cookie *"zbx_session"* используется для авторизации запроса API из веб-интерфейса Zabbix, выполняемого при помощи JavaScript (из модуля или пользовательского виджета).

**Получение узлов сети** Теперь у вас есть действительный токен аутентификации пользователя (в примерах ниже он представлен как переменная), который можно использовать для доступа к данным в Zabbix. Например, вы можете использовать метод host.get, чтобы получить идентификаторы, имена узлов сети и интерфейсы всех настроенных узлов сети:

Запрос:

```
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` - это файл, содержащий JSON-запрос. Вместо файла вы можете передать запрос в аргументе `--data`.

data.json

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": [
            "hostid",
            "host"
        ],
        "selectInterfaces": [
            "interfaceid",
            "ip"
        ]
    },
    "id": 2
}
```

Объект ответа будет содержать запрошенные данные об узлах сети:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "interfaces": [
                {
                    "interfaceid": "1",
                    "ip": "127.0.0.1"
                }
            ]
        }
    ],
    "id": 2
}
```

**Note:**
Из соображений производительности всегда рекомендуется перечислять свойства объекта, которые вы хотите получить. Таким образом вы избежите получения всех данных.

**Создание нового элемента данных** Создайте новый элемент данных на узле сети «Zabbix server», используя данные, полученные из предыдущего запроса`host.get`. Это можно сделать при помощи метода 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_":
```

Успешный ответ будет содержать ID только что созданного элемента данных, который можно использовать для обозначения этого элемента данных в последующих запросах:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 3
}
```

**Note:**
Метод `item.create`, так же как и другие *методы создания*, может принимать массивы объектов и создавать несколько элементов данных с помощью одного вызова API.

**Создание нескольких триггеров** Соответственно, если *методы создания* принимают массивы, вы можете добавить несколько триггеров, например:

```
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
```

Успешный ответ будет содержать ID только что созданных триггеров:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 4
}
```

**Обновление элемента данных** Активируйте элемент данных, указав для «status» значение «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}'
```

Успешный ответ будет содержать ID обновлённого элемента данных:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 5
}
```

**Note:**
Метод `item.update`, так же как и другие *методы обновления*, тоже может принимать массивы объектов и обновлять несколько элементов данных за один вызов API.

**Обновление нескольких триггеров** Активируйте несколько триггеров, указав для «status» значение «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
```

Успешный ответ будет содержать ID обновлённых триггеров:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938",
            "13939"
        ]
    },
    "id": 6
}
```

**Note:**
Это предпочтительный метод обновления. Некоторые методы API, такие как `host.massupdate`, позволяют писать более простой код; однако, не рекомендуется использовать эти методы, так как они будут удалены в будущих релизах.

**Обработка ошибок** До этого момента всё, что мы пробовали, работало прекрасно. Но что может случиться, если мы попробуем выполнить некорректный запрос к API? Давайте попробуем создать ещё один узел сети при помощи вызова host.create, но не станем указывать обязательный параметр `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,
```

Тогда ответ будет содержать сообщение об ошибке:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32602,
        "message": "Invalid params.",
        "data": "No groups for host \"Linux server\"."
    },
    "id": 7
}
```

Если произошла ошибка, то вместо свойства `result`, объект ответа будет содержать свойство `error` со следующими данными:

- `code` — код ошибки;

- `message` — короткое описание ошибки;
- `data` — более подробное сообщение об ошибке.

Ошибки могут возникать в разных ситуациях, таких как некорректные входные значения, превышение времени ожидания сессии или попытка доступа к несуществующим объектам. Ваше приложение должно быть способно корректно обработать такие виды ошибок.

**Версии API** Чтобы упростить версионность API, начиная с Zabbix 2.0.4, версия API совпадает с версией самого Zabbix. Вы можете использовать метод apiinfo.version, чтобы узнать версию API, с которой вы работаете. Знание версии может пригодиться для корректировки вашего приложения, чтобы использовать возможности конкретной версии API.

Zabbix гарантирует обратную совместимость в пределах одной мажорной версии. При выполнении несовместимых изменений между мажорными выпусками, Zabbix обычно оставляет старые функции как устаревшие в следующим выпуске, и удаляет их только в выпуске, следующим за этим. Иногда Zabbix может удалить функции между мажорными выпусками без предоставления какой-либо обратной совместимости. Очень важно никогда не полагаться на какие-либо устаревшие функции и мигрировать на новые альтернативы как можно скорее.

> **Note:**
> Вы можете следить за всеми изменениями, внесёнными в API, на странице журнала изменений API.

**Дополнительная литература** Теперь вы знаете достаточно для начала работы с Zabbix API, но не останавливайтесь на данном этапе. Для дальнейшего чтения мы предлагаем вам взглянуть на список доступных API.

**Справочник методов**

В этом разделе содержится обзор функций, которые имеются в Zabbix API, который поможет вам найти варианты использования доступных классов и методов.

**Мониторинг** Zabbix API позволяет получать доступ к истории и другим данным, собранным в ходе мониторинга.

Панели мониторинга

Управляйте панелями мониторинга и создавайте на их основе запланированные отчеты.

Dashboard API | Template dashboard API | Report API

Кластер высокой доступности

Получить список узлов сервера и их статус.

API кластера высокой доступности

История

Получение исторических значений, собранных процессами мониторинга Zabbix, для отображения или дальнейшей обработки.

History API

Тренды

Получение значений трендов, вычисленных сервером Zabbix, для отображения или дальнейшей обработки.

Trend API

События

Получение событий, которые сгенерированы триггерами, сетевым обнаружением и другими системами Zabbix для более гибкого управления ситуациями или интеграции со сторонними инструментами.

API события

Проблемы

Получить проблемы в соответствии с заданными параметрами.

Problem API

Карты

Настройте карты, чтобы создавать подробные динамические представления вашей ИТ-инфраструктуры.

Map API

Задачи

Взаимодействие с менеджером задач Zabbix сервера, создание задач и получение ответа.

API задач

**Сервисы** Zabbix API позволяет получать доступ к данным, собранным во время мониторинга сервисов.

Соглашение об уровне обслуживания

Определяйте цели уровня обслуживания (SLO) и получайте подробную информацию о показателях уровня обслуживания (SLI) о производительности сервиса.

SLA API

Сервисы

Управляйте сервисами для мониторинга на уровне сервисов и получайте подробную информацию о SLA по любому сервису.

Service API

**Сбор данных** Zabbix API позволяет управлять конфигурацией вашей системы мониторинга.

Узлы сети и группы узлов сети

Manage host groups, hosts and everything related to them, including host interfaces, host macros and maintenance periods.

Управление группами узлов сети, узлами сети и всем, что с ними связано, включая интерфейсы узла сети, макросы узла сети и периоды обслуживания.

API узлов сети | API групп узлов сети | API интерфейсов узлов сети | API пользовательских макросов | API преобразования значений | API обслуживания

Элементы данных

Определяет элементы данных для мониторинга.

API элементов данных

Триггеры

Настройка триггеров для оповещения вас о наличии проблем в вашей системе. Управление зависимостями триггеров.

API триггеров

Графики

Редактируйте графики или отдельные элементы графика для более наглядного представления собранных данных.

Graph API | Graph item API

Шаблоны и группы шаблонов

Управляйте шаблонами и связывайте их с узлами сети или другими шаблонами.

Template API | Template group API | Value map API

Низкоуровневое обнаружение

Настройте правила низкоуровневого обнаружения, а также прототипы элементов данных, триггеров и графиков для мониторинга динамических сущностей.

API правила LLD | API прототипа элемента данных | API прототипа триггера | API прототипа графика | API прототипа узла сети

Корреляция событий

Создавайте пользовательские правила корреляции событий.

Correlation API

Обнаружение сети

Управляйте правилами обнаружения на уровне сети, чтобы автоматически находить и контролировать новые узлы сети. Получите полный доступ к информации об обнаруженных службах и узлах сети.

API правила обнаружения | API проверки обнаружения | API обнаруженного узла сети | API обнаруженной службы

Экспорт и импорт

Экспорт и импорт данных конфигурации Zabbix для резервного копирования конфигурации, миграции или массового обновления конфигурации.

Configuration API

Веб-мониторинг

Настройте веб-сценарии для мониторинга ваших веб-приложений и сервисов.

API веб-сценариев

**Оповещения** Zabbix API позволяет управлять действиями и оповещениями вашей системы мониторинга.

Действия и оповещения

Определяйте действия и операции, чтобы уведомлять пользователей о определенных событиях или автоматически выполнять удаленные команды. Получайте доступ к информации о сгенерированных оповещениях и их получателях.

Action API | Alert API

Типы оповещений

Настройте типы оповещений и несколько способов, с помощью которых пользователи будут получать уведомления.

API типа оповещений

Скрипты

Настраивайте и выполняйте скрипты, чтобы помочь себе в повседневных задачах.

Script API

**Пользователи** Zabbix API позволяет управлять пользователями вашей системы мониторинга.

Пользователи и группы пользователей

Добавьте пользователей, которые будут иметь доступ к Zabbix, назначьте их в группы пользователей и предоставьте разрешения. Создавайте роли для детального управления правами пользователей.

User API | User group API | User directory API | User role API

API-токены

Управление токенами авторизации.

Token API

Аутентификация

Измените параметры конфигурации аутентификации.

API аутентификации

**Администрирование** С помощью Zabbix API вы можете изменять настройки администрирования вашей системы мониторинга.

Общие сведения

Изменение некоторых глобальных параметров конфигурации.

API автрорегистрации | API карты значков | API изображений | API настроек | API регулярных выражений | API модулей | API коннекторов

Журнал аудита

Отслеживайте изменения конфигурации, внесенные каждым пользователем.

API журнала аудита

Обслуживание

Настройте обслуживание.

Housekeeping API

Прокси и группы прокси

Управляйте прокси, используемыми в вашей распределенной системе мониторинга.

Proxy API | Proxy group API

Макросы

Управление макросами.

API пользовательских макросов

**Информация об API** Получите версию API Zabbix, чтобы ваше приложение могло использовать функции, зависящие от версии.

API info API

### Action

Этот класс предназначен для работы с действиями.

Ссылки на объекты:

- Action
- Action operation
  - Action operation message
  - Action operation condition
- Action recovery operation
- Action update operation
- Action filter
  - Action filter condition

Доступные методы:

- action.create - создать новые действия
- action.delete - удалить действия
- action.get - получить действия
- action.update - обновить действия

#### Объект действия

Следующие объекты напрямую связаны с методом API `action`.

Действие

Объект действия имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| actionid | ID | ID действия.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| esc_period | string | Длительность шага операции по умолчанию. Должна быть не менее 60 секунд. Допускаются секунды, единица времени с суффиксом или пользовательский макрос.<br><br>Поведение свойства:<br>- *поддерживается*, если `eventsource` установлено в ”event created by a trigger”, ”internal event” или ”event created on service status update” |

| Property | Type | Description |
|---|---|---|
| eventsource | integer | Тип событий, которые будет обрабатывать действие.<br><br>См. свойство `source` события для списка поддерживаемых типов событий.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |
| name | string | Имя действия.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| status | integer | Включено действие или отключено.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включено;<br>1 - отключено. |
| pause_symptoms | integer | Следует ли приостанавливать эскалацию, если событие является симптомом.<br><br>Возможные значения:<br>0 - Не приостанавливать эскалацию для проблем-симптомов;<br>1 - *(по умолчанию)* Приостанавливать эскалацию для проблем-симптомов.<br><br>Поведение свойства:<br>- *поддерживается*, если `eventsource` установлено в ”event created by a trigger” |
| pause_suppressed | integer | Следует ли приостанавливать эскалацию во время периодов обслуживания.<br><br>Возможные значения:<br>0 - Не приостанавливать эскалацию;<br>1 - *(по умолчанию)* Приостанавливать эскалацию.<br><br>Поведение свойства:<br>- *поддерживается*, если `eventsource` установлено в ”event created by a trigger” |

| Property | Type | Description |
|---|---|---|
| notify_if_canceled | integer | Следует ли уведомлять об отмене эскалации.<br><br>Возможные значения:<br>0 - Не уведомлять об отмене эскалации;<br>1 - *(по умолчанию)* Уведомлять об отмене эскалации.<br><br>Поведение свойства:<br>- *поддерживается*, если `eventsource` установлено в ”event created by a trigger” |

Объект операции действия

Объект операции действия определяет операцию, которая будет выполнена при запуске действия. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| operationtype | integer | Тип операции.<br><br>Возможные значения:<br>0 - отправить сообщение;<br>1 - глобальный скрипт;<br>2 - добавить узел сети;<br>3 - удалить узел сети;<br>4 - добавить в группу узлов сети;<br>5 - удалить из группы узлов сети;<br>6 - привязать шаблон;<br>7 - отвязать шаблон;<br>8 - включить узел сети;<br>9 - отключить узел сети;<br>10 - задать режим инвентаризации узла сети;<br>13 - добавить теги узла сети;<br>14 - удалить теги узла сети.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”event created by a trigger” или ”event created on service status update”:<br>0 - ”send message”;<br>1 - ”global script”.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”internal event”:<br>0 - ”send message”.<br><br>Поведение свойства:<br>- *required* |
| esc_period | string | Длительность шага эскалации в секундах. Должна быть больше 60 секунд. Допускаются секунды, единица времени с суффиксом или пользовательский макрос. Если задано 0 или 0s, будет использован период эскалации действия по умолчанию.<br><br>По умолчанию: 0s.<br><br>Поведение свойства:<br>- *supported* если `eventsource` объекта Action object установлен в ”event created by a trigger”, ”internal event” или ”event created on service status update” |

| Property | Type | Description |
|---|---|---|
| esc_step_from | integer | Шаг, с которого начинается эскалация.<br><br>По умолчанию: 1.<br><br>Поведение свойства:<br>- *supported* если `eventsource` объекта Action object установлен в ”event created by a trigger”, ”internal event” или ”event created on service status update” |
| esc_step_to | integer | Шаг, на котором заканчивается эскалация.<br><br>По умолчанию: 1.<br><br>Поведение свойства:<br>- *supported* если `eventsource` объекта Action object установлен в ”event created by a trigger”, ”internal event” или ”event created on service status update” |
| evaltype | integer | Метод вычисления условий операции.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И/ИЛИ;<br>1 - И;<br>2 - ИЛИ. |
| opcommand | object | Глобальный скрипт для выполнения.<br><br>Глобальный скрипт должен содержать свойство `scriptid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”global script” |
| opcommand_grp | array | Группы узлов сети, на которых будут выполняться глобальные скрипты.<br><br>Группы узлов сети должны содержать свойство `groupid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”global script” и `opcommand_hst` не задан |
| opcommand_hst | array | Узел сети, на котором будут выполняться глобальные скрипты.<br><br>Узлы сети должны содержать свойство `hostid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”global script” и `opcommand_grp` не задан |
| opconditions | array | Условия операции, используемые для действий триггера.<br><br>Объект условия операции подробно описан ниже. |
| opgroup | array | Группы узлов сети, в которые будут добавляться узлы сети.<br><br>Группы узлов сети должны содержать свойство `groupid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”add to host group” или ”remove from host group” |
| opmessage | object | Объект, содержащий данные о сообщении, отправляемом операцией.<br><br>Объект сообщения операции подробно описан ниже.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”send message” |

| Property | Type | Description |
|---|---|---|
| opmessage_grp | array | Группы пользователей, которым будут отправляться сообщения.<br><br>Группы пользователей должны содержать свойство `usrgrpid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”send message” и `opmessage_usr` не задан |
| opmessage_usr | array | Пользователи, которым будут отправляться сообщения.<br><br>Пользователи должны содержать свойство `userid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”send message” и `opmessage_grp` не задан |
| optemplate | array | Шаблоны, которые будут привязаны к узлам сети.<br><br>Шаблоны должны содержать свойство `templateid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”link template” или ”unlink template” |
| opinventory | object | Режим инвентаризации, который будет задан узлу сети.<br><br>Инвентаризация должна содержать свойство `inventory_mode`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”set host inventory mode” |
| optag | array | Теги узла сети, которые будут добавлены или удалены.<br><br>Теги должны содержать свойство `tag`.<br>Свойство `value` необязательно.<br><br>Поведение свойства:<br>- *supported* если `operationtype` установлен в ”add host tags” или ”remove host tags”. |

Сообщение операции действия

Объект сообщения операции содержит данные о сообщении, которое будет отправлено операцией. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| default_msg | integer | Использовать ли текст и тему сообщения действия по умолчанию.<br><br>Возможные значения:<br>0 - использовать данные из операции;<br>1 - *(по умолчанию)* использовать данные из типа медиа. |
| mediatypeid | ID | ID типа медиа, который будет использоваться для отправки сообщения.<br><br>Поведение свойства:<br>- *поддерживается*, если `operationtype` объекта операции действия, объекта операции восстановления действия или объекта операции обновления действия установлен в ”send message”, либо если `operationtype` объекта операции обновления действия установлен в ”notify all involved” |

| Property | Type | Description |
|---|---|---|
| message | string | Текст сообщения операции.<br><br>Поведение свойства:<br>- *поддерживается*, если `default_msg` установлен в ”use the data from the operation” |
| subject | string | Тема сообщения операции.<br><br>Поведение свойства:<br>- *поддерживается*, если `default_msg` установлен в ”use the data from the operation” |

Условие операции действия

Объект условия операции действия определяет условие, которое должно быть выполнено для выполнения текущей операции. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| conditiontype | integer | Тип условия.<br><br>Возможные значения:<br>14 - событие подтверждено.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение для сравнения.<br><br>Поведение свойства:<br>- *required* |
| operator | integer | Оператор условия.<br><br>Возможные значения:<br>0 - *(по умолчанию)* = |

Для каждого типа условия операции поддерживаются следующие операторы и значения.

| Condition | Condition name | Supported operators | Expected value |
|---|---|---|---|
| 14 | Событие подтверждено | = | Подтверждено ли событие.<br><br>Возможные значения:<br>0 - не подтверждено;<br>1 - подтверждено. |

Операция восстановления действия

Объект операции восстановления действия определяет операцию, которая будет выполнена, когда проблема будет устранена. Операции восстановления возможны **только** для действий типа триггер, внутреннее событие и сервис. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| operationtype | integer | Тип операции.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”event created by a trigger” или ”event created on service status update”:<br>0 - отправить сообщение;<br>1 - глобальный скрипт;<br>11 - уведомить всех участников.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”internal event”:<br>0 - отправить сообщение;<br>11 - уведомить всех участников.<br><br>Поведение свойства:<br>- *required* |
| opcommand | object | Глобальный скрипт для выполнения.<br><br>Глобальный скрипт должен иметь определенное свойство `scriptid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”global script” |
| opcommand_grp | array | Группы узлов сети, на которых запускать глобальные скрипты.<br><br>Группы узлов сети должны иметь определенное свойство `groupid`.<br><br>Поведение свойства:<br>- *required* если `eventsource` объекта Action object установлен в ”event created by a trigger”, и `operationtype` установлен в ”global script”, и `opcommand_hst` не задан |
| opcommand_hst | array | Узел сети, на котором запускать глобальные скрипты.<br><br>Узлы сети должны иметь определенное свойство `hostid`.<br><br>Поведение свойства:<br>- *required* если `eventsource` объекта Action object установлен в ”event created by a trigger”, и `operationtype` установлен в ”global script”, и `opcommand_grp` не задан |
| opmessage | object | Объект, содержащий данные о сообщении, отправляемом операцией восстановления.<br><br>Объект сообщения операции подробно описан выше.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”send message” |
| opmessage_grp | array | Группы пользователей, которым отправлять сообщения.<br><br>Группы пользователей должны иметь определенное свойство `usrgrpid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”send message” и `opmessage_usr` не задан |

| Property | Type | Description |
|---|---|---|
| opmessage_usr | array | Пользователи, которым отправлять сообщения.<br><br>Пользователи должны иметь определенное свойство `userid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлен в ”send message” и `opmessage_grp` не задан |

Операция обновления действия

Объект операции обновления действия определяет операцию, которая будет выполнена при обновлении проблемы (при добавлении комментария, подтверждении, изменении серьезности или ручном закрытии). Операции обновления возможны **только** для действий триггера и сервиса. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| operationtype | integer | Тип операции.<br><br>Возможные значения:<br>0 - отправить сообщение;<br>1 - глобальный скрипт;<br>12 - уведомить всех участников.<br><br>Поведение свойства:<br>- *required* |
| opcommand | object | Глобальный скрипт для выполнения.<br><br>Глобальный скрипт должен иметь определенное свойство `scriptid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлено в ”global script” |
| opcommand_grp | array | Группы узлов сети, на которых нужно запускать глобальные скрипты.<br><br>Группы узлов сети должны иметь определенное свойство `groupid`.<br><br>Поведение свойства:<br>- *required* если `eventsource` объекта Action object установлено в ”event created by a trigger”, и `operationtype` установлено в ”global script”, и `opcommand_hst` не задано |
| opcommand_hst | array | Узел сети, на котором нужно запускать глобальные скрипты.<br><br>Узлы сети должны иметь определенное свойство `hostid`.<br><br>Поведение свойства:<br>- *required* если `eventsource` объекта Action object установлено в ”event created by a trigger”, и `operationtype` установлено в ”global script”, и `opcommand_grp` не задано |
| opmessage | object | Объект, содержащий данные о сообщении, отправляемом операцией обновления.<br><br>Объект сообщения операции подробно описан выше. |

| Property | Type | Description |
|---|---|---|
| opmessage_grp | array | Группы пользователей, которым нужно отправлять сообщения.<br><br>Группы пользователей должны иметь определенное свойство `usrgrpid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлено в "send message" и `opmessage_usr` не задано |
| opmessage_usr | array | Пользователи, которым нужно отправлять сообщения.<br><br>Пользователи должны иметь определенное свойство `userid`.<br><br>Поведение свойства:<br>- *required* если `operationtype` установлено в "send message" и `opmessage_grp` не задано |

Фильтр действия

Объект фильтра действия определяет набор условий, которые должны быть выполнены для выполнения настроенных операций действия. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| conditions | array | Набор условий фильтра, используемых для фильтрации результатов. Условия будут отсортированы в порядке их размещения в формуле.<br><br>Поведение свойства:<br>- *required* |
| evaltype | integer | Метод вычисления условий фильтра.<br><br>Возможные значения:<br>0 - And/Or;<br>1 - And;<br>2 - Or;<br>3 - Пользовательское выражение.<br><br>Поведение свойства:<br>- *required* |
| eval_formula | string | Сгенерированное выражение, которое будет использоваться для вычисления условий фильтра. Выражение содержит идентификаторы, которые ссылаются на конкретные условия фильтра по их `formulaid`. Значение `eval_formula` равно значению `formula` для фильтров с пользовательским выражением.<br><br>Поведение свойства:<br>- *read-only* |
| formula | string | Определяемое пользователем выражение, используемое для вычисления условий фильтров с пользовательским выражением. Выражение должно содержать идентификаторы, которые ссылаются на конкретные условия фильтра по их `formulaid`. Идентификаторы, используемые в выражении, должны точно совпадать с идентификаторами, определенными в условиях фильтра: ни одно условие не может остаться неиспользованным или быть пропущенным.<br><br>Поведение свойства:<br>- *required* если `evaltype` установлено в "custom expression" |

Условие фильтра действия

Объект условия фильтра действия определяет конкретное условие, которое должно быть проверено перед выполнением операций действия.

| Property | Type | Description |
|---|---|---|
| conditiontype | integer | Тип условия.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”event created by a trigger”:<br>0 - группа узлов сети;<br>1 - узел сети;<br>2 - триггер;<br>3 - имя события;<br>4 - серьезность триггера;<br>6 - период времени;<br>13 - шаблон узла сети;<br>16 - проблема подавлена;<br>25 - тег события;<br>26 - значение тега события.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”event created by a discovery rule”:<br>7 - IP-адрес узла сети;<br>8 - тип обнаруженной службы;<br>9 - порт обнаруженной службы;<br>10 - статус обнаружения;<br>11 - длительность времени работы или простоя;<br>12 - полученное значение;<br>18 - правило обнаружения;<br>19 - проверка обнаружения;<br>20 - прокси;<br>21 - объект обнаружения.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”event created by active agent autoregistration”:<br>20 - прокси;<br>22 - имя узла сети;<br>24 - метаданные узла сети.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”internal event”:<br>0 - группа узлов сети;<br>1 - узел сети;<br>13 - шаблон узла сети;<br>23 - тип события;<br>25 - тег события;<br>26 - значение тега события.<br><br>Возможные значения, если `eventsource` объекта Action object установлен в ”event created on service status update”:<br>25 - тег события;<br>26 - значение тега события;<br>27 - служба;<br>28 - имя службы.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| value | string | Значение для сравнения.<br><br>Поведение свойства:<br>- *required* |
| value2 | string | Вторичное значение для сравнения.<br><br>Поведение свойства:<br>- *required* если `eventsource` объекта Action object установлен в "event created by a trigger", `conditiontype` установлен в любое возможное значение для действий триггера, а тип условия (см. ниже) равен "26"<br>- *required* если `eventsource` объекта Action object установлен в "internal event", `conditiontype` установлен в любое возможное значение для внутренних действий, а тип условия (см. ниже) равен "26"<br>- *required* если `eventsource` объекта Action object установлен в "event created on service status update", `conditiontype` установлен в любое возможное значение для действий службы, а тип условия (см. ниже) равен "26" |
| formulaid | string | Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но будет сгенерирован заново при последующем запросе. |
| operator | integer | Оператор условия.<br><br>Возможные значения:<br>0 - *(по умолчанию)* равно;<br>1 - не равно;<br>2 - содержит;<br>3 - не содержит;<br>4 - входит в;<br>5 - больше или равно;<br>6 - меньше или равно;<br>7 - не входит в;<br>8 - соответствует;<br>9 - не соответствует;<br>10 - Да;<br>11 - Нет. |

**Note:**
Чтобы лучше понять, как использовать фильтры с различными типами выражений, см. примеры на страницах методов action.get и action.create.

Для каждого типа условия поддерживаются следующие операторы и значения.

| Condition | Condition name | Supported operators | Expected value |
|---|---|---|---|
| 0 | Группа узлов сети | равно,<br>не равно | ID группы узлов сети. |
| 1 | Узел сети | равно,<br>не равно | ID узла сети. |
| 2 | Триггер | равно,<br>не равно | ID триггера. |
| 3 | Имя события | содержит,<br>не содержит | Имя события. |
| 4 | Серьезность триггера | равно,<br>не равно,<br>больше или равно,<br>меньше или равно | Серьезность триггера. Список поддерживаемых уровней серьезности триггера см. в свойстве `severity` триггера. |
| 5 | Значение триггера | равно | Значение триггера. Список поддерживаемых значений триггера см. в свойстве `value` триггера. |
| 6 | Период времени | входит в, не входит в | Время, когда событие было сгенерировано, в виде периода времени. |
| 7 | IP-адрес узла сети | равно,<br>не равно | Один или несколько диапазонов IP-адресов, разделенных запятыми. Дополнительные сведения о поддерживаемых форматах диапазонов IP-адресов см. в разделе конфигурации сетевого обнаружения. |
| 8 | Тип обнаруженной службы | равно,<br>не равно | Тип обнаруженной службы. Тип службы соответствует типу проверки обнаружения, использованной для обнаружения службы. Список поддерживаемых типов см. в свойстве `type` проверки обнаружения. |
| 9 | Порт обнаруженной службы | равно,<br>не равно | Один или несколько диапазонов портов, разделенных запятыми. |
| 10 | Статус обнаружения | равно | Статус обнаруженного объекта.<br><br>Возможные значения:<br>0 - узел сети или служба доступны;<br>1 - узел сети или служба недоступны;<br>2 - узел сети или служба обнаружены;<br>3 - узел сети или служба потеряны. |
| 11 | Длительность времени работы или простоя | больше или равно,<br>меньше или равно | Время в секундах, в течение которого обнаруженный объект находится в текущем статусе. |
| 12 | Полученные значения | равно,<br>не равно,<br>больше или равно,<br>меньше или равно,<br>содержит,<br>не содержит | Значение, возвращаемое при выполнении проверки обнаружения агентом Zabbix, SNMPv1, SNMPv2 или SNMPv3. |
| 13 | Шаблон узла сети | равно,<br>не равно | ID связанного шаблона. |
| 16 | Проблема подавлена | Да, Нет | Значение не требуется: использование оператора ”Да” означает, что проблема должна быть подавлена, ”Нет” - не подавлена. |
| 18 | Правило обнаружения | равно,<br>не равно | ID правила обнаружения. |
| 19 | Проверка обнаружения | равно,<br>не равно | ID проверки обнаружения. |
| 20 | Прокси | равно,<br>не равно | ID прокси. |
| 21 | Объект обнаружения | равно | Тип объекта, который вызвал событие обнаружения.<br><br>Возможные значения:<br>1 - обнаруженный узел сети;<br>2 - обнаруженная служба. |

| Condition | Condition name | Supported operators | Expected value |
|---|---|---|---|
| 22 | Имя узла сети | содержит,<br>не содержит,<br>соответствует,<br>не соответствует | Имя узла сети.<br>Использование регулярного выражения поддерживается для операторов *соответствует* и *не соответствует* в условиях авто-регистрации. |
| 23 | Тип события | равно | Конкретное внутреннее событие.<br><br>Возможные значения:<br>0 - элемент данных в состоянии ”не поддерживается”;<br>1 - элемент данных в состоянии ”нормально”;<br>2 - правило LLD в состоянии ”не поддерживается”;<br>3 - правило LLD в состоянии ”нормально”;<br>4 - триггер в состоянии ”неизвестно”;<br>5 - триггер в состоянии ”нормально”. |
| 24 | Метаданные узла сети | содержит,<br>не содержит,<br>соответствует,<br>не соответствует | Метаданные автоматически зарегистрированного узла сети.<br>Использование регулярного выражения поддерживается для операторов *соответствует* и *не соответствует*. |
| 25 | Тег | равно,<br>не равно,<br>содержит,<br>не содержит | Тег события. |
| 26 | Значение тега | равно,<br>не равно,<br>содержит,<br>не содержит | Значение тега события. |
| 27 | Служба | равно,<br>не равно | ID службы. |
| 28 | Имя службы | равно,<br>не равно | Имя службы. |

**action.create**

Описание

`object action.create( / actions)`

Этот метод позволяет создавать новые действия.

**Note:**
Этот метод доступен только типам пользователей *Администратор* и *Супер-администратор*. Разрешение на использование данного метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей

Параметры

`(object/array)` Действия для создания.

Дополнительно к стандартным свойствам действия, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| filter | object | Объект фильтра действия для действия. |
| operations | array | Операции действия для создания для действия. |
| recovery_operations | array | Операции восстановления действия для создания для действия. |
| update_operations | array | Операции обновления действия для создания для действия. |

Возвращаемые значения

(object) Возвращает объект, который содержит ID созданных действий под свойством actionids. Порядок возвращаемых ID соответствует порядку переданных действий.

Примеры

Создание действия триггера

Создать действие триггера, которое начнется, как только триггер (со словом «память» в названии) с хоста «10084» перейдет в состояние ПРОБЛЕМА. Действие будет иметь 4 настроенные операции. Первая и немедленная операция отправит сообщение всем пользователям в группе пользователей «7» через тип носителя «1». Если событие не будет устранено в течение 30 минут, вторая операция запустит script "5" (скрипт с областью действия "Операция действия") на всех хостах группы «2». Если событие устранено, операция восстановления уведомит всех пользователей, получивших сообщения о проблеме. Если событие обновлено, операция подтверждения/обновления уведомит (с настраиваемой темой и сообщением) всех пользователей, получивших какие-либо сообщения о проблеме.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "17"
        ]
    },
    "id": 1
}
```

Создание действия обнаружения

Создать действие обнаружения, которое свяжет шаблон «10001» с обнаруженными хостами.

Запрос:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Использование пользовательского фильтра выражений

Создать действие триггера, использующее пользовательское выражение "A и (B или C)" для оценки условий действия. Как только триггер с уровнем серьезности выше или равным "Предупреждение" с хоста "10084" или хоста "10106" переходит в состояние ПРОБЛЕМА, действие отправит сообщение всем пользователям в группе пользователей "7" через тип носителя "1". Идентификаторы формул "A", "B" и "C" были выбраны произвольно.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Создание правила автоматической регистрации агента

Создать действие авторегистрации, которое добавляет хост в группу хостов «2», если имя хоста содержит «SRV» или метаданные содержат ”AlmaLinux”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            19
        ]
    },
    "id": 1
}
```

Создание правила автрорегистрации агента с тегами узла сети

Создайте действие автрорегистрации, которое добавляет узел сети в группу узлов сети ”2” и добавляет два тега узла сети.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            20
        ]
    },
    "id": 1
}
```

См. также

- Фильтр действия
- Операция действия
- Скрипт

Источник

CAction::create() в *ui/include/classes/api/services/CAction.php*.

**action.delete**

Описание

`object action.delete(    actionIds)`

Этот метод позволяет удалять действия.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода могут быть отключены в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(    )` ID удаляемых действий.

Возвращаемые значения

`(object)` Возвращает объект, который содержит ID удаленных действий под свойством `actionids`.

Примеры

Удаление нескольких действий

Удалить два действия

Request:

```
{
 · "jsonrpc": "2.0",
 · "method": "action.delete",
 · "params": [
 · "17",
 · "18"
 · ],
 · "auth": "3a57200802b24cda67c4e4010b50c065",
 · "id": 1
}
```

Response:

```
{
 · "jsonrpc": "2.0",
 · "result": {
```

```
        "action ids": [
        "17",
        "18"
        ]
        },
        "id": 1
}
```

Источник

CAction::delete() в *ui/include/classes/api/services/CAction.php*.

**action.get**

Описание

`integer/array action.get(    parameters)`

Метод позволяет получать действия в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен всем типам пользователей. Разрешение на использование данного метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| actionids | ID/array | Возвращать только действия с указанными ID. |
| groupids | ID/array | Возвращать только действия, которые используют указанные группы узлов сети в условиях действия. |
| hostids | ID/array | Возвращать только действия, которые используют указанные узлы сети в условиях действия. |
| triggerids | ID/array | Возвращать только действия, которые используют указанные триггеры в условиях действия. |
| mediatypeids | ID/array | Возвращать только действия, которые используют указанные типы медиа для отправки сообщений. |
| usrgrpids | ID/array | Возвращать только действия, которые настроены на отправку сообщений указанным группам пользователей. |
| userids | ID/array | Возвращать только действия, которые настроены на отправку сообщений указанным пользователям. |
| scriptids | ID/array | Возвращать только действия, которые настроены на запуск указанных скриптов. |
| selectFilter | query | Возвращать свойство `filter` с фильтром условий действия. |
| selectOperations | query | Возвращать свойство `operations` с операциями действия. |

| Parameter | Type | Description |
|---|---|---|
| selectRecoveryOperations | query | Возвращать свойство `recovery_operations` с операциями восстановления действия. |
| selectUpdateOperations | query | Возвращать свойство `update_operations` с операциями обновления действия. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `actionid`, `name`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает:

- массив объектов

или - количество полученных объектов , если был использован параметр `countOutput`.

Примеры

Получение действий триггера

Получение всех настроенных действий триггера вместе с условиями и операциями действий.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
    "output": "extend",
    "selectOperations": "extend",
    "selectRecoveryOperations": "extend",
    "selectUpdateOperations": "extend",
    "selectFilter": "extend",
    "filter": {
    "eventsource": 0
    }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
    {
    "actionid": "3",
    "name": "Report problems to Zabbix administrators",
```

```
"eventsource": "0",
"status": "1",
"esc_period": "1h",
"pause_suppressed": "1",
"filter": {
"evaltype": "0",
"formula": "",
"conditions": [],
"eval_formula": ""
},
"operations": [
{
"operationid": "3",
"actionid": "3",
"operationtype": "0",
"esc_period": "0",
"esc_step_from": "1",
"esc_step_to": "1",
"evaltype": "0",
"opconditions": [],
"opmessage": [
{
"default_msg": "1",
"subject": "",
"message": "",
"mediatypeid" => "0"
}
],
"opmessage_grp": [
{
"usrgrpid": "7"
}
]
}
],
"recovery_operations": [
{
"operationid": "7",
"actionid": "3",
"operationtype": "11",
"evaltype": "0",
"opconditions": [],
"opmessage": {
"default_msg": "0",
"subject": "{TRIGGER.STATUS}: {TRIGGER.NAME}",
"message": "Trigger: {TRIGGER.NAME}\r\nTrigger status: {TRIGGER.STATUS}\r\nTrigger severity: {TRIGGER.
"mediatypeid": "0"
}
}
],
"update_operations": [
{
"operationid": "31",
"operationtype": "12",
"evaltype": "0",
"opmessage": {
"default_msg": "1",
"subject": "",
"message": "",
"mediatypeid": "0"
}
},
```

```
    {
    "operationid": "32",
    "operationtype": "0",
    "evaltype": "0",
    "opmessage": {
    "default_msg": "0",
    "subject": "Updated: {TRIGGER.NAME}",
    "message": "{USER.FULLNAME} updated problem at {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME} with the follow
    "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
}
```


Получение действий обнаружения

Вернуть все настроенные действия обнаружения вместе с условиями и операциями этих действий. Фильтр использует тип вычисления "и", поэтому свойство `formula` является пустым, а свойство`eval_formula` генерируется автоматически.

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
    "output": "extend",
    "selectOperations": "extend",
    "selectFilter": "extend",
    "filter": {
    "eventsource": 1
    }
    },
    "id": 1
}
```


Ответ:


```
{
    "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
}
```

См. также

- Фильтр действия
- Операция действия

Источник

CAction::get() в *ui/include/classes/api/services/CAction.php*.

**action.update**

Описание

`object action.update(  /  actions)`

Этот метод позволяет обновлять существующие действия.

**Note:**
Этот метод доступен только для пользователей, имеющих типы *Администратор* и *Супер-администратор*. Права вызывать этот метод могут быть отменены в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства действия, которые необходимо обновить.

Свойство `actionid` должно быть определено для каждого действия, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам действия, метод принимает следующие параметры.

| Параметр | Тип | Описание |
|---|---|---|
| filter | object | Объект фильтра действия для замены текущего фильтра. |
| operations | array | Операции действия для замены существующих операций. |
| recovery_operations | array | Операции восстановления действия для замены существующих операций восстановления.<br><br>Поведение параметра:<br>- *поддерживается*, если `eventsource` объекта Action object установлен в ”event created by a trigger”, ”internal event” или ”event created on service status update” |
| update_operations | array | Операции обновления действия для замены существующих операций обновления.<br><br>Поведение параметра:<br>- *поддерживается*, если `eventsource` объекта Action object установлен в ”event created by a trigger” или ”event created on service status update” |

Возвращаемые значения

`(object)` Возвращает объект, содержащий ID обновленных действий под свойством `actionids`.

Примеры

Деактивация действия

Деактивировать действие, то есть, выставить его статус в ”1”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "action.update",
    "params": {
        "actionid": "2",
        "status": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "2"
        ]
    },
    "id": 1
}
```

См. также

- Фильтр действия
- Операция действия

Источник

CAction::update() в *ui/include/classes/api/services/CAction.php*.

### Alert

Этот класс предназначен для работы с оповещениями.

Ссылки на объекты:

- Alert

Доступные методы:

- alert.get - получение предупреждений

#### Объект оповещение

Следующие объекты напрямую связаны с методом API `alert`.

Оповещение

**Note:**
Оповещения создаются сервером Zabbix и не могут быть изменены через API.

Объект alert содержит информацию о том, были ли успешно выполнены определенные операции действия. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| alertid | ID | ID оповещения. |
| actionid | ID | ID действия, которое сгенерировало оповещение. |
| alerttype | integer | Тип оповещения.<br><br>Возможные значения:<br>0 - сообщение;<br>1 - удаленная команда. |

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время, когда было сгенерировано оповещение. |
| error | string | Текст ошибки, если возникли проблемы при отправке сообщения или выполнении команды. |
| esc_step | integer | Шаг эскалации действия, во время которого было сгенерировано оповещение. |
| eventid | ID | ID события, которое запустило действие. |
| mediatypeid | ID | ID типа медиа, который был использован для отправки сообщения. |
| message | text | Текст сообщения.<br><br>Property behavior:<br>- *supported* if `alerttype` is set to ”message” |
| retries | integer | Количество попыток, которые Zabbix предпринял для отправки сообщения. |
| sendto | string | Адрес, имя пользователя или другой идентификатор получателя.<br><br>Property behavior:<br>- *supported* if `alerttype` is set to ”message” |
| status | integer | Статус, показывающий, было ли успешно выполнено операцию действия.<br><br>Возможные значения, если `alerttype` is set to ”message”:<br>0 - сообщение не отправлено;<br>1 - сообщение отправлено;<br>2 - сбой после нескольких повторных попыток;<br>3 - новое оповещение еще не обработано менеджером оповещений.<br><br>Возможные значения, если `alerttype` is set to ”remote command”:<br>0 - команда не выполнена;<br>1 - команда выполнена;<br>2 - была предпринята попытка выполнить команду на агенте Zabbix, но он был недоступен. |
| subject | string | Тема сообщения.<br><br>Property behavior:<br>- *supported* if `alerttype` is set to ”message” |
| userid | ID | ID пользователя, которому было отправлено сообщение. |
| p_eventid | ID | ID события проблемы, которое сгенерировало оповещение. |
| acknowledgeid | ID | ID подтверждения, которое сгенерировало оповещение. |

**alert.get**

Описание

`integer/array alert.get(    parameters)`

Метод позволяет получать оповещения по заданным параметрам.

> **Note:**
> Этот метод доступен для пользователей любого типа. Права доступа вызова метода можно отменить в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| alertids | ID/array | Возвращать только оповещения с указанными ID. |

| Parameter | Type | Description |
|---|---|---|
| actionids | ID/array | Возвращать только оповещения, сгенерированные указанными действиями. |
| eventids | ID/array | Возвращать только оповещения, сгенерированные указанными событиями. |
| groupids | ID/array | Возвращать только оповещения, сгенерированные объектами из указанных групп узлов сети. |
| hostids | ID/array | Возвращать только оповещения, сгенерированные объектами из указанных узлов сети. |
| mediatypeids | ID/array | Возвращать только оповещения сообщений, использовавшие указанные типы медиа. |
| objectids | ID/array | Возвращать только оповещения, сгенерированные указанными объектами |
| userids | ID/array | Возвращать только оповещения сообщений, отправленные указанным пользователям. |
| eventobject | integer | Возвращать только оповещения, сгенерированные событиями, связанными с объектами указанного типа.<br><br>См. event `object` для списка поддерживаемых типов объектов.<br><br>По умолчанию: 0 - триггер. |
| eventsource | integer | Возвращать только оповещения, сгенерированные событиями указанного типа.<br><br>См. event `source` для списка поддерживаемых типов событий.<br><br>По умолчанию: 0 - события триггера. |
| time_from | timestamp | Возвращать только оповещения, сгенерированные после указанного времени. |
| time_till | timestamp | Возвращать только оповещения, сгенерированные до указанного времени. |
| selectHosts | query | Возвращать свойство `hosts` с данными узлов сети, которые вызвали операцию действия. |
| selectMediatypes | query | Возвращать свойство `mediatypes` с массивом типов медиа, использованных для оповещения сообщения.<br><br>См. `mediatype.get` для ограничений, зависящих от типа пользователя. |
| selectUsers | query | Возвращать свойство `users` с массивом пользователей, которым было адресовано сообщение.<br><br>См. `user.get` для ограничений, зависящих от типа пользователя. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `alertid`, `clock`, `eventid`, `mediatypeid`, `sendto`, `status`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение предупреждений по ID действия

Получить все оповещения, созданные действием "3".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "alert.get",
    "params": {
        "output": "extend",
        "actionids": "3"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "alertid": "1",
            "actionid": "3",
            "eventid": "21243",
            "userid": "1",
            "clock": "1362128008",
            "mediatypeid": "1",
            "sendto": "support@company.com",
            "subject": "PROBLEM: Zabbix agent on Linux server is unreachable for 5 minutes: ",
            "message": "Trigger: Zabbix agent on Linux server is unreachable for 5 minutes: \nTrigger stat
            "status": "0",
            "retries": "3",
            "error": "",
            "esc_step": "1",
            "alerttype": "0",
            "p_eventid": "0",
            "acknowledgeid": "0"
        }
    ],
    "id": 1
}
```

См. также

- Узел сети
- Тип медиа
- Пользователь

Источник

CAlert::get() в *ui/include/classes/api/services/CAlert.php*.

#### API info

Этот класс предназначен для получения метаинформации об API.

Доступные методы:

- apiinfo.version - получение версии Zabbix API

#### apiinfo.version

Описание

`string apiinfo.version(array)`

Этот метод позволяет получить версию Zabbix API.

> **Attention:**
> Этот метод доступен только неаутентифицированным пользователям.

Параметры

`(array)` Метод принимает пустой массив.

Return values

`(string)` Возвращает версию Zabbix API.

> **Note:**
> Начиная с Zabbix 2.0.4 версия API соответствует версии Zabbix.

Примеры

Получение версии API

Получить версию Zabbix API.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "apiinfo.version",
    "params": [],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": "7.4.0",
    "id": 1
}
```

Источник

CAPIInfo::version() в *ui/include/classes/api/services/CAPIInfo.php*.

### Map

Этот класс предназначен для работы с картами.

Ссылки на объекты:

- Map
- Map element
  - Map element Host
  - Map element Host group
  - Map element Map
  - Map element Trigger
  - Map element Tag
  - Map element URL
- Map link
  - Map link trigger
  - Map link indicators
- Map URL
- Map user
- Map user group
- Map shapes
- Map lines

Доступные методы:

- map.create - создать новые карты
- map.delete - удалить карты
- map.get - получить карты
- map.update - обновить карты

**Объект карты**

Следующие объекты напрямую связаны с API map.

Карта

Объект карты имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| sysmapid | ID | ID карты.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| height | integer | Высота карты в пикселях.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Имя карты.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| width | integer | Ширина карты в пикселях.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| backgroundid | ID | ID изображения, используемого в качестве фона карты. |
| background_scale | integer | Включать ли пропорциональное масштабирование фонового изображения.<br><br>Возможные значения:<br>0 - отключить масштабирование;<br>1 - *(по умолчанию)* включить масштабирование изображения. |
| expand_macros | integer | Следует ли раскрывать макросы в подписях при настройке карты.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не раскрывать макросы;<br>1 - раскрывать макросы. |
| expandproblem | integer | Следует ли отображать триггер проблемы для элементов с одной проблемой.<br><br>Возможные значения:<br>0 - всегда отображать количество проблем;<br>1 - *(по умолчанию)* отображать триггер проблемы, если проблема только одна. |
| grid_align | integer | Включать ли выравнивание по сетке.<br><br>Возможные значения:<br>0 - отключить выравнивание по сетке;<br>1 - *(по умолчанию)* включить выравнивание по сетке. |
| grid_show | integer | Показывать ли сетку на карте.<br><br>Возможные значения:<br>0 - не показывать сетку;<br>1 - *(по умолчанию)* показывать сетку. |

| Property | Type | Description |
|---|---|---|
| grid_size | integer | Размер сетки карты в пикселях.<br><br>Поддерживаемые значения: 20, 40, 50, 75 и 100.<br><br>По умолчанию: 50. |
| highlight | integer | Включена ли подсветка значков.<br><br>Возможные значения:<br>0 - подсветка отключена;<br>1 - *(по умолчанию)* подсветка включена. |
| iconmapid | ID | ID карты значков, используемой на карте. |
| label_format | integer | Включать ли расширенные подписи.<br><br>Возможные значения:<br>0 - *(по умолчанию)* отключить расширенные подписи;<br>1 - включить расширенные подписи. |
| label_location | integer | Расположение подписи элемента карты.<br><br>Возможные значения:<br>0 - *(по умолчанию)* снизу;<br>1 - слева;<br>2 - справа;<br>3 - сверху. |
| label_string_host | string | Пользовательская подпись для элементов узла сети.<br><br>Поведение свойства:<br>- *обязательно*, если `label_type_host` установлено в ”custom” |
| label_string_hostgroup | string | Пользовательская подпись для элементов группы узлов сети.<br><br>Поведение свойства:<br>- *обязательно*, если `label_type_hostgroup` установлено в ”custom” |
| label_string_image | string | Пользовательская подпись для элементов изображения.<br><br>Поведение свойства:<br>- *обязательно*, если `label_type_image` установлено в ”custom” |
| label_string_map | string | Пользовательская подпись для элементов карты.<br><br>Поведение свойства:<br>- *обязательно*, если `label_type_map` установлено в ”custom” |
| label_string_trigger | string | Пользовательская подпись для элементов триггера.<br><br>Поведение свойства:<br>- *обязательно*, если `label_type_trigger` установлено в ”custom” |
| label_type | integer | Тип подписи элемента карты.<br><br>Возможные значения:<br>0 - подпись;<br>1 - IP-адрес;<br>2 - *(по умолчанию)* имя элемента;<br>3 - только статус;<br>4 - ничего. |
| label_type_host | integer | Тип подписи для элементов узла сети.<br><br>Возможные значения:<br>0 - подпись;<br>1 - IP-адрес;<br>2 - *(по умолчанию)* имя элемента;<br>3 - только статус;<br>4 - ничего;<br>5 - пользовательский. |

| Property | Type | Description |
|---|---|---|
| label_type_hostgroup | integer | Тип подписи для элементов группы узлов сети.<br><br>Возможные значения:<br>0 - подпись;<br>2 - *(по умолчанию)* имя элемента;<br>3 - только статус;<br>4 - ничего;<br>5 - пользовательский. |
| label_type_image | integer | Тип подписи для элементов группы узлов сети.<br><br>Возможные значения:<br>0 - подпись;<br>2 - *(по умолчанию)* имя элемента;<br>4 - ничего;<br>5 - пользовательский. |
| label_type_map | integer | Тип подписи для элементов карты.<br><br>Возможные значения:<br>0 - подпись;<br>2 - *(по умолчанию)* имя элемента;<br>3 - только статус;<br>4 - ничего;<br>5 - пользовательский. |
| label_type_trigger | integer | Тип подписи для элементов триггера.<br><br>Возможные значения:<br>0 - подпись;<br>2 - *(по умолчанию)* имя элемента;<br>3 - только статус;<br>4 - ничего;<br>5 - пользовательский. |
| markelements | integer | Следует ли выделять элементы карты, которые недавно изменили свой статус.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не выделять элементы;<br>1 - выделять элементы. |
| severity_min | integer | Минимальная серьезность триггеров, которые будут отображаться на карте.<br><br>См. свойство `severity` триггера для списка поддерживаемых уровней серьезности триггеров. |
| show_element_label | integer | Как по умолчанию отображать подписи элементов.<br><br>Возможные значения:<br>0 - всегда отображать;<br>1 - *(по умолчанию)* автоматически скрывать. |
| show_link_label | integer | Как по умолчанию отображать подписи связей.<br><br>Возможные значения:<br>0 - всегда отображать;<br>1 - *(по умолчанию)* автоматически скрывать. |
| show_unack | integer | Как следует отображать проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* отображать количество всех проблем;<br>1 - отображать только количество неподтвержденных проблем;<br>2 - отображать количество подтвержденных и неподтвержденных проблем отдельно. |
| userid | ID | ID пользователя, являющегося владельцем карты. |

| Property | Type | Description |
|---|---|---|
| private | integer | Тип общего доступа к карте.<br><br>Возможные значения:<br>0 - общедоступная карта;<br>1 - *(по умолчанию)* частная карта. |
| show_suppressed | integer | Показывать ли подавленные проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* скрывать подавленные проблемы;<br>1 - показывать подавленные проблемы. |

Элемент карты

Объект элемента карты определяет объект, отображаемый на карте. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| selementid | ID | ID элемента карты.<br><br>Поведение свойства:<br>- *только для чтения* |
| elements | array | Объект данных элемента.<br><br>Поведение свойства:<br>- *обязательно*, если `elementtype` установлен в ”host”, ”map”, ”trigger” или ”host group” |
| elementtype | integer | Тип элемента карты.<br><br>Возможные значения:<br>0 - host;<br>1 - map;<br>2 - trigger;<br>3 - host group;<br>4 - image.<br><br>Поведение свойства:<br>- *обязательно* |
| iconid_off | ID | ID изображения, используемого для отображения элемента в состоянии по умолчанию.<br><br>Поведение свойства:<br>- *обязательно* |
| areatype | integer | Как должны отображаться узлы сети группы узлов сети, расположенные отдельно.<br><br>Возможные значения:<br>0 - *(по умолчанию)* элемент группы узлов сети занимает всю карту;<br>1 - элемент группы узлов сети имеет фиксированный размер. |
| elementsubtype | integer | Как элемент группы узлов сети должен отображаться на карте.<br><br>Возможные значения:<br>0 - *(по умолчанию)* отображать группу узлов сети как один элемент;<br>1 - отображать каждый узел сети в группе отдельно. |
| evaltype | integer | Метод оценки условия фильтрации тегов элемента карты.<br><br>Возможные значения:<br>0 - *(по умолчанию)* And/Or;<br>2 - Or. |

| Property | Type | Description |
|---|---|---|
| height | integer | Высота элемента группы узлов сети фиксированного размера в пикселях.<br><br>По умолчанию: 200. |
| iconid_disabled | ID | ID изображения, используемого для отображения отключенных элементов карты.<br><br>Поведение свойства:<br>- *поддерживается*, если `elementtype` установлен в ”host”, ”map”, ”trigger” или ”host group” |
| iconid_maintenance | ID | ID изображения, используемого для отображения элементов карты в обслуживании.<br><br>Поведение свойства:<br>- *поддерживается*, если `elementtype` установлен в ”host”, ”map”, ”trigger” или ”host group” |
| iconid_on | ID | ID изображения, используемого для отображения элементов карты с проблемами.<br><br>Поведение свойства:<br>- *поддерживается*, если `elementtype` установлен в ”host”, ”map”, ”trigger” или ”host group” |
| label | string | Метка элемента. |
| label_location | integer | Расположение метки элемента карты.<br><br>Возможные значения:<br>-1 - *(по умолчанию)* расположение по умолчанию;<br>0 - снизу;<br>1 - слева;<br>2 - справа;<br>3 - сверху. |
| permission | integer | Тип уровня доступа.<br><br>Возможные значения:<br>-1 - нет;<br>2 - только чтение;<br>3 - чтение-запись. |
| show_label | integer | Как отображать метку элемента.<br><br>Возможные значения:<br>-1 - *(по умолчанию)* значение карты по умолчанию;<br>0 - всегда отображать;<br>1 - автоматически скрывать. |
| sysmapid | ID | ID карты, к которой принадлежит элемент.<br><br>Поведение свойства:<br>- *только для чтения* |
| urls | array | URL-адреса элемента карты.<br><br>Объект URL-адреса элемента карты подробно описан ниже. |
| use_iconmap | integer | Нужно ли использовать сопоставление значков для элементов host.<br><br>Возможные значения:<br>0 - не использовать сопоставление значков;<br>1 - *(по умолчанию)* использовать сопоставление значков. |
| viewtype | integer | Алгоритм размещения элемента группы узлов сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* сетка. |

| Property | Type | Description |
|---|---|---|
| width | integer | Ширина элемента группы узлов сети фиксированного размера в пикселях.<br><br>По умолчанию: 200. |
| x | integer | Координата X элемента в пикселях.<br><br>По умолчанию: 0. |
| y | integer | Координата Y элемента в пикселях.<br><br>По умолчанию: 0. |
| zindex | integer | Значение, используемое для упорядочивания элементов карты (z-index).<br><br>По умолчанию: 0. |

Элемент карты узел сети Host

Объект узла сети Host элемента карты определяет один элемент узла сети.

| Property | Type | Description |
|---|---|---|
| hostid | ID | ID узла сети. |

Элемент карты Группа узлов сети

Объект элемента карты Группа узлов сети определяет один элемент группы узлов сети.

| Property | Type | Description |
|---|---|---|
| groupid | ID | ID группы узлов сети. |

Элемент карты Map

Объект Map элемента карты определяет один элемент карты.

| Property | Type | Description |
|---|---|---|
| sysmapid | ID | ID карты. |

Объект триггера элемента карты

Объект Trigger элемента карты определяет один или несколько элементов триггера.

| Property | Type | Description |
|---|---|---|
| triggerid | ID | ID триггера. |

Тег элемента карты

Объект Tag элемента карты имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега элемента карты.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| operator | integer | Оператор условия тега элемента карты.<br><br>Возможные значения:<br>0 - *(default)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| value | string | Значение тега элемента карты. |

URL элемента карты

Объект URL элемента карты определяет кликабельную ссылку, которая будет доступна для конкретного элемента карты. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| sysmapelementurlid | ID | ID URL элемента карты.<br><br>Property behavior:<br>- *read-only* |
| name | string | Подпись ссылки.<br><br>Property behavior:<br>- *required* |
| url | string | URL ссылки.<br><br>Property behavior:<br>- *required* |
| selementid | ID | ID элемента карты, которому принадлежит URL. |

Ссылка карты

Объект ссылки карты определяет связь между двумя элементами карты. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| linkid | ID | ID ссылки карты.<br><br>Property behavior:<br>- *read-only* |
| sysmapid | ID | ID карты, к которой принадлежит ссылка. |
| selementid1 | ID | ID первого элемента карты, связанного на одном конце.<br><br>Property behavior:<br>- *required* |
| selementid2 | ID | ID первого элемента карты, связанного на другом конце.<br><br>Property behavior:<br>- *required* |
| drawtype | integer | Стиль отображения линии ссылки.<br><br>Possible values:<br>0 - *(default)* line;<br>2 - bold line;<br>3 - dotted line;<br>4 - dashed line. |
| color | string | Цвет линии в виде шестнадцатеричного кода цвета.<br><br>Default: 000000. |

| Property | Type | Description |
|---|---|---|
| label | string | Метка ссылки. |
| show_label | integer | Как отображать метку ссылки.<br><br>Possible values:<br>-1 - *(default)* map default;<br>0 - always display;<br>1 - auto-hide. |
| indicator_type | integer | Выберите тип индикатора ссылки.<br><br>Possible values:<br>0 - *(default)* static link;<br>1 - trigger;<br>2 - item value. |
| linktriggers | array | Триггеры ссылки карты, используемые в качестве индикаторов состояния ссылки.<br><br>Объект триггера ссылки карты подробно описан ниже.<br><br>Property behavior:<br>- *required* if `indicator_type` is set to ”trigger” (1). |
| itemid | ID | ID элемента данных.<br><br>Property behavior:<br>- *required* if `indicator_type` is set to ”item value” (2). |
| highlights | array | Подсветки ссылки карты, используемые в качестве индикаторов состояния ссылки.<br><br>Объект индикаторов ссылки карты подробно описан ниже.<br><br>Property behavior:<br>- *required* if `indicator_type` is set to ”item value” (2). |
| thresholds | array | Пороги ссылки карты, используемые в качестве индикаторов состояния ссылки.<br><br>Объект индикаторов ссылки карты подробно описан ниже.<br><br>Property behavior:<br>- *required* if `indicator_type` is set to ”item value” (2). |
| permission | integer | Тип уровня доступа.<br><br>Possible values:<br>-1 - none;<br>2 - read only;<br>3 - read-write. |

Триггер ссылки карты

Объект триггера ссылки карты определяет индикатор состояния ссылки карты на основе состояния триггера. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| triggerid | ID | ID триггера, используемый в качестве индикатора ссылки.<br><br>Property behavior:<br>- *required* |
| color | string | Цвет индикатора в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: `DD0000`. |

| Property | Type | Description |
|---|---|---|
| drawtype | integer | Стиль отображения индикатора.<br><br>Возможные значения:<br>0 - *(по умолчанию)* линия;<br>2 - жирная линия;<br>3 - пунктирная линия;<br>4 - штриховая линия. |

Индикаторы ссылок карты

Объект индикаторов ссылок карты определяет индикатор состояния ссылки карты на основе значения элемента данных. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| pattern | string | Регулярное выражение для сравнения. Доступно только для подсветок. |
| sortorder | integer | Используется для задания порядка сортировки подсветок. Доступно только для подсветок. |
| threshold | string | Числовое значение для сравнения. Доступно только для порогов. |
| drawtype | integer | Стиль отображения индикатора.<br><br>Possible values:<br>0 - *(default)* line;<br>2 - bold line;<br>3 - dotted line;<br>4 - dashed line. |
| color | string | Цвет индикатора в виде шестнадцатеричного кода цвета.<br><br>Default: `DD0000`.<br><br>Property behavior:<br>- *required* |

URL карты

Объект URL карты определяет кликабельную ссылку, которая будет доступна для всех элементов определенного типа на карте. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| sysmapurlid | ID | ID URL карты.<br><br>Property behavior:<br>- *read-only* |
| name | string | Подпись ссылки.<br><br>Property behavior:<br>- *required* |
| url | string | URL ссылки.<br><br>Property behavior:<br>- *required* |
| elementtype | integer | Тип элемента карты, для которого будет доступен URL.<br><br>Список поддерживаемых типов см. в свойстве `type` элемента карты.<br><br>По умолчанию: 0. |
| sysmapid | ID | ID карты, к которой относится URL. |

Пользователь карты

Список прав доступа к карте на основе пользователей. Имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| sysmapuserid | ID | ID пользователя карты.<br><br>Property behavior:<br>- *read-only* |
| userid | ID | ID пользователя.<br><br>Property behavior:<br>- *required* |
| permission | integer | Тип уровня доступа.<br><br>Возможные значения:<br>2 - только чтение;<br>3 - чтение и запись.<br><br>Property behavior:<br>- *required* |

Группа пользователей карты

Список прав доступа к карте на основе групп пользователей. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| sysmapusrgrpid | ID | ID группы пользователей карты.<br><br>Property behavior:<br>- *read-only* |
| usrgrpid | ID | ID группы пользователей.<br><br>Property behavior:<br>- *required* |
| permission | integer | Тип уровня доступа.<br><br>Possible values:<br>2 - только чтение;<br>3 - чтение и запись.<br><br>Property behavior:<br>- *required* |

Фигуры карты

Объект фигуры карты определяет геометрическую фигуру (с текстом или без него), отображаемую на карте. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| sysmap_shapeid | ID | ID элемента фигуры карты.<br><br>Поведение свойства:<br>- *только для чтения* |

| Property | Type | Description |
|---|---|---|
| type | integer | Тип элемента фигуры карты.<br><br>Возможные значения:<br>0 - прямоугольник;<br>1 - эллипс.<br><br>Свойство обязательно при создании новых фигур.<br><br>Поведение свойства:<br>- *обязательное* |
| x | integer | X-координата фигуры в пикселях.<br><br>По умолчанию: 0. |
| y | integer | Y-координата фигуры в пикселях.<br><br>По умолчанию: 0. |
| width | integer | Ширина фигуры в пикселях.<br><br>По умолчанию: 200. |
| height | integer | Высота фигуры в пикселях.<br><br>По умолчанию: 200. |
| text | string | Текст фигуры. |
| font | integer | Шрифт текста внутри фигуры.<br><br>Возможные значения:<br>0 - Georgia, serif<br>1 - ”Palatino Linotype”, ”Book Antiqua”, Palatino, serif<br>2 - ”Times New Roman”, Times, serif<br>3 - Arial, Helvetica, sans-serif<br>4 - ”Arial Black”, Gadget, sans-serif<br>5 - ”Comic Sans MS”, cursive, sans-serif<br>6 - Impact, Charcoal, sans-serif<br>7 - ”Lucida Sans Unicode”, ”Lucida Grande”, sans-serif<br>8 - Tahoma, Geneva, sans-serif<br>9 - ”Trebuchet MS”, Helvetica, sans-serif<br>10 - Verdana, Geneva, sans-serif<br>11 - ”Courier New”, Courier, monospace<br>12 - ”Lucida Console”, Monaco, monospace<br><br>По умолчанию: 9. |
| font_size | integer | Размер шрифта в пикселях.<br><br>По умолчанию: 11. |
| font_color | string | Цвет шрифта.<br><br>По умолчанию: 000000. |
| text_halign | integer | Горизонтальное выравнивание текста.<br><br>Возможные значения:<br>0 - по центру;<br>1 - слева;<br>2 - справа.<br><br>По умолчанию: 0. |

| Property | Type | Description |
|---|---|---|
| text_valign | integer | Вертикальное выравнивание текста.<br><br>Возможные значения:<br>0 - по середине;<br>1 - сверху;<br>2 - снизу.<br><br>По умолчанию: 0. |
| border_type | integer | Тип границы.<br><br>Возможные значения:<br>0 - нет;<br>1 - ──────;<br>2 - ┅;<br>3 - ╌ ╌ ╌.<br><br>По умолчанию: 0. |
| border_width | integer | Ширина границы в пикселях.<br><br>По умолчанию: 0. |
| border_color | string | Цвет границы.<br><br>По умолчанию: 000000. |
| background_color | string | Цвет фона (цвет заливки).<br><br>По умолчанию: (empty). |
| zindex | integer | Значение, используемое для упорядочивания всех фигур и линий (z-index).<br><br>По умолчанию: 0. |

Линии карты

Объект линии карты определяет линию, отображаемую на карте. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| sysmap_shapeid | ID | ID элемента фигуры карты.<br><br>Property behavior:<br>- *read-only* |
| x1 | integer | Координата X точки 1 линии в пикселях.<br><br>По умолчанию: 0. |
| y1 | integer | Координата Y точки 1 линии в пикселях.<br><br>По умолчанию: 0. |
| x2 | integer | Координата X точки 2 линии в пикселях.<br><br>По умолчанию: 200. |
| y2 | integer | Координата Y точки 2 линии в пикселях.<br><br>По умолчанию: 200. |

| Property | Type | Description |
|---|---|---|
| line_type | integer | Тип линий.<br><br>Возможные значения:<br>0 - нет;<br>1 - ————;<br>2 - ┅;<br>3 - – – –.<br><br>По умолчанию: 0. |
| line_width | integer | Ширина линий в пикселях.<br><br>По умолчанию: 0. |
| line_color | string | Цвет линии.<br><br>По умолчанию: 000000. |
| zindex | integer | Значение, используемое для упорядочивания всех фигур и линий (z-index).<br><br>По умолчанию: 0. |

**map.create**

Описание

`object map.create(object/array maps)`

Этот метод позволяет создавать новые карты.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Карты для создания.

В дополнение к стандартным свойствам карты, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| links | array | Связи карты для создания на карте. |
| selements | array | Элементы карты для создания на карте. |
| urls | array | URL карты для создания на карте. |
| users | array | Общий доступ карты пользователем для создания на карте. |
| userGroups | array | Общий доступ карты группой пользователей для создания на карте. |
| shapes | array | Фигуры карты для создания на карте. |
| lines | array | Линии карты для создания на карте. |

**Note:**
Чтобы создать связи карты, нужно задать для элемента карты `selementid` произвольное значение, а затем использовать это значение для ссылки на этот элемент в свойствах links `selementid1` или `selementid2`. Когда элемент будет создан, это значение будет заменено на корректный ID, сгенерированный Zabbix. См. пример.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных карт в свойстве `sysmapids`. Порядок возвращаемых идентификаторов соответствует порядку переданных карт.

Примеры

Создание пустой карты

Создайте карту без элементов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Map",
        "width": 600,
        "height": 600
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Создание карты узлов сети

Создайте карту с двумя элементами узлов сети и связью между ними. Обратите внимание на использование временных значений "selementid1" и "selementid2" в объекте связи карты для ссылки на элементы карты.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Создание карты триггеров

Создайте карту с элементом триггера, который содержит два триггера.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Trigger map",
        "width": 600,
        "height": 600,
        "selements": [
            {
                "elements": [
                    {"triggerid": "12345"},
                    {"triggerid": "67890"}
                ],
                "elementtype": 2,
                "iconid_off": "2"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "10"
        ]
    },
    "id": 1
}
```

Совместное использование карты

Создайте карту с двумя типами совместного доступа (пользователь и группа пользователей).

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Фигуры карты

Создайте карту с заголовком имени карты.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "10"
        ]
    },
    "id": 1
}
```

Линии карты

Создайте линию карты.

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "11"
        ]
    },
    "id": 1
}
```

См. также

- Элемент карты
- Ссылка карты
- URL карты
- Пользователь карты
- Группа пользователей карты
- Фигура карты
- Линия карты

Источник

CMap::create() в *ui/include/classes/api/services/CMap.php*.

### map.delete

Описание

`object map.delete(    mapIds)`

Этот метод позволяет удалять карты.

> **Note:**
> Этот метод доступен пользователям любого типа.  Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(array) Идентификаторы карт, которые нужно удалить.

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы удаленных карт под свойством sysmapids.

Примеры

Удаление нескольких карт

Удаление двух карт.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "map.delete",
    "params": [
        "12",
        "34"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "12",
            "34"
        ]
    },
    "id": 1
}
```

Источник

CMap::delete() в *ui/include/classes/api/services/CMap.php*.

**map.get**

Описание

integer/array map.get( )

Метод позволяет получать карты по заданным параметрам.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object) Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| sysmapids | ID/array | Возвращает только карты с указанными ID. |
| userids | ID/array | Возвращает только карты, принадлежащие указанным ID пользователей. |
| expandUrls | flag | Добавляет глобальные URL карты к соответствующим элементам карты и раскрывает макросы во всех URL элементов карты. |
| selectIconMap | query | Возвращает свойство `iconmap` с icon map, используемой на карте. |
| selectLinks | query | Возвращает свойство `links` со связями карты между элементами. |
| selectSelements | query | Возвращает свойство `selements` с элементами карты. |
| selectUrls | query | Возвращает свойство `urls` с URL карты. |

| Parameter | Type | Description |
|---|---|---|
| selectUsers | query | Возвращает свойство `users` с пользователями, с которыми карта расшарена. |
| selectUserGroups | query | Возвращает свойство `userGroups` с группами пользователей, с которыми карта расшарена. |
| selectShapes | query | Возвращает свойство `shapes` с фигурами карты. |
| selectLines | query | Возвращает свойство `lines` с линиями карты. |
| sortfield | string/array | Сортирует результат по указанным свойствам.<br><br>Возможные значения: `name`, `width`, `height`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение карты

Получить все данные о карте "3".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "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
}
```


См. также

- Icon map
- Элемент карты
- Ссылка карты
- URL карты
- Пользователь карты
- Группа пользователей карты
- Фигуры карты
- Линии карты

Источник

CMap::get() в *ui/include/classes/api/services/CMap.php*.

**map.update**

Описание

`object map.update(  /  maps)`

Этот метод позволяет обновлять существующие карты.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Сопоставление свойств для обновления.

Свойство `mapid` должно быть определено для каждой карты, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам карты, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| links | array | Ссылки карты для замены существующих ссылок. |
| selements | array | Элементы карты для замены существующих элементов. |
| urls | array | URL карты для замены существующих URL. |
| users | array | Пользователь карты для замены существующих элементов. |
| userGroups | array | Группа пользователей карты для замены существующих элементов. |
| shapes | array | Фигуры карты для замены существующих фигур. |
| lines | array | Линии карты для замены существующих линий. |

**Note:**
Чтобы создать ссылки карты между новыми элементами карты, вам нужно задать для `selementid` элемента произвольное значение, а затем использовать это значение для ссылки на этот элемент в свойствах `selementid1` или `selementid2` ссылок. Когда элемент будет создан, это значение будет заменено на правильный ID, сгенерированный Zabbix. См. пример для map.create.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных карт под свойством `sysmapids`.

Примеры

Изменение размера карты

Измените размер карты на 1200x1200 пикселей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
```

```
        "sysmapid": "8",
        "width": 1200,
        "height": 1200
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Изменение владельца карты

Доступно только для администраторов и супер-администраторов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "9",
        "userid": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

См. также

- Элемент карты
- Ссылка карты
- URL карты
- Пользователь карты
- Группа пользователей карты
- Фигуры карты
- Линии карты

Источник

CMap::update() в *ui/include/classes/api/services/CMap.php*.

**MFA**

Этот класс предназначен для работы с методами MFA (многофакторной аутентификации).

Ссылки на объекты:

- MFA

Доступные методы:

- mfa.create - создать новые методы MFA
- mfa.delete - удалить методы MFA
- mfa.get - получить методы MFA
- mfa.update - обновить методы MFA

**Объект MFA**

Следующие объекты напрямую связаны с API `mfa`.

MFA

Объект MFA (Multi-Factor Authentication) имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| mfaid | ID | ID метода MFA.<br><br>Property behavior:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| type | integer | Тип метода MFA.<br><br>Возможные значения:<br>1 - TOTP (одноразовые пароли на основе времени);<br>2 - Duo Universal Prompt. |
| name | string | Уникальное имя метода MFA.<br><br>Property behavior:<br>- *обязательно* для операций создания |
| hash_function | integer | Тип хэш-функции для генерации кодов TOTP.<br><br>Возможные значения:<br>1 - SHA-1;<br>2 - SHA-256;<br>3 - SHA-512.<br><br>Property behavior:<br>- *обязательно*, если `type` установлено в ”TOTP” |
| code_length | integer | Длина кода подтверждения.<br><br>Возможные значения:<br>6 - 6 цифр;<br>8 - 8 цифр.<br><br>Property behavior:<br>- *обязательно*, если `type` установлено в ”TOTP” |
| api_hostname | string | Имя хоста API, предоставленное службой аутентификации Duo.<br><br>Property behavior:<br>- *обязательно*, если `type` установлено в ”Duo Universal Prompt” |
| clientid | string | ID клиента, предоставленный службой аутентификации Duo.<br><br>Property behavior:<br>- *обязательно*, если `type` установлено в ”Duo Universal Prompt” |
| client_secret | string | Секрет клиента, предоставленный службой аутентификации Duo.<br><br>Property behavior:<br>- *только для записи*<br>- *обязательно*, если `type` установлено в ”Duo Universal Prompt” |

**mfa.create**

Описание

`object mfa.create(  /  MFA methods)`

Этот метод позволяет создавать новые методы MFA.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Методы MFA для создания.

Метод принимает методы MFA со стандартными свойствами метода MFA.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных методов MFA в свойстве `mfaids`. Порядок возвращаемых идентификаторов соответствует порядку переданных элементов.

Примеры

Создание методов MFA

Создайте метод MFA "Zabbix TOTP", использующий одноразовые пароли на основе времени (TOTP), с функцией хеширования для генерации кодов TOTP, установленной в SHA-1, и длиной кода проверки, установленной в 6 цифр.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.create",
    "params": {
        "type": 1,
        "name": "Zabbix TOTP",
        "hash_function": 1,
        "code_length": 6
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Исходный код

CMfa::create() в *ui/include/classes/api/services/CMfa.php*.

**mfa.delete**

Описание

`object mfa.delete(  mfaids)`

Этот метод позволяет удалять методы MFA.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы методов MFA, которые нужно удалить.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных методов MFA в свойстве `mfaids`.

Примеры

Удаление методов MFA

Удаление метода MFA.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.delete",
    "params": [
        "2"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "2"
        ]
    },
    "id": 1
}
```

Исходный код

CMfa::delete() в *ui/include/classes/api/services/CMfa.php*.

**mfa.get**

Описание

`integer/array mfa.get(          )`

Метод позволяет получать методы MFA по заданным параметрам.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| mfaids | ID/array | Возвращать только методы MFA с указанными ID. |

| Parameter | Type | Description |
|---|---|---|
| selectUsrgrps | query | Возвращать свойство `usrgrps` с группами пользователей, связанными с методами MFA.<br><br>Поддерживает `count`. |
| filter | object | Возвращать только те результаты, которые в точности соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений для сопоставления.<br><br>Поддерживаемые свойства:<br>`mfaid` - ID метода MFA;<br>`type` - тип метода MFA. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `name`. |
| search | object | Возвращать результаты, соответствующие заданному шаблону (без учета регистра).<br><br>Принимает объект, где ключи — это имена свойств, а значения — строки для поиска. Если дополнительные параметры не указаны, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживаемые свойства: `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Поиск методов MFA по имени

Получить все методы MFA, в названии которых есть ”Zabbix”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.get",
    "params": {
        "output": "extend",
        "search": {
            "name": "Zabbix"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Исходный код

CMfa::get() в *ui/include/classes/api/services/CMfa.php*.

**mfa.update**

Описание

`object mfa.update(  /  MFA methods)`

Этот метод позволяет обновлять существующие методы MFA.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства метода MFA, которые необходимо обновить.

Для каждого элемента должно быть определено свойство `mfaid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Метод принимает методы MFA со стандартными свойствами метода MFA.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных методов MFA в свойстве `mfaids`.

Примеры

Обновление свойств метода

Обновите функцию хеширования для генерации кодов TOTP и длину кода проверки для метода MFA "Zabbix TOTP", использующего одноразовые пароли на основе времени (TOTP).

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.update",
    "params": {
        "mfaid": "1",
        "hash_function": 3,
```

```
        "code_length": 8
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Исходный код

CMfa::update() в *ui/include/classes/api/services/CMfa.php*.

### SLA

Этот класс предназначен для работы с объектами SLA (Service Level Agreement), используемыми для оценки производительности ИТ-инфраструктуры и бизнес-сервисов.

Ссылки на объекты:

- SLA
- SLA schedule
- SLA excluded downtime
- SLA service tag

Доступные методы:

- sla.create - создать новые SLA
- sla.delete - удалить SLA
- sla.get - получить SLA
- sla.getsli - получить данные Service Level Indicator (SLI) для SLA
- sla.update - обновить SLA

### Объект SLA

Следующие объекты напрямую связаны с API `sla` (Service Level Agreement).

SLA

Объект SLA имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| slaid | ID | ID SLA.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя SLA.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| period | integer | Период отчетности SLA.<br><br>Возможные значения:<br>0 - ежедневно;<br>1 - еженедельно;<br>2 - ежемесячно;<br>3 - ежеквартально;<br>4 - ежегодно.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| slo | float | Минимально допустимая цель уровня обслуживания, выраженная в процентах. Если показатель уровня обслуживания (SLI) опускается ниже, SLA считается находящимся в состоянии проблемы/невыполненным.<br><br>Возможные значения: 0-100 (до 4 знаков после запятой).<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| effective_date | integer | Дата вступления SLA в силу.<br><br>Возможные значения: временная метка даты в UTC. |
| timezone | string | Часовой пояс отчетности, например: `Europe/London`, `UTC`.<br><br>Полный список поддерживаемых часовых поясов см. в документации PHP.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| status | integer | Статус SLA.<br><br>Возможные значения:<br>0 - *(по умолчанию)* отключенный SLA;<br>1 - включенный SLA. |
| description | string | Описание SLA. |

Расписание SLA

Объект расписания SLA определяет периоды, в течение которых подключенная служба или службы должны находиться в рабочем состоянии. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| period_from | integer | Время начала повторяющегося недельного периода времени (включительно).<br><br>Возможные значения: количество секунд (отсчет от воскресенья).<br><br>Поведение свойства:<br>- *required* |
| period_to | integer | Время окончания повторяющегося недельного периода времени (не включая указанное значение).<br><br>Возможные значения: количество секунд (отсчет от воскресенья).<br><br>Поведение свойства:<br>- *required* |

Исключенное время простоя SLA

Объект исключенного времени простоя определяет периоды, в которые подключенная служба или службы запланированы как недоступные для работы, без влияния на SLI, например, во время планового обслуживания. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя исключенного периода простоя.<br><br>Поведение свойства:<br>- *required* |
| period_from | integer | Время начала исключенного периода простоя (включительно).<br><br>Допустимые значения: timestamp.<br><br>Поведение свойства:<br>- *required* |
| period_to | integer | Время окончания исключенного периода простоя (не включая).<br><br>Допустимые значения: timestamp.<br><br>Поведение свойства:<br>- *required* |

Тег сервиса SLA

Объект тега сервиса SLA связывает сервисы, которые нужно включить в расчеты SLA. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега сервиса SLA.<br><br>Поведение свойства:<br>- *required* |
| operator | integer | Оператор тега сервиса SLA.<br><br>Возможные значения:<br>0 - *(default)* Равно;<br>2 - Содержит. |
| value | string | Значение тега сервиса SLA. |

**sla.create**

Описание

`object sla.create( / SLA)`

Этот метод позволяет создавать новые объекты SLA.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Объекты SLA для создания.

В дополнение к стандартным свойствам SLA, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| service_tags | array | Теги сервиса SLA, которые будут созданы для SLA.<br><br>Поведение параметра:<br>- *required* |
| schedule | array | Расписание SLA, которое будет создано для SLA.<br>Указание пустого параметра будет интерпретировано как расписание 24x7.<br>По умолчанию: расписание 24x7. |
| excluded_downtimes | array | Исключенные периоды простоя SLA, которые будут созданы для SLA. |

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы созданных SLA в свойстве slaids. Порядок возвращаемых идентификаторов соответствует порядку переданных SLA.

Примеры

Создание SLA

Укажите создание записи SLA для: * отслеживания времени безотказной работы служб, связанных с SQL-движком; * пользовательского расписания на все будние дни, за исключением последнего часа в субботу; * даты вступления в силу — последний день 2022 года; * запланированного простоя длительностью 1 час 15 минут, начиная с полуночи 4 июля; * включенного еженедельного расчета отчета SLA; * минимально допустимого SLO на уровне 99.9995%.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "sla.create",
    "params": [
        {
            "name": "Database Uptime",
            "slo": "99.9995",
            "period": "1",
            "timezone": "America/Toronto",
            "description": "Provide excellent uptime for main database engines.",
            "effective_date": 1672444800,
            "status": 1,
            "schedule": [
                {
                    "period_from": 0,
                    "period_to": 601200
                }
            ],
            "service_tags": [
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "mysql"
                },
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "postgresql"
                }
            ],
            "excluded_downtimes": [
                {
                    "name": "Software version upgrade rollout",
                    "period_from": "1648760400",
                    "period_to": "1648764900"
                }
            ]
```

```
        }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CSla::create() в *ui/include/classes/api/services/CSla.php*.

**sla.delete**

Описание

`object sla.delete(    slaids)`

Этот метод позволяет удалять записи SLA.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы SLA, которые нужно удалить.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных SLA в свойстве `slaids`.

Примеры

Удаление нескольких SLA

Удалите две записи SLA.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "sla.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "4",
            "5"
        ]
```

```
    },
    "id": 1
}
```

Источник

CSla::delete() в *ui/include/classes/api/services/CSla.php*.

### sla.get

Описание

`integer/array sla.get(          )`

Метод позволяет получать объекты SLA по заданным параметрам.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| slaids | ID/array | Возвращать только SLA с указанными ID. |
| serviceids | ID/array | Возвращать только SLA, соответствующие указанным сервисам. |
| selectSchedule | query | Возвращать свойство `schedule` с расписаниями SLA.<br><br>Поддерживает `count`. |
| selectExcludedDowntimes | query | Возвращать свойство `excluded_downtimes` с исключенными периодами простоя SLA.<br><br>Поддерживает `count`. |
| selectServiceTags | query | Возвращать свойство `service_tags` с тегами сервисов SLA.<br><br>Поддерживает `count`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `slaid`, `name`, `period`, `slo`, `effective_date`, `timezone`, `status`, `description`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение всех SLA

Получите все данные обо всех SLA и их свойствах.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "sla.get",
    "params": {
        "output": "extend",
        "selectSchedule": ["period_from", "period_to"],
        "selectExcludedDowntimes": ["name", "period_from", "period_to"],
        "selectServiceTags": ["tag", "operator", "value"],
        "preservekeys": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "1": {
            "slaid": "1",
            "name": "Database Uptime",
            "period": "1",
            "slo": "99.9995",
            "effective_date": "1672444800",
            "timezone": "America/Toronto",
            "status": "1",
            "description": "Provide excellent uptime for main SQL database engines.",
            "service_tags": [
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "mysql"
                },
                {
                    "tag": "database",
                    "operator": "0",
                    "value": "postgresql"
                }
            ],
            "schedule": [
                {
                    "period_from": "0",
                    "period_to": "601200"
                }
            ],
            "excluded_downtimes": [
                {
                    "name": "Software version upgrade rollout",
                    "period_from": "1648760400",
                    "period_to": "1648764900"
                }
            ]
        }
    },
    "id": 1
}
```

Источник

CSla:get() в *ui/include/classes/api/services/CSla.php*.

**sla.getsli**

Описание

`object sla.getsli(  )`

Этот метод позволяет рассчитать данные индикатора уровня обслуживания (SLI) для соглашения об уровне обслуживания (SLA).

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие SLA ID, отчетные периоды и, при необходимости, ID сервисов, для которых вычисляется SLI.

| Parameter | Type | Description |
|---|---|---|
| slaid | ID | ID SLA, для которого требуется вернуть информацию о доступности.<br><br>Поведение параметра:<br>- *required* |
| period_from | timestamp | Начальная временная метка (включительно) для отчета SLI.<br><br>Possible values: Unix timestamp. |
| period_to | timestamp | Конечная временная метка (включительно) для отчета SLI.<br><br>Possible values: Unix timestamp. |
| periods | integer | Количество периодов для отчета.<br><br>Possible values: 1-100 |
| serviceids | ID/array | ID сервисов, для которых требуется вернуть SLI. |

Разбиение периодов

Следующая таблица демонстрирует порядок возвращаемых фрагментов периодов в зависимости от комбинаций параметров.

**Note:**
Возвращаемые периоды не предшествуют первому доступному периоду (на основе даты вступления SLA в силу) и не выходят за пределы текущего периода.

| Parameters | | | Returned periods |
|---|---|---|---|
| **period_from** | **period_to** | **periods** | |
| - | - | - | Последние 20 периодов, включая текущий. |
| - | - | N | Последние N периодов. |
| - | specified | - | Последние 20 периодов перед `period_to`. |
| - | specified | N | Последние N периодов перед `period_to`. |
| specified | - | - | Первые 20 периодов, начиная с `period_from`. |
| specified | - | N | Первые N периодов, начиная с `period_from`. |
| specified | specified | - | До 100 периодов в указанном диапазоне. |
| specified | specified | N | N периодов в указанном диапазоне. |

Возвращаемые значения

`(object)` Возвращает результаты вычисления.

| Property | Type | Description |
|---|---|---|
| periods | array | Список отчетных периодов.<br><br>Каждый отчетный период представлен объектом, состоящим из:<br>- `period_from` - (`timestamp`) Начальная метка времени (включительно) отчетного периода.<br>- `period_to` - (`timestamp`) Конечная метка времени (исключительно) отчетного периода.<br><br>Периоды отсортированы по `period_from`, при этом более ранние периоды идут первыми. |
| serviceids | array | Список ID сервисов в отчетных периодах.<br><br>Порядок сортировки списка не определен. Даже если параметр `serviceids` был передан методу `sla.getsli`. |
| sli | array | Данные SLI (в виде **двумерного массива**) для каждого отчетного периода и сервиса.<br><br>Индекс свойства `periods` используется как **первое** измерение свойства `sli`.<br><br>Индекс свойства `serviceids` используется как **второе** измерение свойства `sli`. |

Данные SLI

Данные SLI, возвращаемые для каждого указанного периода и сервиса, состоят из:

| Property | Type | Description |
|---|---|---|
| uptime | integer | Количество времени, которое сервис провел в состоянии *OK* в течение запланированного времени работы, за вычетом исключенных простоев. |
| downtime | integer | Количество времени, которое сервис провел в состоянии *not OK* в течение запланированного времени работы, за вычетом исключенных простоев. |
| sli | float | SLI (в процентах от общего времени работы), основанный на uptime и downtime. |
| error_budget | integer | Бюджет ошибок (в секундах), основанный на SLI и SLO. |

| Property | Type | Description |
|---|---|---|
| excluded_downtimes | array | Массив исключенных простоев в этом отчетном периоде.<br><br>Каждый объект будет содержать следующие параметры:<br>- `name` - (`string`) Имя исключенного простоя.<br>- `period_from` - (`timestamp`) Начальная метка времени (включительно) исключенного простоя.<br>- `period_to` - (`timestamp`) Конечная метка времени (исключительно) исключенного простоя.<br><br>Исключенные простои сортируются по `period_from`, при этом более ранние периоды отображаются первыми. |

Примеры

Расчет SLI для ежедневного SLA

Получите данные SLI для сервисов с ID "1" и "4", которые связаны с SLA с ID "1". Получите данные для одного периода до "1761861599" (Oct 30 2025 23:59:59 GMT+0200). Поскольку период отчетности SLA является ежедневным, данные SLI извлекаются с "1761775200" (Oct 30 2025 00:00:00 GMT+0200) по "1761861600" (Oct 31 2025 00:00:00 GMT+0200).

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "1",
        "serviceids": [
            1,
            4
        ],
        "periods": 1,
        "period_to": 1761861599
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Расчет SLI для ежемесячного SLA

Получите данные SLI по услугам с ID ”50”, ”60” и ”70”, которые связаны с SLA с ID ”5”. Получите данные за три периода, начиная с ”1635724800” (Nov 01 2021 00:00:00 UTC). Поскольку период отчетности SLA является ежемесячным, данные SLI извлекаются за следующие три месяца:

- С ”1635724800” (Nov 01 2021 00:00:00 UTC) по ”1638316800” (Dec 01 2021 00:00:00 UTC)
- С ”1638316800” (Dec 01 2021 00:00:00 UTC) по ”1640995200” (Jan 01 2022 00:00:00 UTC)
- С ”1640995200” (Jan 01 2022 00:00:00 UTC) по ”1643673600” (Feb 01 2022 00:00:00 UTC)

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "5",
        "serviceids": [
            50,
            60,
            70
        ],
        "periods": 3,
        "period_from": 1635724800
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "periods": [
            {
                "period_from": 1635724800,
                "period_to": 1638316800
            },
            {
                "period_from": 1638316800,
```

```
                "period_to": 1640995200
            },
            {
                "period_from": 1640995200,
                "period_to": 1643673600
            }
        ],
        "serviceids": [
            50,
            60,
            70
        ],
        "sli": [
            [
                {
                    "uptime": 1186212,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Nov 25 - Dec 01",
                            "period_from": 1637836212,
                            "period_to": 1638316800
                        }
                    ]
                },
                {
                    "uptime": 1186212,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Nov 25 - Dec 01",
                            "period_from": 1637836212,
                            "period_to": 1638316800
                        }
                    ]
                },
                {
                    "uptime": 1186212,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Nov 25 - Dec 01",
                            "period_from": 1637836212,
                            "period_to": 1638316800
                        }
                    ]
                }
            ],
            [
                {
                    "uptime": 1147548,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
```

```
                            "name": "Maintenance Dec 02 - Dec 10",
                            "period_from": 1638439200,
                            "period_to": 1639109652
                        }
                    ]
                },
                {
                    "uptime": 1147548,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Dec 02 - Dec 10",
                            "period_from": 1638439200,
                            "period_to": 1639109652
                        }
                    ]
                },
                {
                    "uptime": 1147548,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": [
                        {
                            "name": "Maintenance Dec 02 - Dec 10",
                            "period_from": 1638439200,
                            "period_to": 1639109652
                        }
                    ]
                }
            ],
            [
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                },
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                },
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                }
            ]
        ]
    },
    "id": 1
}
```

Источник

CSla::getSli() в *ui/include/classes/api/services/CSla.php*

**sla.update**

Описание

`object sla.update( / slaids)`

Этот метод позволяет обновлять существующие записи SLA.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства SLA, которые необходимо обновить.

Свойство `slaid` должно быть определено для каждого SLA, все остальные свойства являются необязательными. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам SLA, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| service_tags | array | Теги службы SLA для замены текущих тегов службы SLA. |
| schedule | array | Расписание SLA для замены текущего расписания.<br>Если параметр указан пустым, это будет интерпретировано как расписание 24x7. |
| excluded_downtimes | array | Исключенные периоды недоступности SLA для замены текущих. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных SLA в свойстве `slaids`.

Примеры

Обновление тегов сервиса

Сделать так, чтобы SLA с ID "5" рассчитывалась с ежемесячными интервалами для сервисов, связанных с NoSQL, без изменения ее расписания или исключенных простоев; установить SLO на 95%.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Изменение расписания SLA

Переключите SLA с ID ”5” на расписание 24x7.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "sla.update",
    "params": {
        "slaid": "5",
        "schedule": []
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Изменение исключенных простоев для SLA

Добавьте запланированный простой на 4 часа для обновления RAM 6 апреля 2022 года, при этом сохраните (нужно определить заново) ранее существовавшее запланированное обновление программного обеспечения 4 июля для SLA с ID ”5”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CSla::update() в *ui/include/classes/api/services/CSla.php*.

### Token

Этот класс предназначен для работы с токенами.

Ссылки на объекты:

- Token

Доступные методы:

- token.create - создать новые токены
- token.delete - удалить токены
- token.get - получить токены
- token.update - обновить токены
- token.generate - сгенерировать токены

#### Объект token

Следующие объекты напрямую связаны с API `token`.

Токен

Объект token имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tokenid | ID | ID токена.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя токена.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| description | text | Описание токена. |
| userid | ID | ID пользователя, которому назначен токен.<br><br>По умолчанию: *текущий пользователь*.<br><br>Поведение свойства:<br>- *константа* |

| Property | Type | Description |
|---|---|---|
| lastaccess | timestamp | Наиболее поздняя дата и время, когда токен был аутентифицирован.<br><br>”0”, если токен ни разу не был аутентифицирован.<br><br>Поведение свойства:<br>- *только для чтения* |
| status | integer | Статус токена.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включенный токен;<br>1 - отключенный токен. |
| expires_at | timestamp | Дата и время истечения срока действия токена.<br><br>”0” для токенов без срока действия. |
| created_at | timestamp | Дата и время создания токена.<br><br>Поведение свойства:<br>- *только для чтения* |
| creator_userid | ID | ID пользователя, создавшего токен.<br><br>Поведение свойства:<br>- *только для чтения* |

**token.create**

Описание

`object token.create( / tokens)`

Этот метод позволяет создавать новые токены.

**Note:**
Для роли пользователя требуется разрешение *Управление токенами API*, чтобы управлять токенами для других пользователей.

**Attention:**
Токен, созданный этим методом, также должен быть сгенерирован перед использованием.

Параметры

( / ) Создаваемые токены.

Этот метод принимает токены со стандартными свойствами токена.

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных токенов, указанных в свойстве `tokenids`. Порядок возвращаемых ID совпадает с порядком переданных токенов.

Примеры

Создание токена

Создайте включенный токен, срок действия которого никогда не истекает, и который выполняет аутентификацию пользователя с ID 2.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Your token",
        "userid": "2"
```

```
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "188"
        ]
    },
    "id": 1
}
```

Создайте отключенный токен, срок действия которого истекает 21 января 2021 года. Этот токен будет выполнять аутентификацию текущего пользователя.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Your token",
        "status": "1",
        "expires_at": "1611238072"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "189"
        ]
    },
    "id": 1
}
```

Источник

CToken::create() в *ui/include/classes/api/services/CToken.php*.

**token.delete**

Описание

`object token.delete(    tokenids)`

Этот метод позволяет удалять токены.

> **Note:**
> Для роли пользователя требуется разрешение *Управление токенами API*, чтобы управлять токенами для других пользователей.

Parameters

`(array)` ID токенов для удаления.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удалённых токенов, указанных в свойстве `tokenids`.

Примеры

Удаление нескольких токенов

Удалить два токена.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "token.delete",
    "params": [
        "188",
        "192"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "188",
            "192"
        ]
    },
    "id": 1
}
```

Источник

CToken::delete() в *ui/include/classes/api/services/CToken.php*.

**token.generate**

Описание

`object token.generate(    tokenids)`

Этот метод позволяет генерировать токены.

**Note:**
Для роли пользователя требуется разрешение *Управление токенами API*, чтобы управлять токенами для других пользователей.

**Attention:**
Токен может быть сгенерирован этим методом только в том случае, если он был создан.

Параметры

(    ) ID токенов для генерации.

Возвращаемые значения

(array) Возвращает массив объектов, содержащих ID сгенерированного токена в свойстве `tokenid` и сгенерированную строку авторизации в свойстве `token`.

| Property | Type | Description |
|---|---|---|
| tokenid | ID | ID токена. |
| token | string | Сгенерированная строка авторизации для этого токена. |

Примеры

Генерация нескольких токенов

Сгенерируйте два токена.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "token.generate",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "tokenid": "1",
            "token": "bbcfce79a2d95037502f7e9a534906d3466c9a1484beb6ea0f4e7be28e8b8ce2"
        },
        {
            "tokenid": "2",
            "token": "fa1258a83d518eabd87698a96bd7f07e5a6ae8aeb8463cae33d50b91dd21bd6d"
        }
    ],
    "id": 1
}
```

Источник

CToken::generate() в *ui/include/classes/api/services/CToken.php*.

**token.get**

Описание

`/ token.get( )`

Этот метов позволяет получать токены в соответствии с заданными параметрами.

**Note:**
Только пользователю типа *Super admin* разрешено видеть токены других пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| tokenids | ID/array | Возвращать только токены с указанными ID. |
| userids | ID/array | Возвращать только токены, созданные для указанных пользователей. |
| token | string | Возвращать только токены, созданные для указанного *Auth token*. |
| valid_at | timestamp | Возвращать только токены, которые действительны (не истекли) на указанную дату и время. |
| expired_at | timestamp | Возвращать только токены, которые истекли (не действительны) на указанную дату и время. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `tokenid`, `name`, `lastaccess`, `status`, `expires_at`, `created_at`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| excludeSearch | boolean | |
| filter | object | |

| Parameter | Type | Description |
|---|---|---|
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение токена

Получить все данные для токена с ID ”2”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "token.get",
    "params": {
        "output": "extend",
        "tokenids": "2"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Источник

CToken::get() в *ui/include/classes/api/services/CToken.php*.

**token.update**

Описание

`object token.update( / tokens)`

Этот метод позволяет обновлять существующие токены.

**Note:**
Для роли пользователя требуется разрешение *Управление токенами API*, чтобы управлять токенами для других пользователей.

Параметры

`(object/array)` Свойства токена, которые нужно обновить.

Свойство `tokenid` должно быть определено для каждого токена, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Метод принимает токены со стандартными свойствами токена.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновлённых токенов, указанных в свойстве `tokenids`.

Примеры

Удалить срок действия токена

Удалите дату истечения срока действия токена.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "token.update",
    "params": {
        "tokenid": "2",
        "expires_at": "0"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "2"
        ]
    },
    "id": 1
}
```

Источник

CToken::update() в *ui/include/classes/api/services/CToken.php*.

**Автоматическая регистрация**

Этот класс предназначен для работы с автоматической регистрацией.

Ссылки на объекты:

- Autoregistration

Доступные методы:

- autoregistration.get - получить данные автоматической регистрации
- autoregistration.update - обновить данные автоматической регистрации

**Объект авторегистрации**

Следующие объекты напрямую связаны с API .

Автоматическая регистрация

Объект автоматической регистрации имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tls_accept | integer | Тип разрешенных входящих подключений для автоматической регистрации.<br><br>Возможные значения:<br>1 - разрешить незашифрованные подключения;<br>2 - разрешить TLS с PSK;<br>3 - разрешить как незашифрованные подключения, так и подключения TLS с PSK. |
| tls_psk_identity | string | Идентификатор PSK; должен быть связан только с одним PSK (в рамках автоматической регистрации, узлов сети и прокси).<br><br>Не включайте конфиденциальную информацию в идентификатор PSK, так как он передается по сети без шифрования, чтобы сообщить получателю, какой PSK использовать.<br><br>Поведение свойства:<br>- *только для записи* |
| tls_psk | string | Предварительно общий ключ (PSK); должен содержать не менее 32 шестнадцатеричных цифр.<br><br>Поведение свойства:<br>- *только для записи* |

**autoregistration.get**

Описание

`object autoregistration.get( )`

Метод позволяет получить объект авторегистрации по заданным параметрам.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает только один параметр.

| Parameter | Type | Description |
|---|---|---|
| output | query | Этот параметр описан в справочном комментарии. |

Возвращаемые значения

`(object)` Возвращает объект авторегистрации.

Примеры

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "autoregistration.get",
    "params": {
    "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
    "tls_accept": "3"
    },
    "id": 1
}
```

Источник

CAutoregistration::get() в *ui/include/classes/api/services/CAutoregistration.php*.

**autoregistration.update**

Описание

`object autoregistration.update( )`

Этот метод позволяет обновить существующую авторегистрацию.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Свойства авторегистрации для обновления.

Возвращаемые значения

`(boolean)` Возвращает логическое значение true в качестве результата при успешном обновлении.

Примеры

Запрос:

```
{
   "jsonrpc": "2.0",
   "method": "autoregistration.update",
   "params": {
      "tls_accept": "3",
      "tls_psk_identity": "PSK 001",
      "tls_psk": "11111595725ac58dd977beef14b97461a7c1045b9a1c923453302c5473193478"
   },
   "auth": "038e1d7b1735c6a5436ee9eae095879e",
   "id": 1
}
```

Ответ:

```
{
   "jsonrpc": "2.0",
   "result": true,
   "id": 1
}
```

Источник

CAutoregistration::update() в *ui/include/classes/api/services/CAutoregistration.php*.

**Аутентификация**

Этот класс предназначен для работы с настройками аутентификации.

Ссылки на объекты:

- Authentication

Доступные методы:

- authentication.get - получить настройки аутентификации
- authentication.update - обновить настройки аутентификации

**Authentication object**

Следующие объекты напрямую связаны с `authentication` API.

Аутентификация

Объект authentication имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| authentication_type | integer | Аутентификация по умолчанию.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Внутренняя;<br>1 - LDAP. |
| http_auth_enabled | integer | HTTP-аутентификация.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение свойства:<br>- *поддерживается*, если `$ALLOW_HTTP_AUTH` включен в файле конфигурации веб-интерфейса (*zabbix.conf.php*). |
| http_login_form | integer | Форма входа по умолчанию.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Форма входа Zabbix;<br>1 - HTTP-форма входа.<br><br>Поведение свойства:<br>- *поддерживается*, если `$ALLOW_HTTP_AUTH` включен в файле конфигурации веб-интерфейса (*zabbix.conf.php*). |
| http_strip_domains | string | Имя домена для удаления.<br><br>Поведение свойства:<br>- *поддерживается*, если `$ALLOW_HTTP_AUTH` включен в файле конфигурации веб-интерфейса (*zabbix.conf.php*). |
| http_case_sensitive | integer | HTTP-аутентификация с учетом регистра.<br><br>Возможные значения:<br>0 - Выкл.;<br>1 - *(по умолчанию)* Вкл.<br><br>Поведение свойства:<br>- *поддерживается*, если `$ALLOW_HTTP_AUTH` включен в файле конфигурации веб-интерфейса (*zabbix.conf.php*). |
| ldap_auth_enabled | integer | LDAP-аутентификация.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| ldap_case_sensitive | integer | LDAP-аутентификация с учетом регистра.<br><br>Возможные значения:<br>0 - Выкл.;<br>1 - *(по умолчанию)* Вкл. |

| Property | Type | Description |
|---|---|---|
| ldap_userdirectoryid | ID | ID каталога пользователей по умолчанию для LDAP-аутентификации.<br>Используется для групп пользователей с `gui_access`, установленным в LDAP или System default.<br><br>Поведение свойства:<br>- *обязательно*, если `ldap_auth_enabled` установлено в "Enabled" |
| saml_auth_enabled | integer | SAML-аутентификация.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| saml_case_sensitive | integer | SAML-аутентификация с учетом регистра.<br><br>Возможные значения:<br>0 - Выкл.;<br>1 - *(по умолчанию)* Вкл. |
| passwd_min_length | integer | Минимальная требуемая длина пароля.<br><br>Возможные значения: от 1 до 70.<br><br>По умолчанию: 8. |
| passwd_check_rules | integer | Правила проверки пароля.<br><br>Возможные значения битовой карты:<br>0 - Проверять длину пароля;<br>1 - Проверять, использует ли пароль латинские буквы в верхнем и нижнем регистре;<br>2 - Проверять, использует ли пароль цифры;<br>4 - Проверять, использует ли пароль специальные символы;<br>8 - *(по умолчанию)* Проверять, что пароль не входит в список часто используемых паролей и не содержит производных от слова "Zabbix" или имени, фамилии либо имени пользователя.<br><br>Это поле является битовой маской; допустима любая сумма возможных значений битовой карты (например, 15 для проверки всех правил). |
| ldap_jit_status | integer | Состояние LDAP-провижининга.<br><br>Возможные значения:<br>0 - Отключено для настроенных LDAP IdP;<br>1 - Включено для настроенных LDAP IdP. |
| saml_jit_status | integer | Состояние SAML-провижининга.<br><br>Возможные значения:<br>0 - Отключено для настроенных SAML IdP;<br>1 - Включено для настроенных SAML IdP. |
| jit_provision_interval | string | Интервал времени между запросами JIT-провижининга для вошедшего в систему пользователя.<br>Принимает секунды и единицы времени с суффиксом, с поддержкой месяцев и лет (3600s,60m,1h,1d,1M,1y). Минимальное значение: 1h.<br><br>По умолчанию: 1h.<br><br>Доступно только для LDAP-провижининга. |

| Property | Type | Description |
|---|---|---|
| disabled_usrgrpid | ID | ID группы пользователей, в которую назначается деактивированный пользователь.<br>Группа пользователей должна быть отключена и не может быть включена или удалена после настройки.<br><br>Поведение свойства:<br>- *обязательно*, если `ldap_jit_status` установлено в "Enabled for configured LDAP IdPs", или `saml_jit_status` установлено в "Enabled for configured SAML IdPs" |
| mfa_status | integer | Многофакторная аутентификация.<br><br>Возможные значения:<br>0 - Отключено (для всех настроенных методов MFA);<br>1 - Включено (для всех настроенных методов MFA). |
| mfaid | ID | Метод MFA по умолчанию для групп пользователей с включенной MFA.<br><br>Поведение свойства:<br>- *обязательно*, если `mfa_status` установлено в "Enabled" |

**authentication.get**

Описание

`object authentication.get(parameters)`

Метод позволяет получить объект аутентификации в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен только для типа пользователя *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Чтобы получить больше информации смотрите Роли пользователя.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает только один параметр.

| Parameter | Type | Description |
|---|---|---|
| output | query | Этот параметр описан в справочном комментарии. |

Возвращаемые значения

`(object)` Возвращает объект аутентификации.

Примеры

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Источник

CAuthentication::get() в *ui/include/classes/api/services/CAuthentication.php*.

**authentication.update**

Описание

`authentication.update( authentication)`

Этот метод позволяет обновить существующие настройки аутентификации.

> **Note:**
> Этот метод доступен только для типа пользователя *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. См. Роли пользователей, чтобы получить больше информации.

Параметры

`(object)` Свойства аутентификации подлежат обновлению.

Возвращаемые значения

`(array)` Возвращает массив с именами обновленных параметров.

Примеры

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "authentication.update",
    "params": {
        "http_auth_enabled": 1,
        "http_case_sensitive": 0,
        "http_login_form": 1
    },
    "id": 1
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": [
        "http_auth_enabled",
```

```
        "http_case_sensitive",
        "http_login_form"
    ],
    "id": 1
}
```

Источник

CAuthentication::update() в *ui/include/classes/api/services/CAuthentication.php*.

#### Веб-сценарий

Этот класс предназначен для работы с веб-сценариями.

Ссылки на объекты:

- Веб-сценарий
- Шаг сценария
  - Поле HTTP
- Тег веб-сценария

Доступные методы:

- httptest.create - создать новые веб-сценарии
- httptest.delete - удалить веб-сценарии
- httptest.get - получить веб-сценарии
- httptest.update - обновить веб-сценарии

#### Объект веб-сценария

Следующие объекты напрямую связаны с API `webcheck`.

Веб-сценарий

Объект веб-сценария имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| httptestid | ID | ID веб-сценария.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| hostid | ID | ID узла сети, которому принадлежит веб-сценарий.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |
| name | string | Имя веб-сценария.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| agent | string | Строка user agent, которая будет использоваться веб-сценарием.<br><br>По умолчанию: Zabbix |
| authentication | integer | Метод аутентификации, который будет использоваться веб-сценарием.<br><br>Возможные значения:<br>0 - *(по умолчанию)* нет;<br>1 - базовая HTTP-аутентификация;<br>2 - аутентификация NTLM;<br>3 - аутентификация Kerberos;<br>4 - аутентификация Digest. |

| Property | Type | Description |
|---|---|---|
| delay | string | Интервал выполнения веб-сценария.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d), либо пользовательский макрос.<br><br>По умолчанию: 1m. |
| headers | array | HTTP-заголовки, которые будут отправляться при выполнении запроса. |
| http_password | string | Пароль, используемый для базовой HTTP-аутентификации, NTLM, Kerberos или Digest-аутентификации. |
| http_proxy | string | Прокси, который будет использоваться веб-сценарием, в виде *http://[username[:password]@]proxy.example.com[:port]*. |
| http_user | string | Имя пользователя, используемое для базовой HTTP-аутентификации, NTLM, Kerberos или Digest-аутентификации. |
| retries | integer | Количество попыток, которые веб-сценарий выполнит для каждого шага перед сбоем.<br><br>По умолчанию: 1. |
| ssl_cert_file | string | Имя файла SSL-сертификата, используемого для аутентификации клиента (должен быть в формате PEM). |
| ssl_key_file | string | Имя файла закрытого ключа SSL, используемого для аутентификации клиента (должен быть в формате PEM). |
| ssl_key_password | string | Пароль закрытого ключа SSL. |
| status | integer | Включен ли веб-сценарий.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включен;<br>1 - отключен. |
| templateid | ID | ID родительского шаблона веб-сценария.<br><br>Поведение свойства:<br>- *только для чтения* |
| variables | array | Переменные веб-сценария. |
| verify_host | integer | Следует ли проверять, что имя узла для соединения совпадает с именем в сертификате узла.<br><br>Возможные значения:<br>0 - *(по умолчанию)* пропустить проверку узла;<br>1 - проверять узел. |
| verify_peer | integer | Следует ли проверять, что сертификат узла является подлинным.<br><br>Возможные значения:<br>0 - *(по умолчанию)* пропустить проверку peer;<br>1 - проверять peer. |
| uuid | string | Глобальный уникальный идентификатор, используемый для связывания импортированных веб-сценариев с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается*, если веб-сценарий принадлежит шаблону |

Шаг сценария

Объект шага сценария определяет конкретную проверку веб-сценария. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя шага сценария.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| no | integer | Порядковый номер шага в веб-сценарии.<br><br>Поведение свойства:<br>- *required* |
| url | string | URL для проверки.<br><br>Поведение свойства:<br>- *required* |
| follow_redirects | integer | Следует ли выполнять HTTP-перенаправления.<br><br>Возможные значения:<br>0 - не следовать перенаправлениям;<br>1 - *(по умолчанию)* следовать перенаправлениям. |
| headers | array | HTTP-заголовки, которые будут отправлены при выполнении запроса. Заголовки шага сценария будут переопределять заголовки, указанные для веб-сценария. |
| posts | string/array | Переменные HTTP POST в виде строки (сырые данные POST) или массива HTTP-полей (данные полей формы). |
| required | string | Текст, который должен присутствовать в ответе. |
| retrieve_mode | integer | Часть HTTP-ответа, которую должен получить шаг сценария.<br><br>Возможные значения:<br>0 - *(по умолчанию)* только тело;<br>1 - только заголовки;<br>2 - заголовки и тело. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми. |
| timeout | string | Тайм-аут запроса в секундах. Принимает секунды, единицу времени с суффиксом или пользовательский макрос.<br><br>По умолчанию: 15s. Максимум: 1h. Минимум: 1s. |
| variables | array | Переменные шага сценария. |
| query_fields | array | Поля запроса - массив HTTP-полей, которые будут добавлены к URL при выполнении запроса. |

Поле HTTP

Объект поля HTTP определяет имя и значение, которые используются для указания переменных веб-сценария, HTTP-заголовков, а также полей POST или полей запроса. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя заголовка/переменной/поля POST или GET.<br><br>Property behavior:<br>- *required* |
| value | string | Значение заголовка/переменной/поля POST или GET.<br><br>Property behavior:<br>- *required* |

Тег веб-сценария

Объект тега веб-сценария имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега веб-сценария.<br><br>Поведение свойства:<br>- *обязательно* |

| Property | Type | Description |
|---|---|---|
| value | string | Значение тега веб-сценария. |

**httptest.create**

Описание

`object httptest.create(  /  webScenarios)`

Этот метод позволяет создавать новые веб-сценарии.

**Note:**
При создании веб-сценария также будет автоматически создан набор элементов данных веб-мониторинга.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Веб-сценарии для создания.

Помимо стандартных свойств веб-сценария, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| steps | array | Шаги сценария.<br>Поведение параметра:<br>- *required* |
| tags | array | Теги веб-сценария. |

Возвращаемые значения

`(object)` Возвращает объект, который содержит ID созданных веб-сценариев под свойством `httptestids`. Порядок возвращаемых ID совпадает с порядком переданных веб-сценариев.

Примеры

Создание веб-сценария

Создайте веб-сценарий для мониторинга домашней страницы компании. Сценарий будет состоять из двух шагов: проверки домашней страницы и страницы "About", чтобы убедиться, что они возвращают код состояния HTTP 200.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

См. также

- Шаг сценария

Источник

CHttpTest::create() в *ui/include/classes/api/services/CHttpTest.php*.

**httptest.delete**

Описание

`object httptest.delete(    webScenarioIds)`

Этот метод позволяет удалять веб-сценарии.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` ID удаляемых веб-сценариев.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных веб-сценариев в свойстве `httptestids`.

Примеры

Удаление нескольких веб-сценариев

Удаление двух веб-сценариев.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.delete",
    "params": [
        "2",
        "3"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
```

```
            "2",
            "3"
        ]
    },
    "id": 1
}
```

Источник

CHttpTest::delete() в *ui/include/classes/api/services/CHttpTest.php*.

**httptest.get**

Описание

`integer/array httptest.get(object parameters)`

Метод позволяет получать веб-сценарии в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groupids | ID/array | Возвращать только веб-сценарии, которые принадлежат указанным группам узлов сети. |
| hostids | ID/array | Возвращать только веб-сценарии, которые принадлежат указанным узлам сети. |
| httptestids | ID/array | Возвращать только веб-сценарии с указанными ID. |
| inherited | boolean | Если установлено значение `true`, возвращать только веб-сценарии, унаследованные от шаблона. |
| monitored | boolean | Если установлено значение `true`, возвращать только включенные веб-сценарии, которые принадлежат контролируемым узлам сети. |
| templated | boolean | Если установлено значение `true`, возвращать только веб-сценарии, которые принадлежат шаблонам. |
| templateids | ID/array | Возвращать только веб-сценарии, которые принадлежат указанным шаблонам. |
| expandName | flag | Расширять макросы в имени веб-сценария. |
| expandStepName | flag | Расширять макросы в именах шагов сценария. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И;<br>2 - ИЛИ. |
| tags | array | Возвращать только веб-сценарии с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все веб-сценарии.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| selectHosts | query | Возвращать узлы сети, которым принадлежит веб-сценарий, в виде массива в свойстве `hosts`. |

| Parameter | Type | Description |
|---|---|---|
| selectSteps | query | Возвращать шаги веб-сценария в свойстве `steps`.<br><br>Поддерживает `count`. |
| selectTags | query | Возвращать теги веб-сценария в свойстве `tags`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `httptestid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение веб-сценария

Получить все данные о веб-сценарии "9".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.get",
    "params": {
        "output": "extend",
        "selectSteps": "extend",
        "httptestids": "9"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "httptestid": "9",
            "name": "Homepage check",
            "delay": "1m",
            "status": "0",
            "variables": [],
            "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",
            "headers": [],
            "steps": [
                {
                    "httpstepid": "36",
                    "httptestid": "9",
                    "name": "Homepage",
                    "no": "1",
                    "url": "http://example.com",
                    "timeout": "15s",
                    "posts": "",
                    "required": "",
                    "status_codes": "200",
                    "variables": [
                        {
                            "name":"{var}",
                            "value":"12"
                        }
                    ],
                    "follow_redirects": "1",
                    "retrieve_mode": "0",
                    "headers": [],
                    "query_fields": []
                },
                {
                    "httpstepid": "37",
                    "httptestid": "9",
                    "name": "Homepage / About",
                    "no": "2",
                    "url": "http://example.com/about",
                    "timeout": "15s",
                    "posts": "",
                    "required": "",
                    "status_codes": "200",
                    "variables": [],
                    "follow_redirects": "1",
                    "retrieve_mode": "0",
                    "headers": [],
                    "query_fields": []
                }
            ]
        }
    ],
    "id": 1
}
```


См. также

- узел сети
- Шаг сценария

Источник

CHttpTest::get() в *ui/include/classes/api/services/CHttpTest.php*.

**httptest.update**

Описание

`object httptest.update(  /  webScenarios)`

Этот метод позволяет обновлять существующие веб-сценарии.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства веб-сценария, которые необходимо обновить.

Для каждого веб-сценария должно быть определено свойство `httptestid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств веб-сценария, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| steps | array | Шаги сценария для замены существующих шагов. |
| tags | array | Теги веб-сценария. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных веб-сценариев, указанных в свойстве `httptestid`.

Примеры

Включение веб-сценария

Включите веб-сценарий, то есть установите для него статус "0".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.update",
    "params": {
        "httptestid": "5",
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

См. также

- Шаг сценария

Источник

CHttpTest::update() в *ui/include/classes/api/services/CHttpTest.php*.

**Граф**

Этот класс предназначен для работы с графиками.

Ссылки на объекты:

- Graph

Доступные методы:

- graph.create - создать новые графики
- graph.delete - удалить графики
- graph.get - получить графики
- graph.update - обновить графики

**Объект graph**

Следующие объекты напрямую связаны с API `graph`.

График

Объект графика имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| graphid | ID | ID графика.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| height | integer | Высота графика в пикселях.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Имя графика.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| width | integer | Ширина графика в пикселях.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| flags | integer | Происхождение графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* обычный график;<br>4 - график, преобразованный из прототипа.<br><br>Поведение свойства:<br>- *только для чтения* |
| graphtype | integer | Тип компоновки графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* обычный;<br>1 - stacked;<br>2 - pie;<br>3 - exploded. |
| percent_left | float | Левый процентиль.<br><br>По умолчанию: 0. |
| percent_right | float | Правый процентиль.<br><br>По умолчанию: 0. |
| show_3d | integer | Показывать ли круговые и exploded-графики в 3D.<br><br>Возможные значения:<br>0 - *(по умолчанию)* показывать в 2D;<br>1 - показывать в 3D. |

| Property | Type | Description |
|---|---|---|
| show_legend | integer | Показывать ли легенду на графике.<br><br>Возможные значения:<br>0 - скрыть;<br>1 - *(по умолчанию)* показать. |
| show_work_period | integer | Показывать ли рабочее время на графике.<br><br>Возможные значения:<br>0 - скрыть;<br>1 - *(по умолчанию)* показать. |
| show_triggers | integer | Показывать ли линию триггера на графике.<br><br>Возможные значения:<br>0 - скрыть;<br>1 - *(по умолчанию)* показать. |
| templateid | ID | ID родительского графика шаблона.<br><br>Поведение свойства:<br>- *только для чтения* |
| yaxismax | float | Фиксированное максимальное значение для оси Y.<br><br>По умолчанию: 100. |
| yaxismin | float | Фиксированное минимальное значение для оси Y.<br><br>По умолчанию: 0. |
| ymax_itemid | ID | ID элемента данных, который используется как максимальное значение для оси Y.<br><br>Если у пользователя нет доступа к указанному элементу данных, график отображается так, как если бы для `ymax_type` было установлено значение ”calculated”. |
| ymax_type | integer | Метод расчета максимального значения для оси Y.<br><br>Возможные значения:<br>0 - *(по умолчанию)* calculated;<br>1 - fixed;<br>2 - item. |
| ymin_itemid | ID | ID элемента данных, который используется как минимальное значение для оси Y.<br><br>Если у пользователя нет доступа к указанному элементу данных, график отображается так, как если бы для `ymin_type` было установлено значение ”calculated”. |
| ymin_type | integer | Метод расчета минимального значения для оси Y.<br><br>Возможные значения:<br>0 - *(по умолчанию)* calculated;<br>1 - fixed;<br>2 - item. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных графиков с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается*, если график принадлежит шаблону |

**graph.create**

Описание

`object graph.create(  /  graphs)`

Этот метод позволяет создавать новые графики.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Графики для создания.

В дополнение к стандартным свойствам графика, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| gitems | array | Элементы графика, которые будут созданы для графика.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных графиков, указанных в свойстве graphids. Порядок возвращаемых ID совпадает с порядком переданных графиков.

Примеры

Создание графика

Создайте график с двумя элементами данных.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

См. также

- Элемент данных графика

Источник

CGraph::create() в *ui/include/classes/api/services/CGraph.php*.

### graph.delete

Описание

`object graph.delete(    graphIds)`

Этот метод позволяет удалять графики.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых графиков.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных графиков, указанных в свойстве `graphids`.

Примеры

Удаление нескольких графиков

Удаление двух графиков.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graph.delete",
    "params": [
        "652",
        "653"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Источник

CGraph::delete() в *ui/include/classes/api/services/CGraph.php*.

### graph.get

Описание

`integer/array graph.get(          )`

Этот метод позволяет получать графики в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| graphids | ID/array | Возвращать только графики с указанными ID. |
| groupids | ID/array | Возвращать только графики, которые принадлежат узлам сети или шаблонам в указанных группах узлов сети или группах шаблонов. |
| templateids | ID/array | Возвращать только графики, которые принадлежат указанным шаблонам. |
| hostids | ID/array | Возвращать только графики, которые принадлежат указанным узлам сети. |
| itemids | ID/array | Возвращать только графики, которые содержат указанные элементы данных. |
| templated | boolean | Если установлено в `true`, возвращать только графики, которые принадлежат шаблонам. |
| inherited | boolean | Если установлено в `true`, возвращать только графики, унаследованные от шаблона. |
| expandName | flag | Расширять макросы в имени графика. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения графика. Объект обнаружения графика связывает обнаруженный график с прототипом графика, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`parent_graphid` - `(ID)` ID прототипа графика, из которого был создан график;<br>`status` - `(int)` статус обнаружения графика:<br>0 - *(по умолчанию)* график обнаружен,<br>1 - график больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда будет удален график, который больше не обнаружен. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с правилом низкоуровневого обнаружения, которое создало график. |
| selectHostGroups | query | Возвращать свойство `hostgroups` с группами узлов сети, к которым принадлежит график. |
| selectTemplateGroups | query | Возвращать свойство `templategroups` с группами шаблонов, к которым принадлежит график. |
| selectTemplates | query | Возвращать свойство `templates` с шаблонами, к которым принадлежит график. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, к которым принадлежит график. |
| selectItems | query | Возвращать свойство `items` с элементами данных, используемыми в графике. |
| selectGraphItems | query | Возвращать свойство `gitems` с элементами данных, используемыми в графике. |

| Parameter | Type | Description |
|---|---|---|
| filter | object | Возвращать только те результаты, которые точно соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит график;<br>`hostid` - ID узла сети, к которому принадлежит график. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `graphid`, `name`, `graphtype`. |
| countOutput | boolean | Эти параметры описаны в комментарии к справочнику. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |
| selectGraphDiscovery | query | Возвращать свойство `graphDiscovery` с объектом обнаружения графика. Объекты обнаружения графика связывают график с прототипом графика, из которого он был создан.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryData`. |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение графиков с узлов сети

Получите все графики с узла сети ”10107” и отсортируйте их по имени.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graph.get",
    "params": {
        "output": "extend",
        "hostids": 10107,
        "sortfield": "name"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


См. также

- Правило обнаружения
- Элемент графика
- Элемент данных
- Узел сети
- Группа узлов сети
- Шаблон
- Группа шаблонов

Источник

CGraph::get() в *ui/include/classes/api/services/CGraph.php*.

**graph.update**

Описание

`object graph.update(  /  graphs)`

Этот метод позволяет обновлять существующие графики.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства графика, которые нужно обновить.

Для каждого графика должно быть определено свойство `graphid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам графика метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| gitems | array | Элементы графика для замены существующих элементов графика. Если у элемента графика определено свойство `gitemid`, он будет обновлен, в противном случае будет создан новый элемент графика. |

Возвращаемые значения

`(object)` Возвращает объект, который содержит ID обновленных графиков, указанных в свойстве `graphids`.

Примеры

Установка максимума для шкалы Y

Установите максимум шкалы Y в фиксированное значение 100.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graph.update",
    "params": {
        "graphid": "439",
        "ymax_type": 1,
        "yaxismax": 100
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Источник

CGraph::update() в *ui/include/classes/api/services/CGraph.php*.

### Группа пользователей

Этот класс предназначен для работы с группами пользователей.

Ссылки на объекты:

- Группа пользователей
- Разрешение
- Разрешение на основе тегов

Доступные методы:

- usergroup.create - создать новые группы пользователей
- usergroup.delete - удалить группы пользователей
- usergroup.get - получить группы пользователей
- usergroup.update - обновить группы пользователей

### Объект группы пользователей

Следующие объекты напрямую связаны с API `usergroup`.

Группа пользователей

Объект группы пользователей имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| usrgrpid | ID | ID группы пользователей.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя группы пользователей.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| debug_mode | integer | Включен или отключен режим отладки.<br><br>Возможные значения:<br>0 - *(по умолчанию)* отключен;<br>1 - включен. |
| gui_access | integer | Метод аутентификации веб-интерфейса для пользователей в группе.<br><br>Возможные значения:<br>0 - *(по умолчанию)* использовать метод аутентификации по умолчанию системы;<br>1 - использовать внутреннюю аутентификацию;<br>2 - использовать аутентификацию LDAP;<br>3 - запретить доступ к веб-интерфейсу. |
| mfa_status | integer | Включена или отключена MFA для пользователей в группе.<br><br>Возможные значения:<br>0 - отключена (для всех настроенных методов MFA);<br>1 - включена (для всех настроенных методов MFA). |
| mfaid | ID | Метод MFA, используемый для пользователей в группе.<br><br>Поведение свойства:<br>- *поддерживается* для пользователей типа *Super admin*<br>- *поддерживается*, если `mfa_status` объекта аутентификации установлен в ”Enabled” |
| users_status | integer | Включена или отключена группа пользователей.<br>Для пользователей, deprovisioned, группу пользователей нельзя включить.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включена;<br>1 - отключена. |
| userdirectoryid | ID | ID каталога пользователей, используемого для аутентификации.<br><br>Поведение свойства:<br>- *поддерживается* для пользователей типа *Super admin*<br>- *поддерживается*, если `gui_access` установлен в ”use the system default authentication method” или ”use LDAP authentication” |

Разрешение

Объект разрешения имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| id | ID | ID группы узлов сети или группы шаблонов, к которой нужно добавить разрешение.<br><br>Property behavior:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| permission | integer | Уровень доступа к группе узлов сети или группе шаблонов.<br><br>Возможные значения:<br>0 - доступ запрещен;<br>2 - доступ только для чтения;<br>3 - доступ для чтения и записи.<br><br>Property behavior:<br>- *required* для операций создания |

Разрешение на основе тегов

Объект разрешения на основе тегов имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| groupid | ID | ID группы узлов сети, для которой нужно добавить разрешение.<br><br>Property behavior:<br>- *required* |
| tag | string | Имя тега. |
| value | string | Значение тега. |

**usergroup.create**

Описание

`object usergroup.create(object/array userGroups)`

Этот метод позволяет создавать новые группы пользователей.

**Note:**
Этот метод доступен только для типа пользователя *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе User roles.

Параметры

`(object/array)` Группы пользователей для создания.

В дополнение к стандартным свойствам группы пользователей, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hostgroup_rights | object/array | Права для групп узлов сети, назначаемые группе пользователей. |
| templategroup_rights | object/array | Права для групп шаблонов, назначаемые группе пользователей. |
| tag_filters | array | Права на основе тегов, назначаемые группе пользователей. |
| users | object/array | Пользователи, которых нужно добавить в группу пользователей.<br><br>У пользователя должно быть определено только свойство `userid`. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных групп пользователей в свойстве `usrgrpids`. Порядок возвращаемых идентификаторов соответствует порядку переданных групп пользователей.

Примеры

Создание группы пользователей

Создайте группу пользователей *Operation managers* с запрещенным доступом к группе узлов сети "2" и добавьте в нее пользователя.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.create",
    "params": {
        "name": "Operation managers",
        "hostgroup_rights": {
            "id": "2",
            "permission": 0
        },
        "users": [
            {
                "userid": "12"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "20"
        ]
    },
    "id": 1
}
```

Смотрите также

- Права доступа

Источник

CUserGroup::create() в *ui/include/classes/api/services/CUserGroup.php*.

**usergroup.delete**

Описание

`object usergroup.delete(    userGroupIds)`

Этот метод позволяет удалять группы пользователей.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(  ) ID удаляемых групп пользователей.

Возвращаемые значения

(  ) Возвращает объект, который содержит ID удалённых групп пользователей, указанных в свойстве `usrgrpids`.

Примеры

Удаление нескольких групп пользователей

Удалите две группы пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "usergroup.delete",
    "params": [
        "20",
        "21"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "20",
            "21"
        ]
    },
    "id": 1
}
```

Источник

CUserGroup::delete() в *ui/include/classes/api/services/CUserGroup.php*.

**usergroup.get**

Описание

`integer/array usergroup.get(          )`

Метод позволяет получить группы пользователей в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| mfaids | ID/array | Возвращать только группы пользователей с указанными методами MFA.<br><br>Поведение параметра:<br>- *supported* для пользователей типа *Super admin* |
| mfa_status | integer | Возвращать только группы пользователей с указанным статусом MFA.<br><br>Список поддерживаемых статусов см. на странице группы пользователей. |
| status | integer | Возвращать только группы пользователей с указанным статусом.<br><br>Список поддерживаемых статусов см. на странице группы пользователей. |
| userids | ID/array | Возвращать только группы пользователей, которые содержат указанных пользователей. |
| usrgrpids | ID/array | Возвращать только группы пользователей с указанными ID. |
| selectTagFilters | query | Возвращать разрешения группы пользователей на основе тегов в свойстве `tag_filters`. |

| Parameter | Type | Description |
|---|---|---|
| selectUsers | query | Возвращать пользователей из группы пользователей в свойстве `users`.<br><br>См. `user.get` для ограничений, основанных на типе пользователя. |
| selectHostGroupRights | query | Возвращать разрешения группы пользователей для групп узлов сети в свойстве `hostgroup_rights`.<br><br>Список уровней доступа к группам узлов сети см. на странице группы пользователей. |
| selectTemplateGroupRights | query | Возвращать разрешения группы пользователей для групп шаблонов в свойстве `templategroup_rights`.<br><br>Список уровней доступа к группам шаблонов см. на странице группы пользователей. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами. |
| output | query | Свойства объекта группы пользователей, которые нужно вернуть.<br><br>Пользователи типов *Admin* и *User* могут получать только следующие свойства: `usrgrpid`, `name`, `gui_access`, `users_status`, `debug_mode`, `mfa_status`.<br><br>По умолчанию: `extend`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `usrgrpid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение включенных групп пользователей

Получить все включенные группы пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.get",
    "params": {
        "output": "extend",
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- User

Источник

CUserGroup::get() в *ui/include/classes/api/services/CUserGroup.php*.

**usergroup.update**

Описание

`object usergroup.update( / userGroups)`

Этот метод позволяет обновлять существующие группы пользователей.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства группы пользователей, которые необходимо обновить.

Для каждого группы пользователей должно быть определено свойство `usrgrpid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам группы пользователей, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hostgroup_rights | object/array | Права группы узлов сети, которые заменят текущие права, назначенные группе пользователей. |
| templategroup_rights | object/array | Права группы шаблонов, которые заменят текущие права, назначенные группе пользователей. |
| tag_filters | array | Права на основе тегов, которые заменят текущие права, назначенные группе пользователей. |
| users | object/array | Пользователи, которые заменят текущих пользователей, назначенных группе пользователей.<br><br>У пользователя должно быть определено только свойство `userid`. |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID обновленных групп пользователей, указанных в свойстве `usrgrpids`.

Примеры

Включение группы пользователей и обновление прав доступа

Включите группу пользователей и предоставьте ей доступ на чтение и запись к группам узлов сети ”2” и ”4”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.update",
    "params": {
        "usrgrpid": "17",
        "users_status": "0",
        "hostgroup_rights": [
            {
                "id": "2",
```

```
                "permission": 3
            },
            {
                "id": "4",
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "17"
        ]
    },
    "id": 1
}
```

Смотрите также

- Права доступа

Источник

CUserGroup::update() в *ui/include/classes/api/services/CUserGroup.php*.

#### Группа прокси

Этот класс предназначен для работы с группами прокси.

Ссылки на объекты:

- Группа прокси

Доступные методы:

- proxygroup.create - создать новые группы прокси
- proxygroup.delete - удалить группы прокси
- proxygroup.get - получить группы прокси
- proxygroup.update - обновить группы прокси

#### Объект группы прокси

Следующие объекты напрямую связаны с API `proxygroup`.

Группа прокси

Объект группы прокси имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| proxy_groupid | ID | ID группы прокси.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя группы прокси.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| description | text | Описание группы прокси. |

| Property | Type | Description |
|---|---|---|
| failover_delay | string | Период, в течение которого прокси в группе прокси должен связываться с сервером Zabbix, чтобы считаться онлайн.<br><br>Поддерживаются суффиксы времени, например 30s, 1m.<br>Поддерживаются пользовательские макросы.<br><br>Возможные значения: 10s-15m.<br><br>По умолчанию: 1m. |
| min_online | string | Минимальное количество онлайн-прокси, необходимое для сохранения группы прокси в состоянии онлайн.<br><br>Поддерживаются пользовательские макросы.<br><br>Диапазон возможных значений: 1-1000.<br><br>По умолчанию: 1. |
| state | integer | Состояние группы прокси.<br><br>Возможные значения:<br>0 - Unknown;<br>1 - Offline;<br>2 - Recovering;<br>3 - Online;<br>4 - Degrading.<br><br>Поведение свойства:<br>*- только для чтения* |

**proxygroup.create**

Описание

`object proxygroup.create(  /  proxyGroups)`

Этот метод позволяет создавать новые группы прокси.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Группы прокси для создания.

Метод принимает группы прокси со стандартными свойствами группы прокси.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных групп прокси в свойстве `proxy_groupids`. Порядок возвращаемых идентификаторов соответствует порядку переданных групп прокси.

Примеры

Создание группы прокси

Создайте группу прокси с пользовательскими настройками.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.create",
    "params": {
```

```
        "name": "Proxy group",
        "failover_delay": "5m",
        "min_online": "10"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CProxyGroup::create() в *ui/include/classes/api/services/CProxyGroup.php*.

**proxygroup.delete**

Описание

`object proxygroup.delete(    proxyGroups)`

Этот метод позволяет удалять группы прокси.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы групп прокси для удаления.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных групп прокси в свойстве `proxy_groupids`.

Примеры

Удаление нескольких групп прокси

Удаление двух групп прокси.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.delete",
    "params": [
        "5",
        "10"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5",
```

```
            "10"
        ]
    },
    "id": 1
}
```

Источник

CProxyGroup::delete() в *ui/include/classes/api/services/CProxyGroup.php*.

### proxygroup.get

Описание

`integer/array proxygroup.get(          )`

Метод позволяет получать группы прокси по заданным параметрам.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| proxy_groupids | ID/array | Возвращать только группы прокси с указанными ID. |
| proxyids | ID/array | Возвращать только группы прокси, которые содержат указанные прокси. |
| selectProxies | query | Возвращать свойство `proxies` с прокси, которые принадлежат группе прокси.<br><br>Поддерживает `count`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `proxy_groupid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение всех групп прокси

Получение всех настроенных групп прокси с прокси.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.get",
    "params": {
        "output": "extend",
        "selectProxies": ["proxyid", "name"]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Прокси

Исходный код

CProxyGroup::get() в *ui/include/classes/api/services/CProxyGroup.php*.

**proxygroup.update**

Описание

`object proxygroup.update( / proxyGroups)`

Этот метод позволяет обновлять существующие группы прокси.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства группы прокси, которые нужно обновить.

Свойство `proxy_groupid` должно быть определено для каждой группы прокси, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Метод принимает группы прокси со стандартными свойствами группы прокси.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных групп прокси в свойстве `proxy_groupids`.

Примеры

Изменение минимального числа онлайн-прокси

Измените минимальное число онлайн-прокси, необходимое для того, чтобы группа считалась онлайн.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.update",
    "params": {
        "proxy_groupid": "5",
        "min_online": "3"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CProxyGroup::update() в *ui/include/classes/api/services/CProxyGroup.php*.

**Группа узлов сети**

Этот класс предназначен для работы с группами узлов сети.

Ссылки на объекты:

- Группа узлов сети

Доступные методы:

- hostgroup.create - создать новые группы узлов сети
- hostgroup.delete - удалить группы узлов сети
- hostgroup.get - получить группы узлов сети
- hostgroup.massadd - добавить связанные объекты в группы узлов сети
- hostgroup.massremove - удалить связанные объекты из групп узлов сети
- hostgroup.massupdate - заменить или удалить связанные объекты в группах узлов сети
- hostgroup.propagate - распространить права доступа и фильтры тегов на подгруппы групп узлов сети
- hostgroup.update - обновить группы узлов сети

#### Объект группы узлов сети

Следующие объекты напрямую связаны с API `hostgroup`.

Группа узлов сети

Объект группы узлов сети имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| groupid | ID | ID группы узлов сети.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| name | string | Имя группы узлов сети.<br><br>Property behavior:<br>- *required* for create operations |
| flags | integer | Источник группы узлов сети.<br><br>Possible values:<br>0 - обычная группа узлов сети;<br>4 - обнаруженная группа узлов сети.<br><br>Property behavior:<br>- *read-only* |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных групп узлов сети с уже существующими. Автоматически генерируется, если не указан. |

#### hostgroup.create

Описание

`object hostgroup.create(  /  hostGroups)`

Этот метод позволяет создавать новые группы узлов сети.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Группы узлов сети для создания.

Метод принимает группы узлов сети с стандартными свойствами группы узлов сети.

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID созданных групп узлов сети, указанных в свойстве `groupids`. Порядок возвращаемых ID совпадает с порядком переданных групп узлов сети.

Примеры

Создание группы узлов сети

Создайте группу узлов сети с именем ”Linux servers”.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.create",
    "params": {
        "name": "Linux servers"
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107819"
        ]
    },
    "id": 1
}
```

Источник

CHostGroup::create() в *ui/include/classes/api/services/CHostGroup.php*.

**hostgroup.delete**

Описание

`object hostgroup.delete(    hostGroupIds)`

Этот метод позволяет удалять группы узлов сети.

Группу узлов сети нельзя удалить в случае, если:

- она содержит узлы сети, которые принадлежат только этой группе узлов сети;
- она отмечена как внутренняя;
- она используется прототипом узлов сети;
- она используется в глобальном скрипте;
- она используется в условии корреляции.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых групп узлов сети.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных групп узлов сети, указанных в свойстве`groupids`.

Примеры

Удаление нескольких групп узлов сети

Удаление двух групп узлов сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.delete",
```

```
    "params": [
        "107824",
        "107825"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107824",
            "107825"
        ]
    },
    "id": 1
}
```

Источник

CHostGroup::delete() в *ui/include/classes/api/services/CHostGroup.php*.

**hostgroup.get**

Описание

`integer/array hostgroup.get(object parameters)`

Метод позволяет получать группы узлов сети в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода могут быть отозваны в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| graphids | ID/array | Возвращать только группы узлов сети, которые содержат узлы сети с указанными графиками. |
| groupids | ID/array | Возвращать только группы узлов сети с указанными идентификаторами групп узлов сети. |
| hostids | ID/array | Возвращать только группы узлов сети, которые содержат указанные узлы сети. |
| maintenanceids | ID/array | Возвращать только группы узлов сети, на которые влияют указанные обслуживания. |
| triggerids | ID/array | Возвращать только группы узлов сети, которые содержат узлы сети с указанными триггерами. |
| with_graphs | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с графиками. |
| with_graph_prototypes | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с прототипами графиков. |
| with_hosts | boolean | Возвращать только группы узлов сети, которые содержат узлы сети. |
| with_httptests | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с веб-проверками.<br><br>Переопределяет параметр `with_monitored_httptests`. |

| Parameter | Type | Description |
|---|---|---|
| with_items | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с элементами данных.<br><br>Переопределяет параметры `with_monitored_items` и `with_simple_graph_items`. |
| with_item_prototypes | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с прототипами элементов данных.<br><br>Переопределяет параметр `with_simple_graph_item_prototypes`. |
| with_simple_graph_item_prototypes | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с прототипами элементов данных, которые включены для создания и имеют числовой тип информации. |
| with_monitored_httptests | boolean | Возвращать только группы узлов сети, которые содержат контролируемые узлы сети с включенными веб-проверками. |
| with_monitored_hosts | boolean | Возвращать только группы узлов сети, которые содержат контролируемые узлы сети. |
| with_monitored_items | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с включенными элементами данных.<br><br>Переопределяет параметр `with_simple_graph_items`. |
| with_monitored_triggers | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с включенными триггерами. Все элементы данных, используемые в триггере, также должны быть включены. |
| with_simple_graph_items | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с числовыми элементами данных. |
| with_triggers | boolean | Возвращать только группы узлов сети, которые содержат узлы сети с триггерами.<br><br>Переопределяет параметр `with_monitored_triggers`. |
| selectDiscoveryRules | query | Возвращать свойство `discoveryRules` с правилами LLD, которые обнаружили группу узлов сети. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с объектами обнаружения группы узлов сети.<br><br>Каждый объект обнаружения группы узлов сети представляет собой прототип группы узлов сети, связанный с обнаруженной группой узлов сети, и имеет следующие свойства:<br>`parent_group_prototypeid` - `(ID)` идентификатор прототипа группы узлов сети, из которого была обнаружена группа узлов сети;<br>`name` - `(string)` имя прототипа группы узлов сети;<br>`ts_delete` - `(timestamp)` время, когда будет удалена группа узлов сети, которая больше не обнаруживается;<br>`status` - `(int)` статус обнаружения группы узлов сети:<br>0 - *(по умолчанию)* группа узлов сети обнаружена,<br>1 - группа узлов сети больше не обнаруживается. |
| selectHostPrototypes | query | Возвращать свойство `hostPrototypes` с прототипами узлов сети, которые обнаружили эту группу узлов сети. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, которые принадлежат группе узлов сети.<br><br>Поддерживает `count`. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам:<br>`selectHosts` - результаты будут отсортированы по `host`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `groupid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |

| 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 | |
| selectGroupDiscoveries | query | Возвращать свойство `groupDiscoveries` с объектами обнаружения группы узлов сети.<br><br>Каждый объект обнаружения группы узлов сети представляет собой прототип группы узлов сети, связанный с обнаруженной группой узлов сети.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryData`. |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов
- количество полученных объектов, если использовался параметр `countOutput`

Примеры

Получение данных по имени

Получить все данные о двух группах узлов сети с именами ”Zabbix servers” и ”Linux servers”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Zabbix servers",
                "Linux servers"
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "2",
            "name": "Linux servers",
            "flags": "0",
            "uuid": "dc579cd7a1a34222933f24f52a68bcd8"
        },
        {
            "groupid": "4",
            "name": "Zabbix servers",
```

```
            "flags": "0",
            "uuid": "6f6799aa69e844b4b3918f779f2abf08"
        }
    ],
    "id": 1
}
```

См. также

- Узел сети

Источник

CHostGroup::get() в *ui/include/classes/api/services/CHostGroup.php*.

### hostgroup.massadd

Описание

`object hostgroup.massadd(          )`

Этот метод позволяет добавить одновременно несколько связанных объектов во все заданные группы узлов сети.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие ID групп узлов сети, которые нужно обновить, и объекты, которые нужно добавить во все группы узлов сети.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы узлов сети, которые нужно обновить.<br><br>Группы узлов сети должны иметь определенным только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* |
| hosts | object/array | Узлы сети, которые нужно добавить во все группы узлов сети.<br><br>У узлов сети должно быть определено только свойство `hostid`. |

Возвращаемые значения

`(    )` Возвращает объект, который содержит ID обновленных групп узлов сети, указанных в свойстве `groupids`.

Примеры

Добавление узлов сети в группы узлов сети

Добавьте два узла сети в группы узлов сети с ID 5 и 6.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "5"
            },
```

```
            {
                "groupid": "6"
            }
        ],
        "hosts": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30001"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

См. также

- Узел сети

Источник

CHostGroup::massAdd() в *ui/include/classes/api/services/CHostGroup.php*.

**hostgroup.massremove**

Описание

`object hostgroup.massremove(          )`

Этот метод позволяет удалить связанные объекты из нескольких групп узлов сети.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs групп узлов сети, которые нужно обновить, и объекты, которые следует удалить.

| Parameter | Type | Description |
|---|---|---|
| groupids | ID/array | IDs групп узлов сети, которые нужно обновить.<br><br>Поведение параметра:<br>*- required* |
| hostids | ID/array | IDs узлов сети, которые нужно удалить из всех групп узлов сети. |

Возвращаемые значения

`(    )` Возвращает объект, который содержит ID обновленных групп узлов сети, указанных в свойстве `groupids`.

Примеры

Удаление узлов сети из групп узлов сети

Удалите два узла сети из указанных групп узлов сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
        "hostids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Источник

CHostGroup::massRemove() в *ui/include/classes/api/services/CHostGroup.php*.

**hostgroup.massupdate**

Описание

`object hostgroup.massupdate(          )`

Этот метод позволяет заменить узлы сети и шаблоны на указанные сразу в нескольких группах узлов сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs групп узлов сети, которые нужно обновить, и объекты, которые следует обновить.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы узлов сети, которые нужно обновить.<br><br>Группы узлов сети должны содержать только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* |

| Parameter | Type | Description |
|---|---|---|
| hosts | object/array | Узлы сети, которые заменят текущие узлы сети в указанных группах узлов сети.<br>Все остальные узлы сети, кроме указанных, будут исключены из групп узлов сети.<br>Обнаруженные узлы сети не будут затронуты.<br><br>Узлы сети должны содержать только свойство `hostid`.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных групп узлов сети, указанных в свойстве `groupids`.

Примеры

Замена узлов сети в группе узлов сети

Замените все узлы сети в группе узлов сети на указанный узел сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massupdate",
    "params": {
        "groups": [
            {
                "groupid": "6"
            }
        ],
        "hosts": [
            {
                "hostid": "30050"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "6",
        ]
    },
    "id": 1
}
```

См. также

- hostgroup.update
- hostgroup.massadd
- Host

Источник

CHostGroup::massUpdate() в *ui/include/classes/api/services/CHostGroup.php*.

**hostgroup.propagate**

Описание

`object hostgroup.propagate( )`

Этот метод позволяет применять разрешения и фильтры тегов ко всем подгруппам группы хостов.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

|Parameter|Type|Description|

|--|--|------|

|groups|object/array|Группы узлов сети, которые нужно распространить.<br><br>Группы узлов сети должны иметь определенное свойство `groupid`.<br><br>Поведение параметра:<br>- *required*|

|permissions|boolean|Установите значение "true", чтобы распространить права доступа.<br><br>Поведение параметра:<br>- *required* если `tag_filters` не задан|

|tag_filters|boolean|Установите значение "true", чтобы распространить фильтры тегов.<br><br>Поведение параметра:<br>- *required* если `permissions` не задан|

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы распространенных групп узлов сети в свойстве `groupids`.

Примеры

Распространение прав доступа к группе узлов сети и фильтров тегов на ее подгруппы.

Распространить права доступа к группе узлов сети и фильтры тегов на ее подгруппы.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "6"
            }
        ],
        "permissions": true,
        "tag_filters": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "6",
        ]
    },
    "id": 1
}
```

См. также

- hostgroup.update
- hostgroup.massadd

- Узел сети

Исходный код

CHostGroup::propagate() в

*ui/include/classes/api/services/CHostGroup.php*.

**hostgroup.update**

Описание

`object hostgroup.update( / hostGroups)`

Этот метод позволяет обновлять существующие группы узлов сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства группы узлов сети, которые необходимо обновить.

Свойство `groupid` должно быть определено для каждой группы узлов сети, все остальные свойства необязательны. Будут обновлены только указанные свойства, все остальные останутся без изменений.

Возвращаемые значения

`( )` Возвращает объект, который содержит ID обновленных групп узлов сети, указанных в свойстве `groupids`.

Примеры

Переименование группы узлов сети

Переименуйте группу узлов сети в ”Linux hosts”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.update",
    "params": {
        "groupid": "7",
        "name": "Linux hosts"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Источник

CHostGroup::update() в *ui/include/classes/api/services/CHostGroup.php*.

**Группа шаблонов**

Этот класс предназначен для работы с группами шаблонов.

Ссылки на объекты:

- Группа шаблонов

Доступные методы:

- templategroup.create - создать новые группы шаблонов
- templategroup.delete - удалить группы шаблонов
- templategroup.get - получить группы шаблонов
- templategroup.massadd - добавить связанные объекты в группы шаблонов
- templategroup.massremove - удалить связанные объекты из групп шаблонов
- templategroup.massupdate - заменить или удалить связанные объекты из групп шаблонов
- templategroup.propagate - распространить права доступа на подгруппы групп шаблонов
- templategroup.update - обновить группы шаблонов

#### Объект группы шаблонов

Следующие объекты напрямую связаны с API `templategroup`.

Группа шаблонов

Объект группы шаблонов имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| groupid | ID | ID группы шаблонов.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| name | string | Имя группы шаблонов.<br><br>Property behavior:<br>- *required* for create operations |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных групп шаблонов с уже существующими. Автоматически генерируется, если не указан. |

#### templategroup.create

Описание

`object templategroup.create(  /  templateGroups)`

Этот метод позволяет создавать новые группы шаблонов.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Группы шаблонов для создания. Метод принимает группы шаблонов со стандартными свойствами группы шаблонов.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных групп шаблонов в свойстве `groupids`. Порядок возвращаемых идентификаторов соответствует порядку переданных групп шаблонов.

Примеры

Создание группы шаблонов

Создайте группу шаблонов с именем "Templates/Databases".

Request:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.create",
    "params": {
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107820"
        ]
    },
    "id": 1
}
```

Исходный код

CTemplateGroup::create() в *ui/include/classes/api/services/CTemplateGroup.php*.

**templategroup.delete**

Описание

`object templategroup.delete(array templateGroupIds)`

Этот метод позволяет удалять группы шаблонов.

Группу шаблонов нельзя удалить, если она содержит шаблоны, которые принадлежат только этой группе.

> **Note:**
> Этот метод доступен только для типов пользователей *Admin* и *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительные сведения см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы групп шаблонов, которые необходимо удалить.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленного шаблона. группы под свойством `groupids`.

Примеры

Удаление нескольких групп шаблонов

Удаление двух групп шаблонов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.delete",
    "params": [
        "107814",
        "107815"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "groupids": [
            "107814",
            "107815"
        ]
    },
    "id": 1
}
```

Источник

CTemplateGroup::delete() в *ui/include/classes/api/services/CTemplateGroup.php*.

### templategroup.get

Описание

`integer/array templategroup.get(          )`

Метод позволяет получить группы шаблонов в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| graphids | ID/array | Возвращать только группы шаблонов, которые содержат шаблоны с указанными графиками. |
| groupids | ID/array | Возвращать только группы шаблонов с указанными идентификаторами групп шаблонов. |
| templateids | ID/array | Возвращать только группы шаблонов, которые содержат указанные шаблоны. |
| triggerids | ID/array | Возвращать только группы шаблонов, которые содержат шаблоны с указанными триггерами. |
| with_graphs | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с графиками.<br><br>Переопределяет параметры `with_simple_graph_items`. |
| with_graph_prototypes | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с прототипами графиков. |
| with_httptests | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с веб-проверками. |
| with_items | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с элементами данных.<br><br>Переопределяет параметр `with_simple_graph_items`. |
| with_item_prototypes | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с прототипами элементов данных.<br><br>Переопределяет параметр `with_simple_graph_item_prototypes`. |
| with_simple_graph_item_prototypes | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с прототипами элементов данных, включенными для создания и имеющими числовой тип информации. |
| with_simple_graph_items | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с числовыми элементами данных. |
| with_templates | boolean | Возвращать только группы шаблонов, которые содержат шаблоны. |

| Parameter | Type | Description |
|---|---|---|
| with_triggers | boolean | Возвращать только группы шаблонов, которые содержат шаблоны с триггерами. |
| selectTemplates | query | Возвращать свойство `templates` с шаблонами, которые принадлежат группе шаблонов.<br><br>Поддерживает `count`. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам:<br>`selectTemplates` - результаты будут отсортированы по `template`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `groupid`, `name`. |
| countOutput | boolean | Эти параметры описаны в комментарии к справочнику. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение данных по имени

Получить все данные о двух группах шаблонов с именами ”Templates/Databases” и ”Templates/Modules”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Templates/Databases",
                "Templates/Modules"
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "13",
            "name": "Templates/Databases",
```

```
            "uuid": "748ad4d098d447d492bb935c907f652f"
        },
        {
            "groupid": "8",
            "name": "Templates/Modules",
            "uuid": "57b7ae836ca64446ba2c296389c009b7"
        }
    ],
    "id": 1
}
```


См. также

- Шаблон

Источник

CTemplateGroup::get() в *ui/include/classes/api/services/CTemplateGroup.php*.

### templategroup.massadd

Описание

`object templategroup.massadd(          )`

Этот метод позволяет одновременно добавить несколько связанных объектов во все указанные группы шаблонов.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs групп шаблонов для обновления и объекты, которые нужно добавить во все группы шаблонов.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы шаблонов, которые нужно обновить.<br><br>Группы шаблонов должны иметь определенным только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* |
| templates | object/array | Шаблоны, которые нужно добавить во все группы шаблонов.<br><br>У шаблонов должно быть определено только свойство `templateid`.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных групп шаблонов в свойстве `groupids`.

Примеры

Добавление шаблонов в группы шаблонов

Добавьте два шаблона в группы шаблонов с ID 12 и 13.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "12"
            },
            {
                "groupid": "13"
            }
        ],
        "templates": [
            {
                "templateid": "10486"
            },
            {
                "templateid": "10487"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "12",
            "13"
        ]
    },
    "id": 1
}
```

См. также

- Шаблон

Источник

CTemplateGroup::massAdd() в *ui/include/classes/api/services/CTemplateGroup.php*.

**templategroup.massremove**

Описание

`object templategroup.massremove(          )`

Этот метод позволяет удалить связанные объекты из нескольких групп шаблонов.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие ID групп шаблонов, которые нужно обновить, и объекты, которые следует удалить.

| Parameter | Type | Description |
|---|---|---|
| groupids | ID/array | ID групп шаблонов, которые нужно обновить.<br><br>Поведение параметра:<br>- *required* |
| templateids | ID/array | ID шаблонов, которые нужно удалить из всех групп шаблонов.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных групп шаблонов в свойстве `groupids`.

Примеры

Удаление шаблонов из групп шаблонов

Удалите два шаблона из указанных групп шаблонов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
        "templateids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Источник

CTemplateGroup::massRemove() в *ui/include/classes/api/services/CTemplateGroup.php*.

**templategroup.massupdate**

Описание

`object templategroup.massupdate(          )`

Этот метод позволяет заменить шаблоны на указанные в нескольких группах шаблонов.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs групп шаблонов, которые нужно обновить, и объекты, которые следует обновить.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы шаблонов, которые нужно обновить.<br><br>Группы шаблонов должны иметь определенным только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* |
| templates | object/array | Шаблоны, которые заменят текущий шаблон в указанных группах шаблонов.<br>Все остальные шаблоны, кроме упомянутых, будут исключены из групп шаблонов.<br><br>Шаблоны должны иметь определенным только свойство `templateid`.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных групп шаблонов в свойстве `groupids`.

Примеры

Замена шаблонов в группе шаблонов

Замените все шаблоны в группе шаблонов на указанные шаблоны.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massupdate",
    "params": {
        "groups": [
            {
                "groupid": "8"
            }
        ],
        "templates": [
            {
                "templateid": "40050"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "8",
```

```
        ]
    },
    "id": 1
}
```

См. также

- templategroup.update
- templategroup.massadd
- Шаблон

Источник

CTemplateGroup::massUpdate() в *ui/include/classes/api/services/CTemplateGroup.php*.

**templategroup.propagate**

Описание

`object templategroup.propagate(          )`

Этот метод позволяет применить права ко всем подгруппам групп шаблонов.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

|Parameter|Type|Description|

|--|--|------|

|groups|object/array|Группы шаблонов для распространения.<br><br>Группы шаблонов должны содержать только свойство `groupid`.<br><br>Поведение параметра:<br>- *required*|

|permissions|boolean|Установите `true`, если требуется распространить права доступа.<br><br>Поведение параметра:<br>- *required*|

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы распространенных групп шаблонов в свойстве `groupids`.

Примеры

Распространение прав доступа группы шаблонов на ее подгруппы.

Распространить права доступа группы шаблонов на ее подгруппы.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "15"
```

```
            }
        ],
        "permissions": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "15",
        ]
    },
    "id": 1
}
```

См. также

- templategroup.update
- templategroup.massadd
- Шаблон

Источник

CTemplateGroup::propagate() в

*ui/include/classes/api/services/CTemplateGroup.php*.

**templategroup.update**

Описание

`object templategroup.update( / templateGroups)`

Этот метод позволяет обновлять существующие группы шаблонов.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства группы шаблонов, которые необходимо обновить.

Свойство `groupid` должно быть определено для каждой группы шаблонов, все остальные свойства необязательны. Будут обновлены только указанные свойства, все остальные останутся без изменений.

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы обновленных групп шаблонов в свойстве groupids.

Примеры

Переименование группы шаблонов

Переименуйте группу шаблонов в "Templates/Databases"

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.update",
    "params": {
        "groupid": "7",
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Источник

CTemplateGroup::update() в *ui/include/classes/api/services/CTemplateGroup.php*.

### Журнал аудита

Этот класс предназначен для работы с журналом аудита.

Ссылки на объекты:

- Журнал аудита

Доступные методы:

- auditlog.get - получить записи журнала аудита

### Объект журнала аудита

Следующие объекты напрямую связаны с API auditlog.

Журнал аудита

Объект журнала аудита содержит информацию о действиях пользователей. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| auditid | ID | ID записи журнала аудита. Сгенерирован с использованием алгоритма CUID. |
| userid | ID | ID пользователя-автора записи журнала аудита. |
| username | string | Имя пользователя-автора записи журнала аудита. |
| clock | timestamp | Метка времени создания записи журнала аудита. |
| ip | string | IP-адрес пользователя-автора записи журнала аудита. |

| Property | Type | Description |
|---|---|---|
| action | integer | Действие записи журнала аудита.<br><br>Возможные значения:<br>0 - Add;<br>1 - Update;<br>2 - Delete;<br>4 - Logout;<br>7 - Execute;<br>8 - Login;<br>9 - Failed login;<br>10 - History clear;<br>11 - Config refresh;<br>12 - Push. |

| Property | Type | Description |
|---|---|---|
| resourcetype | integer | Тип ресурса записи журнала аудита.<br><br>Возможные значения:<br>0 - User;<br>3 - Media type;<br>4 - Host;<br>5 - Action;<br>6 - Graph;<br>11 - User group;<br>13 - Trigger;<br>14 - Host group;<br>15 - Item;<br>16 - Image;<br>17 - Value map;<br>18 - Service;<br>19 - Map;<br>22 - Web scenario;<br>23 - Discovery rule;<br>25 - Script;<br>26 - Proxy;<br>27 - Maintenance;<br>28 - Regular expression;<br>29 - Macro;<br>30 - Template;<br>31 - Trigger prototype;<br>32 - Icon mapping;<br>33 - Dashboard;<br>34 - Event correlation;<br>35 - Graph prototype;<br>36 - Item prototype;<br>37 - Host prototype;<br>38 - Autoregistration;<br>39 - Module;<br>40 - Settings;<br>41 - Housekeeping;<br>42 - Authentication;<br>43 - Template dashboard;<br>44 - User role;<br>45 - API token;<br>46 - Scheduled report;<br>47 - High availability node;<br>48 - SLA;<br>49 - User directory;<br>50 - Template group;<br>51 - Connector;<br>52 - LLD rule;<br>53 - History;<br>54 - Multi-factor authentication;<br>55 - Proxy group;<br>56 - LLD rule prototype. |
| resourceid | ID | Идентификатор ресурса записи журнала аудита. |
| resource_cuid | ID | Уникальный идентификатор ресурса записи журнала аудита, сгенерированный с использованием алгоритма CUID. |
| resourcename | string | Читаемое имя ресурса записи журнала аудита. |
| recordsetid | ID | ID набора записей журнала аудита. Записи журнала аудита, созданные в ходе одной и той же операции, будут иметь одинаковый ID набора записей. Сгенерирован с использованием алгоритма CUID. |

| Property | Type | Description |
|---|---|---|
| details | text | Подробности записи журнала аудита. Подробности хранятся в виде объекта JSON, где имя каждого свойства является путем к свойству или вложенному объекту, в котором произошло изменение, а каждое значение содержит данные об изменении этого свойства или вложенного объекта в виде массива.<br><br>Возможные форматы значений:<br>["add"] - Вложенный объект был добавлен;<br>["add", "<value>"] - Свойство добавленного объекта равно <value>;<br>["update"] - Вложенный объект был обновлен;<br>["update", "<new value>", "<old value>"] - Свойство обновленного объекта было изменено с <old value> на <new value>;<br>["delete"] - Вложенный объект был удален. |

**auditlog.get**

Описание

`integer/array auditlog.get(object parameters)`

Метод позволяет получать записи журнала аудита в соответствии с заданными параметрами.

**Note:**
Этот метод доступен только пользователям типа *Super admin*. Разрешение на вызов метода можно отозвать в настройках роли пользователя. Дополнительные сведения см. в разделе User roles.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| auditids | ID/array | Возвращать только журнал аудита с указанными ID. |
| userids | ID/array | Возвращать только журнал аудита, созданный указанными пользователями. |
| time_from | timestamp | Возвращать только записи журнала аудита, созданные после указанного времени или в указанное время. |
| time_till | timestamp | Возвращать только записи журнала аудита, созданные до указанного времени или в указанное время. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `auditid`, `userid`, `clock`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;

- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение журнала аудита

Получить два последних записи журнала аудита.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "auditlog.get",
    "params": {
        "output": "extend",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 2
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Смотрите также

- ·Audit log object

Источник

CAuditLog::get() в *ui/include/classes/api/services/CAuditLog.php*.

### Задача

Этот класс предназначен для работы с задачами (например, для проверки элементов данных или правил низкоуровневого обнаружения без перезагрузки конфигурации).

Ссылки на объекты:

- Task
  - объект запроса 'Execute now'
  - объект запроса 'Refresh proxy configuration'
  - объект запроса 'Diagnostic information'
    - объект запроса Statistic
    - объект результата Statistic

Доступные методы:

- task.create - создать новые задачи
- task.get - получить задачи

### Объект task

Следующие объекты напрямую связаны с API `task`.

Task

Объект task имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| taskid | ID | ID задачи.<br><br>Поведение свойства:<br>- *только для чтения* |
| type | integer | Тип задачи.<br><br>Возможные значения:<br>1 - диагностическая информация;<br>2 - обновление конфигурации прокси;<br>6 - выполнить сейчас.<br><br>Начиная с Zabbix 7.4.3, пользователи типов *Admin* и *User* могут создавать задачи типа "Execute now".<br><br>Обратите внимание, что `task.get` всегда возвращает "7" (сводка выполнения задачи).<br><br>Поведение свойства:<br>- *обязательно* |
| status | integer | Статус задачи.<br><br>Возможные значения:<br>1 - новая задача;<br>2 - задача в процессе выполнения;<br>3 - задача завершена;<br>4 - срок действия задачи истек.<br><br>Поведение свойства:<br>- *только для чтения* |

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время создания задачи.<br><br>Поведение свойства:<br>- *только для чтения* |
| ttl | integer | Время в секундах, по истечении которого задача становится просроченной.<br><br>Поведение свойства:<br>- *только для чтения* |
| proxyid | ID | ID прокси, по которому собирается статистика диагностической информации.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлено в ”Diagnostic information” или ”Refresh proxy configuration” |
| request | object | Объект запроса задачи в зависимости от типа задачи:<br>Объект задачи ’Execute now’ подробно описан ниже;<br>Объект задачи ’Refresh proxy configuration’ подробно описан ниже;<br>Объект задачи ’Diagnostic information’ подробно описан ниже.<br><br>Поведение свойства:<br>- *обязательно* |
| result | object | Объект результата задачи диагностической информации. Может содержать NULL, если результат еще не готов.<br>Объект результата подробно описан ниже.<br><br>Поведение свойства:<br>- *только для чтения* |

Объект запроса ’Execute now’

Объект запроса задачи ’Execute now’ имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| itemid | ID | ID элемента данных и правил низкоуровневого обнаружения.<br><br>Начиная с Zabbix 7.4.3, пользователи типов *Admin* и *User* могут выполнять ’Execute now’ для элементов данных на узлах сети, для которых у них есть разрешение *read-write* или разрешение *read* и для их роли включено действие `invoke_execute_now`. То же самое относится к пользователям типа *Admin* для правил низкоуровневого обнаружения (LLD). |

Объект запроса ’Refresh proxy configuration’

Объект запроса задачи ’Refresh proxy configuration’ имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| proxyids | array | Идентификаторы прокси. |

Объект запроса 'Diagnostic information'

Объект запроса задачи диагностической информации имеет следующие свойства. Объект запроса статистики для всех типов свойств подробно описан ниже.

| Property | Type | Description |
|---|---|---|
| historycache | object | Запрос статистики кэша истории. Доступно на сервере и прокси. |
| valuecache | object | Запрос статистики кэша элементов данных. Доступно на сервере. |
| preprocessing | object | Запрос статистики менеджера предварительной обработки. Доступно на сервере и прокси. |
| alerting | object | Запрос статистики менеджера оповещений. Доступно на сервере. |
| lld | object | Запрос статистики менеджера LLD. Доступно на сервере. |

Объект запроса статистики

Объект запроса статистики используется для определения того, какой тип информации следует собирать о внутренних процессах сервера/прокси. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| stats | query | Свойства объекта статистики, которые нужно вернуть.<br>Список доступных полей для каждого типа статистики диагностической информации подробно описан ниже.<br><br>По умолчанию: `extend` вернет все доступные поля статистики. |
| top | object | Объект для сортировки и ограничения возвращаемых значений статистики.<br>Список доступных полей для каждого типа статистики диагностической информации подробно описан ниже.<br><br>Пример: { "source.alerts": 10 } |

Список полей статистики, доступных для каждого типа запроса диагностической информации

Для каждого свойства запроса диагностической информации можно запросить следующие поля статистики.

| Diagnostic type | Available fields | Description |
|---|---|---|
| historycache | items | Количество кэшированных элементов данных. |
| | values | Количество кэшированных значений. |
| | memory | Статистика общей памяти (свободное пространство, количество используемых блоков, количество свободных блоков, максимальный размер свободного блока). |
| | memory.data | Статистика общей памяти кэша исторических данных. |
| | memory.index | Статистика общей памяти кэша индекса истории. |
| valuecache | items | Количество кэшированных элементов данных. |
| | values | Количество кэшированных значений. |
| | memory | Статистика общей памяти (свободное пространство, количество используемых блоков, количество свободных блоков, максимальный размер свободного блока). |
| | mode | Режим кэша значений. |
| preprocessing | values | Количество значений в очереди. |
| | preproc.values | Количество значений в очереди со шагами предварительной обработки. |
| alerting | alerts | Количество оповещений в очереди. |
| lld | rules | Количество правил в очереди. |
| | values | Количество значений в очереди. |

Список полей сортировки, доступных для каждого типа запроса диагностической информации

Следующие статистические поля можно использовать для сортировки и ограничения запрашиваемой информации.

| Diagnostic type | Available fields | Type |
|---|---|---|
| historycache | values | integer |
| valuecache | values | integer |
| | request.values | integer |
| preprocessing | values | integer |
| alerting | media.alerts | integer |
| | source.alerts | integer |
| lld | values | integer |

Объект результата статистики

Объект результата статистики извлекается в поле `result` объекта задачи.

| Property | Type | Description |
|---|---|---|
| status | integer | Статус результата задачи.<br><br>Возможные значения:<br>-1 - при выполнении задачи произошла ошибка;<br>0 - результат задачи создан.<br><br>Поведение свойства:<br>*- только для чтения* |
| data | string/object | Результаты в соответствии с объектом запроса статистики для конкретной задачи диагностической информации.<br>Содержит строку сообщения об ошибке, если при выполнении задачи произошла ошибка. |

**task.create**

Описание

`object task.create(object/array tasks)`

Этот метод позволяет создавать задачи.

**Note:**
Этот метод доступен пользователям любого типа (пользователям типов *Admin* и *User* начиная с Zabbix 7.4.3). Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Задача для создания.

Метод принимает задачи со стандартными свойствами задачи.

Обратите внимание, что задачи типа ”Выполнить сейчас” можно создавать только для следующих типов элементов данных/правил обнаружения:

- Zabbix agent (passive)
- Simple check
- SNMP agent (v1/v2/v3)
- Zabbix internal
- External check
- Database monitor
- HTTP agent
- IPMI agent
- SSH agent
- TELNET agent

- JMX agent
- Calculated
- Dependent item
- Script
- Browser

Если элемент данных/правило обнаружения имеет тип ”Dependent item”, то его основной элемент данных также должен относиться к одному из указанных выше типов.

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных задач, указанных в свойстве `taskids`. Одна задача создается по каждому элементу данных и правилу низкоуровневого обнаружения. Порядок возвращаемых ID совпадает с порядком переданных `itemids`.

Примеры

Создание задачи

Создайте задачу «Выполнить сейчас» для элемента данных и правила низкоуровневого обнаружения.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 6,
            "request": {
                "itemid": "10092"
            }
        },
        {
            "type": 6,
            "request": {
                "itemid": "10093"
            }
        }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Создайте задачу «Обновить конфигурацию прокси» для двух прокси.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 2,
            "request": {
                "proxyids": ["10459", "10460"]
            }
```

```
        }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1"
        ]
    },
    "id": 1
}
```

Создайте задачу «Диагностическая информация».

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "3"
        ]
    },
    "id": 1
}
```

См. также

- Задача
- Объект запроса 'Execute now'
- Объект запроса 'Refresh proxy configuration'

- Объект запроса 'Diagnostic information'

Источник

CTask::create() в *ui/include/classes/api/services/CTask.php*.

**task.get**

Описание

`integer/array task.get(object parameters)`

Метод позволяет получать задачи в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен только для типа пользователя *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. См. Роли пользователей для получения дополнительной информации.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| taskids | ID/array | Возвращать только задачи с указанными ID. |
| output | query | Эти параметры описаны в справочном комментарии. |
| preservekeys | boolean | |

Возвращаемые значения

( / ) Возвращает массив объектов.

Примеры

Получение задачи по ID

Получите все данные о задаче с ID "1".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "task.get",
    "params": {
        "output": "extend",
        "taskids": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


См. также

- Задача
- Объект запроса 'Execute now'
- Объект запроса 'Refresh proxy configuration'
- Объект запроса 'Diagnostic information'

Источник

CTask::get() в *ui/include/classes/api/services/CTask.php*.

#### Изображение

Этот класс предназначен для работы с изображениями.

Ссылки на объекты:

- Image

Доступные методы:

- image.create - создать новые изображения
- image.delete - удалить изображения
- image.get - получить изображения
- image.update - обновить изображения

#### Объект image

Следующие объекты напрямую связаны с API image.

Изображение

Объект изображения имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| imageid | ID | ID изображения.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя изображения.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| imagetype | integer | Тип изображения.<br><br>Возможные значения:<br>1 - *(по умолчанию)* значок;<br>2 - фоновое изображение.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |
| image | string | Изображение, закодированное в Base64.<br>Максимальный размер закодированного изображения — 1 МБ.<br>Максимальный размер можно изменить, изменив значение константы ZBX_MAX_IMAGE_SIZE.<br>Поддерживаемые форматы изображений: PNG, JPEG, GIF и WebP.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |

**image.create**

Описание

`object image.create( / images)`

Этот метод позволяет создавать новые изображения.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Изображения для создания.

Метод принимает изображения со стандартными свойствами изображения.

Возвращаемые значения

`( )` Возвращает объект, который содержит ID созданных изображений, указанных в свойстве imageids. Порядок возвращаемых ID совпадает с порядком переданных изображений.

Примеры

Создание изображения значка

Создайте значок облака.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 1,
        "name": "Cloud_(24)",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAAApgB
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "188"
        ]
    },
    "id": 1
}
```

Создание фонового изображения

Создайте небольшое фоновое изображение, которое будет использоваться в качестве шаблона.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 2,
        "name": "background_image",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAAYCAIAAABvFaqvAAABhWlDQ1BJQ0MgcHJvZmlsZQAAKJF9kb1Lw0AYxp+m
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "189"
        ]
    },
    "id": 1
}
```

Источник

CImage::create() в *ui/include/classes/api/services/CImage.php*.

**image.delete**

Описание

`object image.delete(    imageIds)`

Этот метод позволяет удалять изображения.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID удаляемых изображений.

Возвращаемые значения

( ) Возвращает объект, который содержит ID удаленных изображений, указанных в свойстве `imageids`.

Примеры

Удаление нескольких изображений

Удаление двух изображений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "image.delete",
    "params": [
        "188",
        "192"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "188",
            "192"
        ]
    },
    "id": 1
}
```

Источник

CImage::delete() в *ui/include/classes/api/services/CImage.php*.

**image.get**

Описание

`integer/array image.get( )`

Этот метод позволяет получать изображения в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| imageids | ID/array | Возвращать только изображения с указанными ID. |

| Parameter | Type | Description |
|---|---|---|
| sysmapids | ID/array | Возвращать изображения, которые используются на указанных картах. |
| select_image | flag | Возвращать свойство image с изображением, закодированным в Base64. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: imageid, name. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр countOutput.

Примеры

Получение изображения

Получить все данные для изображения с ID "2".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "image.get",
    "params": {
        "output": "extend",
        "select_image": true,
        "imageids": "2"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "imageid": "2",
            "imagetype": "1",
            "name": "Cloud_(24)",
            "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAA
        }
    ],
    "id": 1
}
```

Источник

CImage::get() в *ui/include/classes/api/services/CImage.php*.

### image.update

Описание

`object image.update( / images)`

Этот метод позволяет обновлять существующие изображения.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства изображения, которые необходимо обновить.

Для каждого изображения должно быть определено свойство `imageid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Метод принимает изображения со стандартными свойствами изображения.

Возвращаемые значения

`( )` Возвращает объект, который содержит ID обновленных изображений, указанных в свойстве `imageids`.

Примеры

Переименовать изображение

Переименуйте изображение в "Cloud icon".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "image.update",
    "params": {
        "imageid": "2",
        "name": "Cloud icon"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "2"
        ]
    },
    "id": 1
}
```

Источник

CImage::update() в *ui/include/classes/api/services/CImage.php*.

### Интерфейс узла сети

Этот класс предназначен для работы с интерфейсами узлов сети.

Ссылки на объекты:

- Интерфейс узла сети
  - Подробности

Доступные методы:

- hostinterface.create - создать новые интерфейсы узлов сети
- hostinterface.delete - удалить интерфейсы узлов сети
- hostinterface.get - получить интерфейсы узлов сети
- hostinterface.massadd - добавить интерфейсы узлов сети к узлам сети
- hostinterface.massremove - удалить интерфейсы узлов сети с узлов сети
- hostinterface.replacehostinterfaces - заменить интерфейсы узла сети
- hostinterface.update - обновить интерфейсы узлов сети

**Объект интерфейса узла сети**

Следующие объекты напрямую связаны с API `hostinterface`.

Интерфейс узла сети

Объект интерфейса узла сети имеет следующие свойства.

**Attention:**
Обратите внимание, что свойства `ip` и `dns` являются *обязательными* для операций создания. Если вы не хотите использовать DNS, задайте для него пустую строку.

| Property | Type | Description |
|---|---|---|
| interfaceid | ID | ID интерфейса.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательное* для операций обновления |
| available | integer | Доступность интерфейса узла сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* неизвестно;<br>1 - доступен;<br>2 - недоступен.<br><br>Поведение свойства:<br>- *только для чтения* |
| hostid | ID | ID узла сети, которому принадлежит интерфейс.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательное* для операций создания |
| type | integer | Тип интерфейса.<br><br>Возможные значения:<br>1 - агент;<br>2 - SNMP;<br>3 - IPMI;<br>4 - JMX.<br><br>Поведение свойства:<br>- *обязательное* для операций создания |
| ip | string | IP-адрес, используемый интерфейсом.<br><br>Может быть пустым, если соединение выполняется через DNS.<br><br>Поведение свойства:<br>- *обязательное* для операций создания |
| dns | string | DNS-имя, используемое интерфейсом.<br><br>Может быть пустым, если соединение выполняется через IP.<br><br>Поведение свойства:<br>- *обязательное* для операций создания |

| Property | Type | Description |
|---|---|---|
| port | string | Номер порта, используемый интерфейсом.<br>Может содержать пользовательские макросы.<br><br>Поведение свойства:<br>- *обязательное* для операций создания |
| useip | integer | Следует ли выполнять соединение через IP.<br><br>Возможные значения:<br>0 - подключаться, используя DNS-имя узла сети;<br>1 - подключаться, используя IP-адрес узла сети.<br><br>Поведение свойства:<br>- *обязательное* для операций создания |
| main | integer | Используется ли интерфейс как интерфейс по умолчанию на узле сети. Только один интерфейс некоторого типа может быть задан как интерфейс по умолчанию на узле сети.<br><br>Возможные значения:<br>0 - не по умолчанию;<br>1 - по умолчанию.<br><br>Поведение свойства:<br>- *обязательное* для операций создания |
| details | object | Дополнительный объект details для интерфейса.<br><br>Поведение свойства:<br>- *обязательное*, если `type` установлено в ”SNMP” |
| disable_until | timestamp | Время следующего опроса недоступного интерфейса узла сети.<br><br>Поведение свойства:<br>- *только для чтения* |
| error | string | Текст ошибки, если интерфейс узла сети недоступен.<br><br>Поведение свойства:<br>- *только для чтения* |
| errors_from | timestamp | Время, когда интерфейс узла сети стал недоступен.<br><br>Поведение свойства:<br>- *только для чтения* |

Подробности

Объект details имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| version | integer | Версия интерфейса SNMP.<br><br>Возможные значения:<br>1 - SNMPv1;<br>2 - SNMPv2c;<br>3 - SNMPv3.<br><br>Поведение свойства:<br>- *required* |
| bulk | integer | Использовать ли массовые запросы SNMP.<br><br>Возможные значения:<br>0 - не использовать массовые запросы;<br>1 - (по умолчанию) - использовать массовые запросы. |

| Property | Type | Description |
|---|---|---|
| community | string | Сообщество SNMP. Используется только интерфейсами SNMPv1 и SNMPv2.<br><br>Поведение свойства:<br>- *required* если `version` установлено в ”SNMPv1” или ”SNMPv2c” |
| max_repetitions | integer | Максимальное значение повторений для нативных массовых запросов SNMP (PDU GetBulkRequest).<br>Используется только для элементов данных `discovery[]` и `walk[]` в SNMPv2 и v3.<br><br>По умолчанию: 10. |
| securityname | string | Имя безопасности SNMPv3. Используется только интерфейсами SNMPv3. |
| securitylevel | integer | Уровень безопасности SNMPv3. Используется только интерфейсами SNMPv3.<br><br>Возможные значения:<br>0 - (по умолчанию) - noAuthNoPriv;<br>1 - authNoPriv;<br>2 - authPriv. |
| authpassphrase | string | Фраза-пароль аутентификации SNMPv3. Используется только интерфейсами SNMPv3. |
| privpassphrase | string | Фраза-пароль конфиденциальности SNMPv3. Используется только интерфейсами SNMPv3. |
| authprotocol | integer | Протокол аутентификации SNMPv3. Используется только интерфейсами SNMPv3.<br><br>Возможные значения:<br>0 - (по умолчанию) - MD5;<br>1 - SHA1;<br>2 - SHA224;<br>3 - SHA256;<br>4 - SHA384;<br>5 - SHA512. |
| privprotocol | integer | Протокол конфиденциальности SNMPv3. Используется только интерфейсами SNMPv3.<br><br>Возможные значения:<br>0 - (по умолчанию) - DES;<br>1 - AES128;<br>2 - AES192;<br>3 - AES256;<br>4 - AES192C;<br>5 - AES256C. |
| contextname | string | Имя контекста SNMPv3. Используется только интерфейсами SNMPv3. |

**hostinterface.create**

Описание

`object hostinterface.create(  /  hostInterfaces)`

Этот метод позволяет создавать новые интерфейсы узлов сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Интерфейсы узла сети для создания.

Метод принимает интерфейсы узла сети со стандартными свойствами интерфейса узла сети.

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных интерфейсов узлов сети, указанных в свойстве interfaceids. Порядок возвращаемых ID совпадает с порядком переданных интерфейсов узлов сети.

Примеры

Создание нового интерфейса

Создайте дополнительный IP-интерфейс агента на узле сети "30052."

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.create",
    "params": {
        "hostid": "30052",
        "main": "0",
        "type": "1",
        "useip": "1",
        "ip": "127.0.0.1",
        "dns": "",
        "port": "10050"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30062"
        ]
    },
    "id": 1
}
```

Создание интерфейса с данными SNMP

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.create",
    "params": {
        "hostid": "10456",
        "main": "0",
        "type": "2",
        "useip": "1",
        "ip": "127.0.0.1",
        "dns": "",
        "port": "1601",
        "details": {
            "version": "2",
            "bulk": "1",
            "community": "{$SNMP_COMMUNITY}"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30063"
        ]
    },
    "id": 1
}
```

Смотрите также

- ·hostinterface.massadd
- ·host.massadd

Источник

CHostInterface::create() в *ui/include/classes/api/services/CHostInterface.php*.

**hostinterface.delete**

Описание

`object hostinterface.delete(    hostInterfaceIds)`

Этот метод позволяет удалять интерфейсы узлов сети.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых интерфейсов узлов сети.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных интерфейсов узлов сети, указанных в свойстве `interfaceids`.

Примеры

Удаление интерфейса узла сети

Удалите интерфейс узла сети с ID 30062.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.delete",
    "params": [
        "30062"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30062"
        ]
    },
    "id": 1
}
```

Смотрите также

- hostinterface.massremove
- host.massremove

Источник

CHostInterface::delete() в *ui/include/classes/api/services/CHostInterface.php*.

**hostinterface.get**

Описание

`integer/array hostinterface.get(parameters)`

Этот метод позволяет получать интерфейсы узлов сети в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hostids | ID/array | Возвращать только интерфейсы узлов сети, используемые указанными узлами сети. |
| interfaceids | ID/array | Возвращать только интерфейсы узлов сети с указанными ID. |
| itemids | ID/array | Возвращать только интерфейсы узлов сети, используемые указанными элементами данных. |
| triggerids | ID/array | Возвращать только интерфейсы узлов сети, используемые элементами данных в указанных триггерах. |
| selectItems | query | Возвращать свойство `items` с элементами данных, которые используют интерфейс.<br><br>Поддерживает `count`. |
| selectHosts | query | Возвращать свойство `hosts` с массивом узлов сети, которые используют интерфейс. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам: `selectItems`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `interfaceid`, `dns`, `ip`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer / array)` Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение интерфейсов узла сети

Получить все данные об интерфейсах, используемых узлом сети "30057."

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.get",
    "params": {
        "output": "extend",
        "hostids": "30057"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Смотрите также

- Узел сети
- Элемент данных

Источник

CHostInterface::get() в *ui/include/classes/api/services/CHostInterface.php*.

### hostinterface.massadd

Описание

`object hostinterface.massadd(          )`

Этот метод позволяет добавить одновременно несколько интерфейсов узлов сети на узлы сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие интерфейсы узлов сети, которые будут созданы на указанных узлах сети.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| interfaces | object/array | Интерфейсы узлов сети, которые нужно создать на указанных узлах сети.<br><br>Поведение параметра:<br>- *required* |
| hosts | object/array | Узлы сети, которые нужно обновить.<br><br>У узлов сети должно быть определено только свойство `hostid`.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

`(    )` Возвращает объект, который содержит ID созданных интерфейсов узла сети, указанных в свойстве `interfaceids`.

Примеры

Создание интерфейсов

Создайте интерфейс на двух узлах сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.massadd",
    "params": {
        "hosts": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30052"
            }
        ],
        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "main": 0,
            "port": "10050",
            "type": 1,
```

```
            "useip": 1
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30069",
            "30070"
        ]
    },
    "id": 1
}
```

Смотрите также

- hostinterface.create
- host.massadd
- Узел сети

Источник

CHostInterface::massAdd() в *ui/include/classes/api/services/CHostInterface.php*.

**hostinterface.massremove**

Описание

`object hostinterface.massremove(  )`

Этот метод позволяет удалить интерфейсы узлов сети у заданных узлов сети.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие ID узлов сети, которые нужно обновить, и интерфейсы, которые нужно удалить.

| Parameter | Type | Description |
|---|---|---|
| interfaces | object/array | Интерфейсы узла сети, которые нужно удалить у указанных узлов сети.<br><br>Объект интерфейса узла сети должен содержать только свойства `ip`, `dns` и `port`.<br><br>Поведение параметра:<br>- *required* |
| hostids | ID/array | ID узлов сети, которые нужно обновить.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID удаленных интерфейсов узлов сети, указанных в свойстве `interfaceids`.

Примеры

Удаление интерфейсов

Удалите SNMP-интерфейс "127.0.0.1" у двух узлов сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.massremove",
    "params": {
        "hostids": [
            "30050",
            "30052"
        ],
        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "port": "161"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30069",
            "30070"
        ]
    },
    "id": 1
}
```

Смотрите также

- hostinterface.delete
- host.massremove

Источник

CHostInterface::massRemove() в *ui/include/classes/api/services/CHostInterface.php*.

**hostinterface.replacehostinterfaces**

Описание

`object hostinterface.replacehostinterfaces(          )`

Этот метод позволяет заменить все интерфейсы узла сети на заданном узле сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие ID узла сети, который нужно обновить, и новые интерфейсы узла сети.

| Параметр | Тип | Описание |
|---|---|---|
| interfaces | object/array | Интерфейсы узла сети, которыми будут заменены текущие интерфейсы узла сети.<br><br>Поведение параметра:<br>- *required* |
| hostid | ID | ID узла сети, который нужно обновить.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных интерфейсов узла сети, указанных в свойстве `interfaceids`.

Примеры

Замена интерфейсов узла сети

Замените все интерфейсы узла сети одним интерфейсом агента.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.replacehostinterfaces",
    "params": {
        "hostid": "30052",
        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "main": 1,
            "port": "10050",
            "type": 1,
            "useip": 1
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30081"
        ]
    },
    "id": 1
}
```

Смотрите также

- host.update
- host.massupdate

Источник

CHostInterface::replaceHostInterfaces() в *ui/include/classes/api/services/CHostInterface.php*.

**hostinterface.update**

Описание

`object hostinterface.update( / hostInterfaces)`

Этот метод позволяет обновлять существующие интерфейсы узлов сети.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства интерфейса узла сети, которые необходимо обновить.

Свойство `interfaceid` должно быть определено для каждого интерфейса узла сети, все остальные свойства необязательны. Будут обновлены только указанные свойства, все остальные останутся без изменений.

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных интерфейсов узлов сети, указанных в свойстве `interfaceids`.

Примеры

Изменение порта интерфейса узла сети

Измените порт интерфейса узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.update",
    "params": {
        "interfaceid": "30048",
        "port": "30050"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "interfaceids": [
            "30048"
        ]
    },
    "id": 1
}
```

Источник

CHostInterface::update() в *ui/include/classes/api/services/CHostInterface.php*.

**История**

Этот класс предназначен для работы с данными истории.

Ссылки на объекты:

- История float
- История integer
- История string
- История text
- История log

Доступные методы:

- history.clear - очистить данные истории
- history.get - получить данные истории
- history.push - отправить данные истории на сервер Zabbix

### Объект history

Следующие объекты напрямую связаны с API `history`.

**Note:**
Объекты history различаются в зависимости от типа информации элемента данных. Они создаются сервером Zabbix и не могут быть изменены через API.

История float

Объект истории float имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время, когда было получено это значение. |
| itemid | ID | ID связанного элемента данных. |
| ns | integer | Наносекунды, когда было получено значение. |
| value | float | Полученное значение. |

История целых чисел

Объект истории целых чисел имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время, когда было получено это значение. |
| itemid | ID | ID связанного элемента данных. |
| ns | integer | Наносекунды, когда было получено значение. |
| value | integer | Полученное значение. |

История строки

Объект истории строки имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время, когда было получено это значение. |
| itemid | ID | ID связанного элемента данных. |
| ns | integer | Наносекунды, когда было получено значение. |
| value | string | Полученное значение. |

История текста

Объект истории текста имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| id | ID | ID записи истории. |
| clock | timestamp | Время, когда было получено это значение. |
| itemid | ID | ID связанного элемента данных. |
| ns | integer | Наносекунды, когда было получено значение. |
| value | text | Полученное значение. |

История журнала

Объект истории журнала имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| id | ID | ID записи истории. |
| clock | timestamp | Время, когда было получено это значение. |
| itemid | ID | ID связанного элемента данных. |
| logeventid | integer | ID записи журнала событий Windows. |

| Property | Type | Description |
|---|---|---|
| ns | integer | Наносекунды, когда было получено значение. |
| severity | integer | Уровень записи журнала событий Windows. |
| source | string | Источник записи журнала событий Windows. |
| timestamp | timestamp | Время записи журнала событий Windows. |
| value | text | Полученное значение. |

Двоичная история

Объект двоичной истории имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время, когда было получено это значение. |
| itemid | ID | ID связанного элемента данных. |
| ns | integer | Наносекунды, когда было получено значение. |
| value | text | Полученное значение. |

**history.clear**

Описание

`object history.clear(    itemids)`

Этот метод позволяет очистить историю элемента.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` ID элементов для очистки.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы очищенных элементов в свойстве `itemids`.

Примеры

Очистка истории

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "history.clear",
    "params": [
        "10325",
        "13205"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10325",
            "13205"
        ]
    },
    "id": 1
}
```

Источник

CHistory::clear() в *ui/include/classes/api/services/CHistory.php*.

**history.get**

Описание

`integer/array history.get( )`

Метод позволяет получить данные истории по заданным параметрам.

**Attention:**
Этот метод может возвращать исторические данные удаленного объекта, если эти данные еще не были удалены служебной программой.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| history | integer | Типы объектов истории, которые нужно вернуть.<br><br>Возможные значения:<br>0 - числовое число с плавающей точкой;<br>1 - символ;<br>2 - журнал;<br>3 - *(по умолчанию)* числовое беззнаковое;<br>4 - текст;<br>5 - двоичный. |
| hostids | ID/array | Возвращать только историю с указанных узлов сети. |
| itemids | ID/array | Возвращать только историю с указанных элементов данных. |
| time_from | timestamp | Возвращать только значения, полученные после указанного времени или в указанное время. |
| time_till | timestamp | Возвращать только значения, полученные до указанного времени или в указанное время. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `itemid`, `clock`, `ns`. |
| search | object | Возвращать результаты, соответствующие заданному шаблону (без учета регистра).<br><br>Принимает объект, где ключи — это имена свойств, а значения — строки для поиска. Если дополнительные параметры не заданы, будет выполнен поиск `LIKE "%…%"`.<br><br>Не поддерживает поиск по `value`, если для `history` задано значение 6 (JSON). |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |

| Parameter | Type | Description |
|---|---|---|
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение данных истории элемента данных

Вернуть 10 последних значений, полученных от числового (float) элемента данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "history.get",
    "params": {
        "output": "extend",
        "history": 0,
        "itemids": "23296",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 10
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Источник

CHistory::get() в *ui/include/classes/api/services/CHistory.php*.

**history.push**

Описание

`object history.push(  /  itemHistoryData)`

Этот метод позволяет отправлять данные истории элементов на сервер Zabbix.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Данные истории элемента данных для отправки.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| itemid | ID | ID связанного элемента данных.<br><br>Поведение параметра:<br>- *required* if `host` and `key` are not set |

| Parameter | Type | Description |
|---|---|---|
| host | string | Техническое имя узла сети.<br><br>Поведение параметра:<br>- *required* if `itemid` is not set |
| key | string | Ключ элемента данных.<br><br>Поведение параметра:<br>- *required* if `itemid` is not set |
| value | mixed | Значение элемента данных.<br><br>Поведение параметра:<br>- *required* |
| clock | timestamp | Время, когда было получено значение. |
| ns | integer | Наносекунды, когда было получено значение. |

Возвращаемые значения

`(object)` Возвращает результат операции отправки данных.

Примеры

Отправка данных истории элемента данных

Отправьте данные истории элемента данных на сервер Zabbix для элементов ”10600”, ”10601” и ”999999”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "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
}
```


См. также

- Элементы данных типа trapper
- Элементы данных HTTP-агента
- Узел сети
- Элемент данных

Исходный текст

CHistory::push() в *ui/include/classes/api/services/CHistory.php*.

#### Карта значений

Этот класс предназначен для работы с картами значений.

Ссылки на объекты:

- Карта значений
  - Сопоставления значений

Доступные методы:

- valuemap.create - создать новые карты значений
- valuemap.delete - удалить карты значений
- valuemap.get - получить карты значений
- valuemap.update - обновить карты значений

#### Объект карты значений

Следующие объекты напрямую связаны с API `valuemap`.

Карта значений

Объект карты значений имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| valuemapid | ID | ID карты значений.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| hostid | ID | ID узла сети или шаблона, которому принадлежит карта значений.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |
| name | string | Имя карты значений.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| mappings | array | Сопоставления значений для текущей карты значений. Объект сопоставления подробно описан ниже.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных карт значений с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается*, если карта значений принадлежит шаблону |

Сопоставления значений

Объект сопоставлений значений определяет сопоставления значений карты значений. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип сопоставления совпадения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* сопоставление будет применено, если значение равно;<br>1 - сопоставление будет применено, если значение больше или равно[1];<br>2 - сопоставление будет применено, если значение меньше или равно[1];<br>3 - сопоставление будет применено, если значение находится в диапазоне (диапазоны включительны; можно задать несколько диапазонов, разделенных символом запятой)[1];<br>4 - сопоставление будет применено, если значение соответствует регулярному выражению[2];<br>5 - если совпадения не найдены, сопоставление не будет применено, и будет использовано значение по умолчанию.<br><br>Если `type` установлено в ”0”, ”1”, ”2”, ”3”, ”4”, то `value` не может быть пустым.<br><br>Если `type` установлено в ”5”, то `value` должно быть пустым. |
| value | string | Исходное значение.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлено в ”1”, ”2”, ”3”, ”4”<br>- *поддерживается* если `type` установлено в ”5” |
| newvalue | string | Значение, на которое сопоставляется исходное значение.<br><br>Поведение свойства:<br>- *обязательно* |

[1] поддерживается только для элементов данных с типом значения ”numeric unsigned”, ”numeric float”.<br> [2] поддерживается только для элементов данных с типом значения ”character”.<br>

**valuemap.create**

Описание

`object valuemap.create( / valuemaps)`

Этот метод позволяет создавать новые карты значений.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Карты значений для создания.

Метод принимает карты значений со стандартными свойствами карты значений.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных карт значений в свойстве `valuemapids`. Порядок возвращаемых идентификаторов соответствует порядку переданных карт значений.

Примеры

Создание карты значений

Создайте одну карту значений с двумя сопоставлениями.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.create",
    "params": {
        "hostid": "50009",
        "name": "Service state",
        "mappings": [
            {
                "type": "1",
                "value": "1",
                "newvalue": "Up"
            },
            {
                "type": "5",
                "newvalue": "Down"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "1"
        ]
    },
    "id": 1
}
```

Источник

CValueMap::create() в *ui/include/classes/api/services/CValueMap.php*.

**valuemap.delete**

Описание

`object valuemap.delete( valuemapids)`

Этот метод позволяет удалить преобразование значений.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID удаляемых преобразований значений.

Возвращаемые значения

( ) Возвращает объект, который содержит ID удаленных преобразований значений, указанных в свойстве `valuemapids`.

Примеры

Удаление нескольких карт значений

Удаление двух карт значений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Источник

CValueMap::delete() в *ui/include/classes/api/services/CValueMap.php*.

**valuemap.get**

Описание

`integer/array valuemap.get( )`

Метод позволяет получить преобразование значений в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| valuemapids | ID/array | Возвращать только карты значений с указанными ID. |
| selectMappings | query | Возвращать сопоставления значений для текущей карты значений в свойстве `mappings`.<br><br>Поддерживает `count`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `valuemapid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение карт значений

Получить все настроенные карты значений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Получить одну карту значений с ее сопоставлениями.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend",
        "selectMappings": "extend",
        "valuemapids": ["4"]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Источник

CValueMap::get() в *ui/include/classes/api/services/CValueMap.php*.

**valuemap.update**

Описание

`object valuemap.update(  /  valuemaps)`

Этот метод позволяет обновлять существующие преобразования значений.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства карты значений, которые необходимо обновить.

Свойство `valuemapid` должно быть определено для каждой карты значений, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных преобразований значений, указанных в свойстве`valuemapids`.

Примеры

Изменение имени карты значений

Измените имя карты значений на "Device status".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "name": "Device status"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Изменение сопоставлений для одной карты значений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "mappings": [
            {
                "type": "0",
```

```
                "value": "0",
                "newvalue": "Online"
            },
            {
                "type": "0",
                "value": "1",
                "newvalue": "Offline"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Источник

CValueMap::update() в *ui/include/classes/api/services/CValueMap.php*.

#### Карта значков

Этот класс предназначен для работы с картами значков.

Ссылки на объекты:

- Карта значков
- Сопоставление значков

Доступные методы:

- iconmap.create - создать новые карты значков
- iconmap.delete - удалить карты значков
- iconmap.get - получить карты значков
- iconmap.update - обновить карты значков

#### Объект icon map

Следующие объекты напрямую связаны с API `iconmap`.

Карта значков

Объект карты значков имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| iconmapid | ID | ID карты значков.<br><br>Property behavior:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| default_iconid | ID | ID значка по умолчанию.<br><br>Property behavior:<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| name | string | Имя карты значков.<br><br>Property behavior:<br>- *обязательно* для операций создания |

Сопоставление значков

Объект сопоставления значков определяет конкретный значок, который будет использоваться для узлов сети с определенным значением поля инвентаризации. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| iconid | ID | ID значка, используемого сопоставлением значков.<br><br>Поведение свойства:<br>- *required* |
| expression | string | Выражение для сопоставления со значением поля инвентаризации.<br><br>Поведение свойства:<br>- *required* |
| inventory_link | integer | ID поля инвентаризации узла сети.<br><br>См. объект инвентаризации узла сети для списка поддерживаемых полей инвентаризации.<br><br>Поведение свойства:<br>- *required* |
| sortorder | integer | Позиция сопоставления значков в карте значков.<br><br>Поведение свойства:<br>- *read-only* |

**iconmap.create**

Описание

`object iconmap.create(  /  iconMaps)`

Этот метод позволяет создавать новые соответствия иконок.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Карта значков для создания.

В дополнение к стандартным свойствам карты значков, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| mappings | array | Сопоставления значков, которые будут созданы для карты значков.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных соответствий иконок, указанных в свойстве `iconmapids`. Порядок возвращаемых ID совпадает с порядком переданных соответствий иконок.

Примеры

Создание карты значков

Создайте карту значков для отображения узлов сети разных типов.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2"
        ]
    },
    "id": 1
}
```

См. также

- Сопоставление значков

Источник

CIconMap::create() в *ui/include/classes/api/services/CIconMap.php*.

**iconmap.delete**

Описание

`object iconmap.delete( iconMapIds)`

Этот метод позволяет удалять соответствия иконок.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID удаляемых соответствий иконок.

Возвращаемые значения

( ) Возвращает объект, который содержит ID удаленных соответствий иконок, указанных в свойстве `iconmapids`.

Примеры

Удаление нескольких карт значков

Удаление двух карт значков.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.delete",
    "params": [
        "2",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CIconMap::delete() в *ui/include/classes/api/services/CIconMap.php*.

**iconmap.get**

Описание

`integer/array iconmap.get( )`

Метод позволяет получать карты иконок по заданным параметры.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| iconmapids | ID/array | Возвращать только карты значков с указанными ID. |
| sysmapids | ID/array | Возвращать только карты значков, используемые в указанных картах. |
| selectMappings | query | Возвращать свойство `mappings` с используемыми сопоставлениями значков. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `iconmapid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |

| Parameter | Type | Description |
|---|---|---|
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение карты значков

Получить все данные о карте значков "3".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.get",
    "params": {
        "iconmapids": "3",
        "output": "extend",
        "selectMappings": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "               ",
            "default_iconid": "2"
        }
```

```
    ],
    "id": 1
}
```

См. также

- Сопоставление значков

Источник

CIconMap::get() в *ui/include/classes/api/services/CIconMap.php*.

**iconmap.update**

Описание

`object iconmap.update(  /  iconMaps)`

Этот метод позволяет обновлять существующие карты значков.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства карты значков, которые нужно обновить.

Для каждой карты значков должно быть определено свойство `iconmapid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств карты значков, метод принимает следующие параметры.

| Параметр | Тип | Описание |
|---|---|---|
| mappings | array | Сопоставления значков для замены существующих сопоставлений значков. |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных соответствий иконок, указанных в свойстве `iconmapids`.

Примеры

Переименование карты значков

Переименуйте карту значков в "OS icons".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.update",
    "params": {
        "iconmapid": "1",
        "name": "OS icons"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "1"
        ]
```

```
    },
    "id": 1
}
```

См. также

- Сопоставление значков

Источник

CIconMap::update() в *ui/include/classes/api/services/CIconMap.php*.

**Каталог пользователей**

Этот класс предназначен для работы с каталогами пользователей.

Ссылки на объекты:

- Каталог пользователей
  - Сопоставления типов медиа
  - Сопоставления групп для автоматического создания

Доступные методы:

- userdirectory.create - создать новый каталог пользователей
- userdirectory.delete - удалить каталог пользователей
- userdirectory.get - получить каталог пользователей
- userdirectory.update - обновить каталог пользователей
- userdirectory.test - проверить подключение к каталогу пользователей

**Объект каталога пользователей**

Следующие объекты напрямую связаны с API `userdirectory`.

Каталог пользователей

Объект каталога пользователей имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| userdirectoryid | ID | ID каталога пользователей.<br><br>Если каталог пользователей удален, значение свойства объекта User `userdirectoryid` для всех пользователей, связанных с удаленным каталогом пользователей, устанавливается в ”0”.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| idp_type | integer | Тип протокола аутентификации, используемого поставщиком удостоверений для каталога пользователей.<br>Обратите внимание, что может существовать только один каталог пользователей типа SAML.<br><br>Возможные значения:<br>1 - каталог пользователей типа LDAP;<br>2 - каталог пользователей типа SAML.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| group_name | string | Атрибут каталога пользователей LDAP/SAML, содержащий имя группы, используемое для сопоставления групп между каталогом пользователей LDAP/SAML и Zabbix.<br><br>Пример: *cn*<br><br>Поведение свойства:<br>- *обязательно*, если `provision_status` установлен в ”Enabled” и `saml_jit_status` объекта Authentication установлен в ”Enabled for configured SAML IdPs” |
| user_username | string | Атрибут каталога пользователей LDAP/SAML (также атрибут SCIM, если `scim_status` установлен в ”SCIM provisioning is enabled”), содержащий имя пользователя, которое используется как значение свойства объекта User `name` при создании пользователя.<br><br>Примеры: *cn*, *commonName*, *displayName*, *name* |
| user_lastname | string | Атрибут каталога пользователей LDAP/SAML (также атрибут SCIM, если `scim_status` установлен в ”SCIM provisioning is enabled”), содержащий фамилию пользователя, которая используется как значение свойства объекта User `surname` при создании пользователя.<br><br>Примеры: *sn*, *surname*, *lastName* |
| provision_status | integer | Статус provision для каталога пользователей.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Disabled (provisioning пользователей, созданных этим каталогом пользователей, отключен);<br>1 - Enabled (provisioning пользователей, созданных этим каталогом пользователей, включен; дополнительно должен быть включен статус LDAP или SAML provisioning (`ldap_jit_status` или `saml_jit_status` объекта Authentication)). |
| provision_groups | array | Массив объектов сопоставлений групп provisioning для сопоставления шаблона группы пользователей LDAP/SAML с группой пользователей и ролью пользователя Zabbix.<br><br>Поведение свойства:<br>- *обязательно*, если `provision_status` установлен в ”Enabled” |
| provision_media | array | Массив объектов сопоставлений типов медиа для сопоставления атрибутов медиа пользователя LDAP/SAML (например, email) с медиа пользователя Zabbix для отправки уведомлений. |
| **LDAP-specific properties:** | | |
| name | string | Уникальное имя каталога пользователей.<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type LDAP” |
| host | string | Имя узла сети, IP-адрес или URI LDAP-сервера.<br>URI должен содержать схему (`ldap://` или `ldaps://`), узел сети и порт (необязательно).<br><br>Примеры:<br>*host.example.com*<br>*127.0.0.1*<br>*ldap://ldap.example.com:389*<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type LDAP” |

| Property | Type | Description |
|---|---|---|
| port | integer | Порт LDAP-сервера.<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type LDAP” |
| base_dn | string | Базовый путь каталога пользователей LDAP к учетным записям пользователей.<br><br>Примеры:<br>*ou=Users,dc=example,dc=org*<br>*ou=Users,ou=system* (для OpenLDAP)<br>*DC=company,DC=com* (для Microsoft Active Directory)<br>*uid=%{user},dc=example,dc=com* (для прямой привязки пользователя; заполнитель ”*%{user}*” обязателен)<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type LDAP” |
| search_attribute | string | Атрибут каталога пользователей LDAP, по которому определяется учетная запись пользователя на основе информации, переданной в запросе на вход.<br><br>Примеры:<br>*uid* (для OpenLDAP)<br>*sAMAccountName* (для Microsoft Active Directory)<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type LDAP” |
| bind_dn | string | Учетная запись LDAP-сервера для привязки и поиска на LDAP-сервере.<br><br>Для прямой привязки пользователя и анонимной привязки `bind_dn` должен быть пустым.<br><br>Примеры:<br>*uid=ldap_search,ou=system* (для OpenLDAP)<br>*CN=ldap_search,OU=user_group,DC=company,DC=com* (для Microsoft Active Directory)<br>*CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| bind_password | string | Пароль LDAP для учетной записи, используемой для привязки и поиска на LDAP-сервере.<br><br>Для прямой привязки пользователя и анонимной привязки `bind_password` должен быть пустым.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| description | string | Описание каталога пользователей.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |

| Property | Type | Description |
|---|---|---|
| group_basedn | string | Базовый путь каталога пользователей LDAP к группам; используется для настройки проверки членства пользователя в каталоге пользователей LDAP.<br><br>Игнорируется при provisioning пользователя, если задан `group_membership`.<br><br>Пример: *ou=Groups,dc=example,dc=com*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| group_filter | string | Строка фильтра для получения групп каталога пользователей LDAP, членом которых является пользователь; используется для настройки проверки членства пользователя в каталоге пользователей LDAP.<br><br>Игнорируется при provisioning пользователя, если задан `group_membership`.<br><br>Поддерживаемые заполнители `group_filter`:<br>*%{attr}* - атрибут поиска (заменяется значением свойства `search_attribute`);<br>*%{groupattr}* - атрибут группы (заменяется значением свойства `group_member`);<br>*%{host}* - имя узла сети, IP-адрес или URI LDAP-сервера (заменяется значением свойства `host`);<br>*%{user}* - имя пользователя Zabbix.<br><br>По умолчанию: *(%{groupattr}=%{user})*<br><br>Примеры:<br>- *(member=uid=%{ref},ou=Users,dc=example,dc=com)* будет соответствовать ”User1”, если объект группы LDAP содержит атрибут ”*member*” со значением ”*uid=User1,ou=Users,dc=example,dc=com*”, и вернет группу, членом которой является ”User1”;<br>- *(%{groupattr}=cn=%{ref},ou=Users,ou=Zabbix,DC=example,DC=com)* будет соответствовать ”User1”, если объект группы LDAP содержит атрибут, указанный в свойстве `group_member`, со значением ”*cn=User1,ou=Users,ou=Zabbix,DC=example,DC=com*”, и вернет группу, членом которой является ”User1”.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| group_member | string | Атрибут каталога пользователей LDAP, содержащий информацию о членах группы; используется для настройки проверки членства пользователя в каталоге пользователей LDAP.<br><br>Игнорируется при provisioning пользователя, если задан `group_membership`.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |

| Property | Type | Description |
|---|---|---|
| group_membership | string | Атрибут каталога пользователей LDAP, содержащий информацию о группах, к которым принадлежит пользователь.<br><br>Пример: *memberOf*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| search_filter | string | Пользовательская строка фильтра, используемая для поиска и аутентификации пользователя в каталоге пользователей LDAP на основе информации, переданной в запросе на вход.<br><br>Поддерживаемые заполнители `search_filter`:<br>*%{attr}* - имя атрибута поиска (например, *uid*, *sAMAccountName*);<br>*%{user}* - имя пользователя Zabbix.<br><br>По умолчанию: *(%{attr}=%{user})*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| start_tls | integer | Параметр конфигурации LDAP-сервера, который позволяет защищать связь с LDAP-сервером с помощью Transport Layer Security (TLS).<br><br>Обратите внимание, что `start_tls` должен быть установлен в ”Disabled” для узлов сети, использующих протокол `ldaps://`.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Disabled;<br>1 - Enabled.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| user_ref_attr | string | Атрибут каталога пользователей LDAP, используемый для ссылки на объект пользователя. Значение `user_ref_attr` используется для получения значений из указанного атрибута в каталоге пользователей и подстановки их вместо заполнителя *%{ref}* в строке `group_filter`.<br><br>Примеры: *cn*, *uid*, *member*, *uniqueMember*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type LDAP” |
| **SAML-specific properties:** | | |
| idp_entityid | string | URI, который идентифицирует поставщика удостоверений и используется для обмена сообщениями SAML с поставщиком удостоверений.<br><br>Пример: *https://idp.example.com/idp*<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type SAML” |

| Property | Type | Description |
|---|---|---|
| sp_entityid | string | URL или любая строка, которая идентифицирует поставщика услуг поставщика удостоверений.<br><br>Примеры:<br>*https://idp.example.com/sp*<br>*zabbix*<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type SAML” |
| username_attribute | string | Атрибут каталога пользователей SAML (также атрибут SCIM, если `scim_status` установлен в ”SCIM provisioning is enabled”), содержащий имя пользователя, которое сравнивается со значением свойства объекта User `username` при аутентификации.<br><br>Примеры: *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*<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type SAML” |
| sso_url | string | URL службы единого входа SAML поставщика удостоверений, на которую Zabbix будет отправлять запросы аутентификации SAML.<br><br>Пример: *http://idp.example.com/idp/sso/saml*<br><br>Поведение свойства:<br>- *обязательно*, если `idp_type` установлен в ”User directory of type SAML” |
| slo_url | string | URL службы единого выхода SAML поставщика удостоверений, на которую Zabbix будет отправлять запросы выхода SAML.<br><br>Пример: *https://idp.example.com/idp/slo/saml*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| encrypt_nameid | integer | Следует ли шифровать SAML Name ID.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не шифровать Name ID;<br>1 - Шифровать Name ID.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| encrypt_assertions | integer | Следует ли шифровать SAML assertions.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не шифровать assertions;<br>1 - Шифровать assertions.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |

| Property | Type | Description |
|---|---|---|
| nameid_format | string | Формат Name ID для поставщика услуг SAML поставщика удостоверений.<br><br>Примеры:<br>*urn:oasis:names:tc:SAML:2.0:nameid-format:persistent*<br>*urn:oasis:names:tc:SAML:2.0:nameid-format:transient*<br>*urn:oasis:names:tc:SAML:2.0:nameid-format:kerberos*<br>*urn:oasis:names:tc:SAML:2.0:nameid-format:entity*<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| scim_status | integer | Включено или отключено provisioning SCIM для SAML.<br><br>Возможные значения:<br>0 - *(по умолчанию)* SCIM provisioning отключен;<br>1 - SCIM provisioning включен.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| sign_assertions | integer | Следует ли подписывать SAML assertions с помощью подписи SAML.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не подписывать assertions;<br>1 - Подписывать assertions.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| sign_authn_requests | integer | Следует ли подписывать запросы SAML AuthN с помощью подписи SAML.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не подписывать запросы AuthN;<br>1 - Подписывать запросы AuthN.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| sign_messages | integer | Следует ли подписывать сообщения SAML с помощью подписи SAML.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не подписывать сообщения;<br>1 - Подписывать сообщения.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |
| sign_logout_requests | integer | Следует ли подписывать запросы выхода SAML с помощью подписи SAML.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не подписывать запросы выхода;<br>1 - Подписывать запросы выхода.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в ”User directory of type SAML” |

| Property | Type | Description |
|---|---|---|
| sign_logout_responses | integer | Следует ли подписывать ответы на выход SAML с помощью подписи SAML.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не подписывать ответы на выход;<br>1 - Подписывать ответы на выход.<br><br>Поведение свойства:<br>- *поддерживается*, если `idp_type` установлен в "User directory of type SAML" |

Сопоставления типов носителей

Объект сопоставлений типов носителей имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| userdirectory_mediaid | ID | ID сопоставления типа носителя.<br><br>Поведение свойства:<br>- *только для чтения* |
| name | string | Отображаемое имя в списке сопоставлений типов носителей.<br><br>Поведение свойства:<br>- *обязательное* |
| mediatypeid | ID | ID типа носителя, который будет создан; используется как значение свойства объекта Media `mediatypeid`.<br><br>Поведение свойства:<br>- *обязательное* |
| attribute | string | Атрибут каталога пользователей LDAP/SAML (также атрибут SCIM, если `scim_status` имеет значение "SCIM provisioning is enabled"), который содержит носитель пользователя (например, *user@example.com*), используемый как значение свойства объекта Media `sendto`.<br><br>Если он присутствует в данных, полученных от поставщика удостоверений LDAP/SAML, и его значение не пустое, это вызовет создание носителя для подготовленного пользователя.<br><br>Поведение свойства:<br>- *обязательное* |
| active | integer | Значение свойства `active` носителя пользователя при создании носителя для подготовленного пользователя.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включено;<br>1 - отключено. |
| severity | integer | Значение свойства `severity` носителя пользователя при создании носителя для подготовленного пользователя.<br><br>По умолчанию: 63. |
| period | string | Значение свойства `period` носителя пользователя при создании носителя для подготовленного пользователя.<br><br>По умолчанию: 1-7,00:00-24:00. |

Сопоставления групп для автоматического создания пользователей

Сопоставления групп для автоматического создания пользователей имеют следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Полное имя группы (например, *Zabbix administrators*) в каталоге пользователей LDAP/SAML (а также SCIM, если `scim_status` имеет значение "SCIM provisioning is enabled").<br>Поддерживает символ подстановки "*".<br>Уникально среди всех сопоставлений групп для автоматического создания пользователей.<br><br>Property behavior:<br>- *required* |
| roleid | ID | ID роли пользователя, назначаемой пользователю.<br><br>Если совпадает несколько сопоставлений групп для автоматического создания пользователей, пользователю назначается роль с наивысшим типом пользователя (*User*, *Admin* или *Super admin*). Если есть несколько ролей с одинаковым типом пользователя, пользователю назначается первая роль (отсортированная в алфавитном порядке).<br><br>Property behavior:<br>- *required* |
| user_groups | array | Массив объектов ID групп пользователей Zabbix. Каждый объект имеет следующие свойства:<br>`usrgrpid` - (ID) ID группы пользователей Zabbix, назначаемой пользователю.<br><br>Если совпадает несколько сопоставлений групп для автоматического создания пользователей, пользователю назначаются группы пользователей Zabbix из всех совпавших сопоставлений.<br><br>Property behavior:<br>- *required* |

**userdirectory.create**

Описание

`object userdirectory.create(object/array userDirectory)`

Этот метод позволяет создавать новые каталоги пользователей.

> **Note:**
> Этот метод доступен только пользователям типа *Super admin*.

Параметры

`(object/array)` Каталоги пользователей для создания.

Метод принимает каталоги пользователей со стандартными свойствами каталога пользователей.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных каталогов пользователей в свойстве `userdirectoryids`. Порядок возвращаемых идентификаторов соответствует порядку переданных каталогов пользователей.

Примеры

Создание каталога пользователей

Создайте каталог пользователей для аутентификации пользователей с использованием StartTLS через LDAP. Обратите внимание, что для аутентификации пользователей через LDAP должна быть включена LDAP-аутентификация.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Создание каталога пользователей (с включенным JIT provisioning)

Создайте каталог пользователей для аутентификации пользователей через LDAP (с включенным JIT provisioning). Обратите внимание, что для аутентификации пользователей через LDAP должна быть включена LDAP-аутентификация.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Исходный код

CUserDirectory::create() в *ui/include/classes/api/services/CUserDirectory.php*.

**userdirectory.delete**

Описание

`object userdirectory.delete(array userDirectoryIds)`

Этот метод позволяет удалять каталоги пользователей. Каталог пользователей нельзя удалить, если он напрямую используется хотя бы одной группой пользователей.<br> Каталог пользователей LDAP по умолчанию нельзя удалить, если `authentication.ldap_configured` установлено в `1` или если после удаления останется меньше каталогов пользователей.

**Note:**
Этот метод доступен только для типа пользователя *Super admin*.

Параметры

`(array)` Идентификаторы каталогов пользователей, которые нужно удалить.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных каталогов пользователей в свойстве `userdirectoryids`.

Примеры

Удаление нескольких каталогов пользователей

Удаление двух каталогов пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.delete",
    "params": [
        "2",
        "12"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2",
            "12"
        ]
    },
    "id": 1
}
```

Исходный код

CUserDirectory::delete() в *ui/include/classes/api/services/CUserDirectory.php*.

**userdirectory.get**

Описание

`integer/array userdirectory.get(object parameters)`

Метод позволяет получать каталоги пользователей в соответствии с заданными параметрами.

**Note:**
Этот метод доступен только для типов пользователей *Super admin*.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| userdirectoryids | ID/array | Возвращать только каталоги пользователей с указанными ID. |
| selectUsrgrps | query | Возвращать свойство `usrgrps` с группами пользователей, связанными с каталогом пользователей.<br><br>Поддерживает `count`. |
| selectProvisionMedia | query | Возвращать свойство `provision_media` с сопоставлениями типов медиа, связанными с каталогом пользователей. |
| selectProvisionGroups | query | Возвращать свойство `provision_groups` с сопоставлениями групп для автоматического создания, связанными с каталогом пользователей. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `name`. |

| Parameter | Type | Description |
|---|---|---|
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений.<br><br>Поддерживаемые свойства: `userdirectoryid`, `idp_type`, `provision_status`. |
| search | object | Возвращать результаты, соответствующие указанному шаблону (без учета регистра).<br><br>Принимает объект, где ключи — это имена свойств, а значения — строки для поиска. Если дополнительные параметры не указаны, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживаемые свойства: `name`, `description`.<br><br>Каталог пользователей типа SAML будет иметь пустые значения свойств `name` и `description`. Оба свойства можно изменить с помощью операции `userdirectory.update`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение каталогов пользователей

Получите все каталоги пользователей с дополнительными свойствами, которые отображают сопоставления типов медиа и сопоставления групп автоматического создания, связанные с каждым каталогом пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.get",
    "params": {
        "output": "extend",
        "selectProvisionMedia": "extend",
        "selectProvisionGroups": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Группа пользователей

Исходный код

CUserDirectory::get() в *ui/include/classes/api/services/CUserDirectory.php*.

**userdirectory.test**

Описание

`object userdirectory.test(array userDirectory)`

Этот метод позволяет проверить настройки подключения к каталогу пользователей.

**Note:**
Этот метод также позволяет проверить, какие настроенные данные соответствуют параметрам каталога пользователей для подготовки пользователей (например, какая роль пользователя, какие группы пользователей, какие медиа пользователя будут назначены пользователю). Для такого типа проверки API-запрос следует выполнять для каталога пользователей, у которого `provision_status` установлен в enabled.

**Note:**
Этот метод доступен только для типа пользователя *Super admin*.

Параметры

`(object)` Свойства каталога пользователей.

Поскольку API `userdirectory.get` не возвращает поле `bind_password`, необходимо указать `userdirectoryid` и/или `bind_password`.<br> Помимо стандартных свойств каталога пользователей, метод принимает следующие параметры.

| Параметр | Тип | Описание |
|---|---|---|
| test_username | string | Имя пользователя для проверки в каталоге пользователей. |
| test_password | string | Пароль, связанный с именем пользователя, для проверки в каталоге пользователей. |

Возвращаемые значения

`(bool)` Возвращает true в случае успешного выполнения.

Примеры

Тестирование каталога пользователей для существующего пользователя

Проверка каталога пользователей ”3” для ”user1”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Тестирование каталога пользователей для несуществующего пользователя

Проверка каталога пользователей ”3” для несуществующего ”user2”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32500,
        "message": "Application error.",
        "data": "                    ,                        ."
    },
    "id": 1
}
```

Тестирование каталога пользователей для подготовки пользователей

Проверьте каталог пользователей ”3”, чтобы определить, какие настроенные данные соответствуют параметрам каталога пользователей для подготовки ”user3” (например, какая роль пользователя, группы пользователей и средства оповещения будут назначены пользователю).

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "username": "user3",
        "name": "John",
        "surname": "Doe",
        "medias": [],
        "usrgrps": [
            {
                "usrgrpid": "8"
            },
            {
                "usrgrpid": "7"
            }
        ],
        "roleid": "2",
        "userdirectoryid": "2"
    },
    "id": 1
}
```

Исходный код

CUserDirectory::test() в *ui/include/classes/api/services/CUserDirectory.php*.

**userdirectory.update**

Описание

`object userdirectory.update(object/array userDirectory)`

Этот метод позволяет обновлять существующие каталоги пользователей.

> **Note:**
> Этот метод доступен только для пользователей типа *Super admin*.

Параметры

`(object/array)` Свойства каталога пользователей, которые нужно обновить.

Свойство `userdirectoryid` должно быть определено для каждого каталога пользователей, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных каталогов пользователей в свойстве `userdirectoryids`.

Примеры

Обновление bind-пароля для каталога пользователей

Установите новый bind-пароль для каталога пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.update",
    "params": {
        "userdirectoryid": "3",
        "bind_password": "newldappassword"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Обновление сопоставлений для каталога пользователей

Обновите сопоставления групп предоставления и сопоставления типов медиа для каталога пользователей "2".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Источник

CUserDirectory::update() в *ui/include/classes/api/services/CUserDirectory.php*.

**Коннектор**

Этот класс предназначен для работы с коннекторами.

Ссылки на объекты:

- Коннектор
- Фильтр по тегам

Доступные методы:

- connector.create - создать новые коннекторы
- connector.delete - удалить коннекторы
- connector.get - получить коннекторы
- connector.update - обновить коннекторы

**Объект connector**

Следующие объекты напрямую связаны с API `connector`.

Connector

Объект connector имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| connectorid | ID | ID connector.<br><br>Property behavior:<br>- *только для чтения*<br>- *required* для операций обновления |
| name | string | Имя connector.<br><br>Property behavior:<br>- *required* для операций создания |
| url | string | URL конечной точки, то есть URL получателя.<br>Поддерживаются пользовательские макросы.<br><br>Property behavior:<br>- *required* для операций создания |
| protocol | integer | Протокол связи.<br><br>Possible values:<br>0 - *(default)* Zabbix Streaming Protocol v1.0. |
| data_type | integer | Тип данных.<br><br>Possible values:<br>0 - *(default)* Значения элементов данных;<br>1 - События. |
| item_value_type | integer | Сумма типов значений элементов данных, которые будут отправлены.<br><br>Possible values:<br>1 - Numeric (float);<br>2 - Character;<br>4 - Log;<br>8 - Numeric (unsigned);<br>16 - Text;<br>32 - Binary.<br><br>Default: 31 - Все типы элементов данных (кроме binary).<br><br>Property behavior:<br>- *supported* если `data_type` установлено в ”Item values”. |

| Property | Type | Description |
|---|---|---|
| max_records | integer | Максимальное число событий или элементов данных, которые можно отправить в одном сообщении.<br><br>Possible values: 0-2147483647 (max value of 32-bit signed integer).<br><br>Default: 0 - Без ограничений. |
| max_senders | integer | Количество процессов отправителя, которые будут запущены для этого connector.<br><br>Possible values: 1-100.<br><br>Default: 1. |
| max_attempts | integer | Количество попыток.<br><br>Possible values: 1-5.<br><br>Default: 1. |
| attempt_interval | string | Интервал между повторными попытками.<br>Принимает секунды.<br><br>Possible values: 0s-10s.<br><br>Default: 5s.<br><br>Property behavior:<br>- *supported* если `max_attempts` больше 1. |
| timeout | string | Тайм-аут.<br>Поддерживаются суффиксы времени (например, 30s, 1m).<br>Поддерживаются пользовательские макросы.<br><br>Possible values: 1s-60s.<br><br>Default: 5s. |
| http_proxy | string | Строка подключения к HTTP(S) proxy в виде *[protocol]://[username[:password]@]proxy.example.com[:port]*.<br><br>Поддерживаются пользовательские макросы. |
| authtype | integer | Метод HTTP-аутентификации.<br><br>Possible values:<br>0 - *(default)* Нет;<br>1 - Basic;<br>2 - NTLM;<br>3 - Kerberos;<br>4 - Digest;<br>5 - Bearer. |
| username | string | Имя пользователя.<br>Поддерживаются пользовательские макросы.<br><br>Property behavior:<br>- *supported* если `authtype` установлено в ”Basic”, ”NTLM”, ”Kerberos” или ”Digest” |
| password | string | Пароль.<br>Поддерживаются пользовательские макросы.<br><br>Property behavior:<br>- *supported* если `authtype` установлено в ”Basic”, ”NTLM”, ”Kerberos” или ”Digest” |

| Property | Type | Description |
|---|---|---|
| token | string | Bearer token.<br>Поддерживаются пользовательские макросы.<br><br>Property behavior:<br>- *required* если `authtype` установлено в "Bearer" |
| verify_peer | integer | Проверять ли, что сертификат узла сети является подлинным.<br><br>Possible values:<br>0 - Не проверять;<br>1 - *(default)* Проверять. |
| verify_host | integer | Проверять ли, что имя узла сети для соединения совпадает с именем в сертификате узла сети.<br><br>Possible values:<br>0 - Не проверять;<br>1 - *(default)* Проверять. |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br>Поддерживаются пользовательские макросы. |
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br>Поддерживаются пользовательские макросы. |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br>Поддерживаются пользовательские макросы. |
| description | text | Описание connector. |
| status | integer | Включен ли connector.<br><br>Possible values:<br>0 - Disabled;<br>1 - *(default)* Enabled. |
| tags_evaltype | integer | Метод evaluation тегов.<br><br>Possible values:<br>0 - *(default)* And/Or;<br>2 - Or. |

Фильтр тегов

Фильтр тегов позволяет экспортировать только совпадающие значения элементов данных или события. Если он не задан, будет экспортировано все. Объект фильтра тегов имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега.<br><br>Поведение свойства:<br>- *required* |
| operator | integer | Оператор условия.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Равно;<br>1 - Не равно;<br>2 - Содержит;<br>3 - Не содержит;<br>12 - Существует;<br>13 - Не существует. |
| value | string | Значение тега. |

**connector.create**

Описание

`object connector.create(object/array connectors)`

Этот метод позволяет создавать новые объекты коннектора.

> **Note:**
> Этот метод доступен только для типа пользователя *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Чтобы получить больше информации смотрите Роли пользователя.

Параметры

`(object/array)` Объекты коннектора для создания.

Помимо стандартных свойств коннектора, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| tags | array | Фильтр тегов коннектора. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных коннекторов в свойстве `connectorids`. Порядок возвращаемых идентификаторов соответствует порядку переданных коннекторов.

Примеры

Создание соединителя

Создайте соединитель для экспорта событий триггера с фильтром тегов. HTTP-аутентификация будет выполняться с использованием токена Bearer.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorid": [
            "3"
        ]
    },
    "id": 1
}
```

Источник

CConnector::create() в *ui/include/classes/api/services/CConnector.php*.

**connector.delete**

Описание

`object connector.delete( connectorids)`

Этот метод позволяет удалить записи коннектора.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Parameters

`(array)` IDs of the connectors to delete.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных коннекторов в свойстве `connectorids`.

Примеры

Удаление нескольких соединителей

Удалить две записи соединителя.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "connector.delete",
    "params": [
        3,
        5
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "3",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CConnector::delete() в *ui/include/classes/api/services/CConnector.php*.

**connector.get**

Описание

`integer/array connector.get( )`

Метод позволяет получать объекты коннектора по заданным параметрам.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| connectorids | ID/array | Возвращать только коннекторы с указанными ID. |
| selectTags | query | Возвращать свойство `tags` с фильтром тегов коннектора.<br><br>Поддерживает `count`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `connectorid`, `name`, `data_type`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если был использован параметр `countOutput`.

Примеры

Получение всех коннекторов

Получите все данные обо всех коннекторах и их свойствах.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "connector.get",
    "params": {
        "output": "extend",
        "selectTags": ["tag", "operator", "value"],
        "preservekeys": true
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Источник

CConnector:get() в *ui/include/classes/api/services/CConnector.php*.

### connector.update

Описание

`object connector.update( / connectors)`

Этот метод позволяет обновить существующие коннекторы.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства коннектора, которые необходимо обновить.

Для каждого коннектора должно быть определено свойство `connectorid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам коннектора, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| tags | array | Фильтр тегов коннектора, который заменит текущий фильтр тегов. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных коннекторов в свойстве `connectorids`.

Примеры

Изменение типа HTTP-аутентификации

Измените тип аутентификации HTTP на «Носитель» для соединителя с идентификатором «3».

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "connector.update",
    "params": {
        "connectorid": 3,
        "authtype": 5,
        "token": "{$DATA_EXPORT_BEARER_TOKEN}"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "3"
        ]
    },
    "id": 1
}
```

Обновление фильтра тегов

Измените фильтр тегов для соединителя с идентификатором «5».

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CConnector::update() в *ui/include/classes/api/services/CConnector.php*.

#### Конфигурация

Этот класс предназначен для экспорта и импорта данных конфигурации Zabbix.

Доступные методы:

- configuration.export - экспорт конфигурации
- configuration.import - импорт конфигурации
- configuration.importcompare - сравнение файла импорта с текущими элементами системы

#### configuration.export

Описание

`string configuration.export(          )`

Этот метод позволяет экспортировать данные конфигурации в виде сериализованной строки.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие объекты, которые нужно экспортировать, и используемый формат.

| Parameter | Type | Description |
|---|---|---|
| format | string | Формат, в котором должны быть экспортированы данные.<br><br>Возможные значения:<br>`yaml` - YAML;<br>`xml` - XML;<br>`json` - JSON;<br>`raw` - необработанный массив PHP.<br><br>Поведение параметра:<br>- *required* |
| prettyprint | boolean | Сделать вывод более удобочитаемым для человека, добавив отступы.<br><br>Возможные значения:<br>`true` - добавить отступы;<br>`false` - *(по умолчанию)* не добавлять отступы. |
| options | object | Объекты, которые нужно экспортировать.<br><br>Объект `options` имеет следующие параметры:<br>`host_groups` - (`array`) IDs групп узлов сети для экспорта;<br>`hosts` - (`array`) IDs узлов сети для экспорта;<br>`images` - (`array`) IDs изображений для экспорта;<br>`maps` - (`array`) IDs карт для экспорта;<br>`mediaTypes` - (`array`) IDs типов медиа для экспорта;<br>`template_groups` - (`array`) IDs групп шаблонов для экспорта;<br>`templates` - (`array`) IDs шаблонов для экспорта.<br><br>Пользователи типов *Admin* и *User* могут экспортировать только те объекты, для которых у них есть права *read-only* или *read-write* permission, а также изображения, но не типы медиа.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

(`string`) Возвращает сериализованную строку, содержащую запрошенные данные конфигурации.

Примеры

Экспорт шаблона

Экспортируйте конфигурацию шаблона ”10571” в виде XML-строки.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.export",
    "params": {
        "options": {
            "templates": [
                "10571"
            ]
        },
        "format": "xml"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
```

```
    "result": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>7.4</version><template_
    "id": 1
}
```


Источник

CConfiguration::export() в *ui/include/classes/api/services/CConfiguration.php*.

### configuration.import

Описание

`boolean configuration.import(          )`

Этот метод позволяет импортировать данные конфигурации из сериализованной строки.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие данные для импорта, и правила обработки этих данных.

| Parameter | Type | Description |
|---|---|---|
| format | string | Формат сериализованной строки.<br><br>Возможные значения:<br>`yaml` - YAML;<br>`xml` - XML;<br>`json` - JSON.<br><br>Поведение параметра:<br>- *required* |
| source | string | Сериализованная строка, содержащая данные конфигурации.<br><br>Поведение параметра:<br>- *required* |
| rules | object | Правила импорта новых и существующих объектов.<br><br>Пользователи типа *Admin* могут импортировать только те объекты, для которых у них есть разрешение *read-write*, а также карты. Например, узел сети и его сущности (элементы данных, триггеры, графики и т. д.) могут быть импортированы только в том случае, если группа пользователей имеет разрешение на группу узлов сети, к которой будет принадлежать импортируемый узел сети. Изображения и типы оповещений не могут быть импортированы пользователями типа *Admin*.<br><br>Параметр `rules` подробно описан в таблице ниже.<br><br>Поведение параметра:<br>- *required* |

**Attention:**
Если правила не заданы, конфигурация не будет обновлена.

Объект `rules` поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| discoveryRules | object | Правила импорта правил LLD.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые правила LLD; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие правила LLD будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено значение `true`, правила LLD, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| graphs | object | Правила импорта графиков.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые графики; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие графики будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено значение `true`, графики, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| host_groups | object | Правила импорта групп узлов сети.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые группы узлов сети; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие группы узлов сети будут обновлены; по умолчанию: `false`. |
| template_groups | object | Правила импорта групп шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые группы шаблонов; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие группы шаблонов будут обновлены; по умолчанию: `false`. |
| hosts | object | Правила импорта узлов сети.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые узлы сети; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие узлы сети будут обновлены; по умолчанию: `false`. |
| httptests | object | Правила импорта веб-сценариев.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые веб-сценарии; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие веб-сценарии будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено значение `true`, веб-сценарии, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |

| Parameter | Type | Description |
|---|---|---|
| images | object | Правила импорта изображений.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено значение `true`, будут созданы новые изображения; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено значение `true`, существующие изображения будут обновлены; по умолчанию: `false`. |
| items | object | Правила импорта элементов данных.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено значение `true`, будут созданы новые элементы данных; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено значение `true`, существующие элементы данных будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено значение `true`, элементы данных, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| maps | object | Правила импорта карт.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено значение `true`, будут созданы новые карты; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено значение `true`, существующие карты будут обновлены; по умолчанию: `false`. |
| mediaTypes | object | Правила импорта типов оповещений.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено значение `true`, будут созданы новые типы оповещений; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено значение `true`, существующие типы оповещений будут обновлены; по умолчанию: `false`. |
| templateLinkage | object | Правила импорта связей шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено значение `true`, шаблоны, которые не связаны с импортируемым узлом сети или шаблоном, но присутствуют в импортируемых данных, будут связаны; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено значение `true`, шаблоны, которые связаны с импортируемым узлом сети или шаблоном, но отсутствуют в импортируемых данных, будут отвязаны без удаления сущностей (элементов данных, триггеров и т. д.), унаследованных от отвязанных шаблонов; по умолчанию: `false`. |
| templates | object | Правила импорта шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено значение `true`, будут созданы новые шаблоны; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено значение `true`, существующие шаблоны будут обновлены; по умолчанию: `false`. |

| Parameter | Type | Description |
|---|---|---|
| templateDashboards | object | Правила импорта дашбордов шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые дашборды шаблонов; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие дашборды шаблонов будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено значение `true`, дашборды шаблонов, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| triggers | object | Правила импорта триггеров.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые триггеры; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие триггеры будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено значение `true`, триггеры, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| valueMaps | object | Правила импорта карт значений узла сети или шаблона.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено значение `true`, будут созданы новые карты значений; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено значение `true`, существующие карты значений будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено значение `true`, карты значений, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |

Возвращаемые значения

( ) Возвращает `true` при успешном импорте.

Примеры

Импорт шаблона

Импортируйте конфигурацию шаблона, содержащуюся в XML-строке. Если какие-либо элементы данных или триггеры в XML-строке отсутствуют, они будут удалены из базы данных, а все остальное останется без изменений.

Запрос:

```
{
    "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>7.4</version><templ
    },
    "id": 1
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```


Источник

CConfiguration::import() в *ui/include/classes/api/services/CConfiguration.php*.

#### configuration.importcompare

Описание

`array configuration.importcompare(  )`

Этот метод позволяет сравнить файл импорта с текущими элементами системы и показывает, что изменится, если этот файл импорта будет импортирован.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие возможные данные для импорта, и правила обработки этих данных.

| Parameter | Type | Description |
|---|---|---|
| format | string | Формат сериализованной строки.<br><br>Возможные значения:<br>`yaml` - YAML;<br>`xml` - XML;<br>`json` - JSON.<br><br>Поведение параметра:<br>- *required* |
| source | string | Сериализованная строка, содержащая данные конфигурации.<br><br>Поведение параметра:<br>- *required* |

| Parameter | Type | Description |
|---|---|---|
| rules | object | Правила сравнения новых и существующих объектов.<br><br>Пользователи типов *Admin* и *User* могут сравнивать новые объекты только с существующими, для которых у них есть права *read-only* или *read-write* permission.<br><br>Параметр `rules` подробно описан в таблице ниже.<br><br>Поведение параметра:<br>- *required* |

**Attention:**
Если правила не заданы, обновлять будет нечего, и результат будет пустым.

**Note:**
Сравнение будет выполняться только для групп узлов сети и шаблонов. Триггеры и графики будут сравниваться только для импортируемых шаблонов, все остальные будут считаться ”новыми”.

Объект `rules` поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| discoveryRules | object | Правила импорта правил LLD.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые правила LLD будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие правила LLD будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, правила LLD, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| graphs | object | Правила импорта графиков.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые графики будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие графики будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, графики, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| host_groups | object | Правила импорта групп узлов сети.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые группы узлов сети будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие группы узлов сети будут обновлены; по умолчанию: `false`. |
| template_groups | object | Правила импорта групп шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые группы шаблонов будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие группы шаблонов будут обновлены; по умолчанию: `false`. |

| Parameter | Type | Description |
|---|---|---|
| hosts | object | Правила импорта узлов сети.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые узлы сети будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие узлы сети будут обновлены; по умолчанию: `false`.<br><br>Этот параметр не влияет на результат. Он разрешен только для согласованности с `configuration.import`. |
| httptests | object | Правила импорта веб-сценариев.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые веб-сценарии будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие веб-сценарии будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, веб-сценарии, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| images | object | Правила импорта изображений.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые изображения будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие изображения будут обновлены; по умолчанию: `false`.<br><br>Этот параметр не влияет на результат. Он разрешен только для согласованности с `configuration.import`. |
| items | object | Правила импорта элементов данных.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые элементы данных будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие элементы данных будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, элементы данных, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| maps | object | Правила импорта карт сети.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые карты сети будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие карты сети будут обновлены; по умолчанию: `false`.<br><br>Этот параметр не влияет на результат. Он разрешен только для согласованности с `configuration.import`. |

| Parameter | Type | Description |
|---|---|---|
| mediaTypes | object | Правила импорта типов оповещений.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые типы оповещений будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие типы оповещений будут обновлены; по умолчанию: `false`.<br><br>Этот параметр не влияет на результат. Он разрешен только для согласованности с `configuration.import`. |
| templateLinkage | object | Правила импорта связей шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, шаблоны, которые не связаны с импортируемым узлом сети или шаблоном, но присутствуют в импортируемых данных, будут связаны; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, шаблоны, которые связаны с импортируемым узлом сети или шаблоном, но отсутствуют в импортируемых данных, будут отвязаны без удаления сущностей (элементов данных, триггеров и т. д.), унаследованных от отвязанных шаблонов; по умолчанию: `false`. |
| templates | object | Правила импорта шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые шаблоны будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие шаблоны будут обновлены; по умолчанию: `false`. |
| templateDashboards | object | Правила импорта дашбордов шаблонов.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые дашборды шаблонов будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие дашборды шаблонов будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, дашборды шаблонов, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |
| triggers | object | Правила импорта триггеров.<br><br>Поддерживаемые параметры:<br>`createMissing` - `(boolean)` если установлено в `true`, новые триггеры будут созданы; по умолчанию: `false`;<br>`updateExisting` - `(boolean)` если установлено в `true`, существующие триггеры будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - `(boolean)` если установлено в `true`, триггеры, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |

| Parameter | Type | Description |
|---|---|---|
| valueMaps | object | Правила импорта карт значений узла сети или шаблона.<br><br>Поддерживаемые параметры:<br>`createMissing` - (`boolean`) если установлено в `true`, новые карты значений будут созданы; по умолчанию: `false`;<br>`updateExisting` - (`boolean`) если установлено в `true`, существующие карты значений будут обновлены; по умолчанию: `false`;<br>`deleteMissing` - (`boolean`) если установлено в `true`, карты значений, отсутствующие в импортируемых данных, будут удалены из базы данных; по умолчанию: `false`. |

Возвращаемые значения

(`array`) Возвращает массив с изменениями в конфигурации, которые будут сделаны.

Примеры

Сравнение импорта шаблона

Сравните шаблон, содержащийся в XML-строке, с текущими элементами системы и покажите, что будет изменено, если этот шаблон будет импортирован.

Запрос:

```
{
    "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>7.4</version><templ
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Источник

CConfiguration::importcompare() в *ui/include/classes/api/services/CConfiguration.php*.

### Корреляция

Этот класс предназначен для работы с корреляциями.

Ссылки на объекты:

- Корреляция
- Операция корреляции
- Фильтр корреляции
  - Условие фильтра корреляции

Доступные методы:

- correlation.create - создать новые корреляции
- correlation.delete - удалить корреляции
- correlation.get - получить корреляции
- correlation.update - обновить корреляции

### Объект корреляции

Следующие объекты напрямую связаны с API корреляций (`correlation`).

Корреляция

Объект корреляции имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| correlationid | ID | ID корреляции.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя корреляции.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| description | string | Описание корреляции. |
| status | integer | Включена корреляция или отключена.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включена;<br>1 - отключена. |

Операция корреляции

Объект операции корреляции определяет операцию, которая будет выполнена при запуске корреляции. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип операции.<br><br>Возможные значения:<br>0 - закрыть старые события;<br>1 - закрыть новое событие.<br><br>Поведение свойства:<br>- *required* |

Фильтр корреляции

Объект фильтра корреляции определяет набор условий, которые должны быть выполнены для выполнения настроенных операций корреляции. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| conditions | array | Набор условий фильтра для использования при фильтрации результатов. Условия будут отсортированы в порядке их размещения в формуле.<br><br>Поведение свойства:<br>- *required* |
| evaltype | integer | Метод оценки условий фильтра.<br><br>Возможные значения:<br>0 - And/Or;<br>1 - And;<br>2 - Or;<br>3 - Custom expression.<br><br>Поведение свойства:<br>- *required* |
| eval_formula | string | Сгенерированное выражение, которое будет использоваться для оценки условий фильтра. Выражение содержит идентификаторы, которые ссылаются на конкретные условия фильтра по их `formulaid`. Значение `eval_formula` равно значению `formula` для фильтров с пользовательским выражением.<br><br>Поведение свойства:<br>- *read-only* |
| formula | string | Определяемое пользователем выражение, которое будет использоваться для оценки условий фильтров с пользовательским выражением. Выражение должно содержать идентификаторы, которые ссылаются на конкретные условия фильтра по их `formulaid`. Идентификаторы, используемые в выражении, должны в точности совпадать с теми, что определены в условиях фильтра: ни одно условие не может остаться неиспользованным или опущенным.<br><br>Поведение свойства:<br>- *required* если `evaltype` установлен в значение ”custom expression” |

Условие фильтра корреляции

Объект условия фильтра корреляции определяет конкретное условие, которое должно быть проверено перед выполнением операций корреляции.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип условия.<br><br>Возможные значения:<br>0 - старый тег события;<br>1 - новый тег события;<br>2 - новая группа узлов сети события;<br>3 - пара тегов события;<br>4 - значение старого тега события;<br>5 - значение нового тега события.<br><br>Поведение свойства:<br>- *required* |
| tag | string | Тег события (старый или новый).<br><br>Поведение свойства:<br>- *required* если `type` установлено в ”old event tag”, ”new event tag”, ”old event tag value” или ”new event tag value” |

| Property | Type | Description |
|---|---|---|
| groupid | ID | ID группы узлов сети.<br><br>Поведение свойства:<br>- *required* если `type` установлено в ”new event host group” |
| oldtag | string | Старый тег события.<br><br>Поведение свойства:<br>- *required* если `type` установлено в ”event tag pair” |
| newtag | string | Старый тег события.<br><br>Поведение свойства:<br>- *required* если `type` установлено в ”event tag pair” |
| value | string | Значение тега события (старого или нового).<br><br>Поведение свойства:<br>- *required* если `type` установлено в ”old event tag value” или ”new event tag value” |
| formulaid | string | Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но будет сгенерирован заново при последующем запросе. |
| operator | integer | Оператор условия.<br><br>Поведение свойства:<br>- *required* если `type` установлено в ”new event host group”, ”old event tag value” или ”new event tag value” |

**Note:**
Чтобы лучше понять, как использовать фильтры с различными типами выражений, см. примеры на страницах методов correlation.get и correlation.create.

Для каждого типа условия поддерживаются следующие операторы и значения.

| Condition | Condition name | Supported operators | Expected value |
|---|---|---|---|
| 2 | Группа узлов сети | =, <> | ID группы узлов сети. |
| 4 | Значение старого тега события | =, <>, like, not like | string |
| 5 | Значение нового тега события | =, <>, like, not like | string |

correlation.create

Описание

`object correlation.create( / correlations)`

Этот метод позволяет создавать новые корреляции.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Корреляции для создания.

Помимо стандартных свойств корреляции, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| operations | array | Операции корреляции для создания корреляции.<br><br>Поведение параметра:<br>- *required* |
| filter | object | Объект фильтра корреляции для корреляции.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

(object) Возвращает объект, который содержит ID созданных корреляций под свойством correlationids. Порядок возвращаемых ID совпадает с порядком переданных корреляций.

Примеры

Создание новой корреляции тегов событий

Создайте корреляцию, используя метод оценки AND/OR с одним условием и одной операцией. По умолчанию корреляция будет включена.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Использование пользовательского фильтра выражения

Создайте корреляцию, которая будет использовать пользовательское условие фильтра. Идентификаторы формулы "A" или "B" выбраны произвольно. Тип условия будет "Группа узлов сети" с оператором "<>".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "2"
        ]
    },
    "id": 1
}
```

См. также

- Фильтр корреляции
- Операция корреляции

Источник

CCorrelation::create() в *ui/include/classes/api/services/CCorrelation.php*.

**correlation.delete**

Описание

`object correlation.delete(    correlationids)`

Этот метод позволяет удалять корреляции.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(  ) ID удаляемых корреляций.

Возвращаемые значения

(  ) Возвращает объект, который содержит ID удаленных корреляций, переданных в свойстве `correlationids`.

Пример

Удаление нескольких корреляций

Удаление двух корреляций.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Источник

CCorrelation::delete() в *ui/include/classes/api/services/CCorrelation.php*.

**correlation.get**

Описание

`integer/array correlation.get(  )`

Этот метод позволяет получать корреляции в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие желаемый результат.

Метод поддерживает следующие параметры.

| Параметр | Тип | Описание |
|---|---|---|
| correlationids | ID/array | Возвращать только корреляции с указанными ID. |
| selectFilter | query | Возвращать свойство `filter` с условиями корреляции. |
| selectOperations | query | Возвращать свойство `operations` с операциями корреляции. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `correlationid`, `name`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |

| Параметр | Тип | Описание |
|---|---|---|
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение корреляций

Получение всех настроенных корреляций вместе с условиями и операциями корреляции. Фильтр использует тип вычисления "and/or", поэтому свойство `formula` пустое, а `eval_formula` генерируется автоматически.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectFilter": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Фильтр корреляции
- Операция корреляции

Источник

CCorrelation::get() в *ui/include/classes/api/services/CCorrelation.php*.

**correlation.update**

Описание

`object correlation.update( / correlations)`

Этот метод позволяет обновить существующие корреляции.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства корреляции, которые необходимо обновить.

Для каждой корреляции должно быть определено свойство `correlationid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств корреляции, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| filter | object | Объект фильтра корреляции для замены текущего фильтра. |
| operations | array | Операции корреляции для замены существующих операций. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных корреляций в свойстве `correlationid`.

Примеры

Отключение корреляции

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "status": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
```

```
            "1"
        ]
    },
    "id": 1
}
```

Заменить условия, но сохранить метод оценки

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "filter": {
            "conditions": [
                {
                    "type": 3,
                    "oldtag": "error",
                    "newtag": "ok"
                }
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

См. также

- Фильтр корреляции
- Операция корреляции

Источник

Correlation::update() в *ui/include/classes/api/services/CCorrelation.php*.

#### Модуль

Этот класс предназначен для работы с модулями веб-интерфейса.

Ссылки на объекты:

- Module

Доступные методы:

- module.create - установить новые модули
- module.delete - удалить модули
- module.get - получить модули
- module.update - обновить модули

#### Объект module

Следующие объекты напрямую связаны с API `module`.

Модуль

Объект module имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| moduleid | ID | ID модуля, как он хранится в базе данных.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| id | string | Уникальный ID модуля, определенный разработчиком в файле manifest.json модуля.<br><br>Possible values for built-in modules:<br>see property "type" description in Dashboard widget.<br><br>Property behavior:<br>- *required* for create operations |
| relative_path | string | Путь к каталогу модуля относительно каталога веб-интерфейса Zabbix.<br><br>Possible values:<br>`widgets/*` - для встроенных модулей виджетов;<br>`modules/*` - для сторонних модулей.<br><br>Property behavior:<br>- *required* for create operations |
| status | integer | Включен модуль или отключен.<br><br>Possible values:<br>0 - *(default)* Disabled;<br>1 - Enabled. |
| config | object | Конфигурация модуля. |

**module.create**

Описание

`object module.create( / modules)`

Этот метод позволяет устанавливать новые интерфейсные модули.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

**Attention:**
Файлы модулей необходимо распаковать вручную в правильные подкаталоги, соответствующие свойству `relative_path` модулей.

Параметры

`(object/array)` Модули для создания.

Метод принимает модули со стандартными свойствами модуля.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы установленных модулей в свойстве `moduleids`. Порядок возвращаемых идентификаторов соответствует порядку переданных модулей.

Примеры

Установка модуля

Установите модуль со статусом ”Enabled”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "module.create",
    "params": {
        "id": "example_module",
        "relative_path": "modules/example_module",
        "status": 1
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

См. также

- Модуль
- Модули веб-интерфейса

Исходный код

CModule::create() в *ui/include/classes/api/services/CModule.php*.

**module.delete**

Описание

`object module.delete(    moduleids)`

Этот метод позволяет удалять модули.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

**Attention:**
Файлы модуля необходимо удалять вручную.

Параметры

`(array)` Идентификаторы модулей для удаления.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных модулей в свойстве `moduleids`.

Примеры

Удаление нескольких модулей

Удалите модули ”27” и ”28”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "module.delete",
    "params": [
        "27",
        "28"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "27",
            "28"
        ]
    },
    "id": 1
}
```

Источник

CModule::delete() в *ui/include/classes/api/services/CModule.php*.

**module.get**

Описание

`integer/array module.get(object parameters)`

Метод позволяет получать модули в соответствии с заданными параметрами.

**Note:**
Этот метод доступен только пользователю типа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительные сведения см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| moduleids | ID/array | Возвращать только модули с указанными ID. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `moduleid`, `relative_path`. |
| countOutput | boolean | Эти параметры описаны в Reference commentary. |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение модуля по ID

Получить все данные о модулях "1", "2" и "25".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "module.get",
    "params": {
        "output": "extend",
        "moduleids": [
            "1",
            "2",
            "25"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Модуль
- Виджет панели мониторинга
- Модули веб-интерфейса

Источник

CModule::get() в *ui/include/classes/api/services/CModule.php*.

**module.update**

Описание

`object module.update( / modules)`

Этот метод позволяет обновлять существующие модули.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства модуля, которые необходимо обновить.

Свойство `moduleid` должно быть определено для каждого модуля, все остальные свойства необязательны. Будут обновлены только указанные свойства.

Метод принимает модули со стандартными свойствами модуля.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных модулей в свойстве `moduleids`.

Примеры

Отключение модуля

Отключить модуль "25".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "module.update",
    "params": {
        "moduleid": "25",
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

См. также

- Модуль
- Модули веб-интерфейса

Источник

CModule::update() в *ui/include/classes/api/services/CModule.php*.

**Настройки**

Этот класс предназначен для работы с общими административными настройками.

Ссылки на объекты:

- Settings

Доступные методы:

- settings.get - получить настройки

- settings.update - обновить настройки

**Объект settings**

Следующие объекты напрямую связаны с API `settings`.

Настройки

Объект settings имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| default_lang | string | Язык системы по умолчанию.<br><br>По умолчанию: `en_US`. |
| default_timezone | string | Часовой пояс системы по умолчанию.<br><br>По умолчанию: `system` - системное значение по умолчанию.<br><br>Полный список поддерживаемых часовых поясов см. в документации PHP. |
| default_theme | string | Тема по умолчанию.<br><br>Возможные значения:<br>`blue-theme` - *(по умолчанию)* Синяя;<br>`dark-theme` - Темная;<br>`hc-light` - Светлая с высокой контрастностью;<br>`hc-dark` - Темная с высокой контрастностью. |
| search_limit | integer | Ограничение для результатов поиска и фильтрации.<br><br>По умолчанию: 1000. |
| max_overview_table_size | integer | Максимальное число столбцов и строк в виджетах панели Data overview и Trigger overview.<br><br>По умолчанию: 50. |
| max_in_table | integer | Максимальное количество элементов, отображаемых внутри ячейки таблицы.<br><br>По умолчанию: 50. |
| server_check_interval | integer | Показывать предупреждение, если сервер Zabbix недоступен.<br><br>Возможные значения:<br>0 - Не показывать предупреждение;<br>10 - *(по умолчанию)* Показывать предупреждение. |
| work_period | string | Рабочее время.<br><br>По умолчанию: 1-5,09:00-18:00. |

| Property | Type | Description |
|---|---|---|
| show_technical_errors | integer | Показывать технические ошибки (PHP/SQL) пользователям, не являющимся супер-администраторами, а также пользователям, не входящим в группы пользователей с включенным режимом отладки.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не показывать технические ошибки;<br>1 - Показывать технические ошибки. |
| history_period | string | Максимальный период отображения исторических данных в виджетах Latest data, Web и Data overview. Принимает секунды и единицы времени с суффиксом.<br><br>По умолчанию: 24h. |
| period_default | string | Период фильтра времени по умолчанию.<br>Принимает секунды и единицы времени с суффиксом с поддержкой месяца и года (30s, 1m, 2h, 1d, 1M, 1y).<br><br>По умолчанию: 1h. |
| max_period | string | Максимальный период для фильтра времени.<br>Принимает секунды и единицы времени с суффиксом с поддержкой месяца и года (30s, 1m, 2h, 1d, 1M, 1y).<br><br>По умолчанию: 2y. |
| severity_color_0 | string | Цвет для уровня ”Not classified” в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: 97AAB3. |
| severity_color_1 | string | Цвет для уровня ”Information” в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: 7499FF. |
| severity_color_2 | string | Цвет для уровня ”Warning” в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: FFC859. |
| severity_color_3 | string | Цвет для уровня ”Average” в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: FFA059. |
| severity_color_4 | string | Цвет для уровня ”High” в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: E97659. |
| severity_color_5 | string | Цвет для уровня ”Disaster” в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: E45959. |

| Property | Type | Description |
|---|---|---|
| severity_name_0 | string | Название для уровня ”Not classified”.<br><br>По умолчанию: Not classified. |
| severity_name_1 | string | Название для уровня ”Information”.<br><br>По умолчанию: Information. |
| severity_name_2 | string | Название для уровня ”Warning”.<br><br>По умолчанию: Warning. |
| severity_name_3 | string | Название для уровня ”Average”.<br><br>По умолчанию: Average. |
| severity_name_4 | string | Название для уровня ”High”.<br><br>По умолчанию: High. |
| severity_name_5 | string | Название для уровня ”Disaster”.<br><br>По умолчанию: Disaster. |
| custom_color | integer | Использовать пользовательские цвета статуса событий.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не использовать пользовательские цвета статуса событий;<br>1 - Использовать пользовательские цвета статуса событий. |
| ok_period | string | Отображать период для триггеров OK.<br>Принимает секунды и единицы времени с суффиксом.<br><br>По умолчанию: 5m. |
| blink_period | string | Период мигания триггеров при изменении статуса.<br>Принимает секунды и единицы времени с суффиксом.<br><br>По умолчанию: 2m. |
| problem_unack_color | string | Цвет для неподтвержденных событий PROBLEM в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: CC0000. |
| problem_ack_color | string | Цвет для подтвержденных событий PROBLEM в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: CC0000. |
| ok_unack_color | string | Цвет для неподтвержденных событий RESOLVED в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: 009900. |
| ok_ack_color | string | Цвет для подтвержденных событий RESOLVED в виде шестнадцатеричного кода цвета.<br><br>По умолчанию: 009900. |

| Property | Type | Description |
|---|---|---|
| problem_unack_style | integer | Мигание для неподтвержденных событий PROBLEM.<br><br>Возможные значения:<br>0 - Не показывать мигание;<br>1 - *(по умолчанию)* Показывать мигание. |
| problem_ack_style | integer | Мигание для подтвержденных событий PROBLEM.<br><br>Возможные значения:<br>0 - Не показывать мигание;<br>1 - *(по умолчанию)* Показывать мигание. |
| ok_unack_style | integer | Мигание для неподтвержденных событий RESOLVED.<br><br>Возможные значения:<br>0 - Не показывать мигание;<br>1 - *(по умолчанию)* Показывать мигание. |
| ok_ack_style | integer | Мигание для подтвержденных событий RESOLVED.<br><br>Возможные значения:<br>0 - Не показывать мигание;<br>1 - *(по умолчанию)* Показывать мигание. |
| url | string | URL веб-интерфейса. |
| discovery_groupid | ID | ID группы узлов сети, в которую будут автоматически помещаться обнаруженные узлы сети. |
| default_inventory_mode | integer | Режим инвентаризации узла сети по умолчанию.<br><br>Возможные значения:<br>-1 - *(по умолчанию)* Отключено;<br>0 - Вручную;<br>1 - Автоматически. |
| alert_usrgrpid | ID | ID группы пользователей, которой будет отправлено сообщение тревоги о недоступности базы данных.<br><br>Если установлено значение ”0”, сообщение тревоги отправляться не будет. |
| snmptrap_logging | integer | Журналировать несопоставленные SNMP traps.<br><br>Возможные значения:<br>0 - Не журналировать несопоставленные SNMP traps;<br>1 - *(по умолчанию)* Журналировать несопоставленные SNMP traps. |
| login_attempts | integer | Количество неудачных попыток входа, после которого форма входа будет заблокирована.<br><br>По умолчанию: 5. |

| Property | Type | Description |
|---|---|---|
| login_block | string | Интервал времени, в течение которого форма входа будет заблокирована, если количество неудачных попыток входа превысит значение, заданное в поле login_attempts.<br>Принимает секунды и единицы времени с суффиксом.<br><br>По умолчанию: 30s. |
| validate_uri_schemes | integer | Проверять схемы URI.<br><br>Возможные значения:<br>0 - Не проверять;<br>1 - *(по умолчанию)* Проверять. |
| uri_valid_schemes | string | Допустимые схемы URI.<br><br>По умолчанию: http,https,ftp,file,mailto,tel,ssh. |
| x_frame_options | string | HTTP-заголовок X-Frame-Options.<br><br>По умолчанию: SAMEORIGIN. |
| iframe_sandboxing_enabled | integer | Использовать sandboxing для iframe.<br><br>Возможные значения:<br>0 - Не использовать;<br>1 - *(по умолчанию)* Использовать. |
| iframe_sandboxing_exceptions | string | Исключения для sandboxing iframe. |
| connect_timeout | string | Тайм-аут соединения с сервером Zabbix.<br><br>Диапазон возможных значений: 1-30s.<br><br>По умолчанию: 3s.<br><br>Поведение свойства:<br>- *required* |
| socket_timeout | string | Тайм-аут сети по умолчанию.<br><br>Диапазон возможных значений: 1-300s.<br><br>По умолчанию: 3s.<br><br>Поведение свойства:<br>- *required* |
| media_type_test_timeout | string | Тайм-аут сети для проверки типа медиа.<br><br>Диапазон возможных значений: 1-300s.<br><br>По умолчанию: 65s.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| item_test_timeout | string | Тайм-аут сети для тестов элемента данных.<br><br>Диапазон возможных значений: 1-600s.<br><br>По умолчанию: 60s.<br><br>Поведение свойства:<br>- *required* |
| script_timeout | string | Тайм-аут сети для выполнения скрипта.<br><br>Диапазон возможных значений: 1-300s.<br><br>По умолчанию: 60s.<br><br>Поведение свойства:<br>- *required* |
| report_test_timeout | string | Тайм-аут сети для теста запланированного отчета.<br><br>Диапазон возможных значений: 1-300s.<br><br>По умолчанию: 60s.<br><br>Поведение свойства:<br>- *required* |
| auditlog_enabled | integer | Включить ли аудит логирования.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| auditlog_mode | integer | Включить ли аудит логирования действий low-level discovery, network discovery и autoregistration, выполняемых сервером (пользователем System).<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| ha_failover_delay | string | Задержка переключения в секундах.<br><br>По умолчанию: 1m.<br><br>Поведение свойства:<br>- *read-only* |

| Property | Type | Description |
|---|---|---|
| geomaps_tile_provider | string | Поставщик тайлов геокарт.<br><br>Возможные значения:<br>`OpenStreetMap.Mapnik` - *(по умолчанию)* OpenStreetMap Mapnik;<br>`OpenTopoMap` - OpenTopoMap;<br>`Stamen.TonerLite` - Stamen Toner Lite;<br>`Stamen.Terrain` - Stamen Terrain;<br>`USGS.USTopo` - USGS US Topo;<br>`USGS.USImagery` - USGS US Imagery.<br><br>Поддерживает пустую строку для указания пользовательских значений `geomaps_tile_url`, `geomaps_max_zoom` и `geomaps_attribution`. |
| geomaps_tile_url | string | URL тайлов геокарт.<br><br>Поведение свойства:<br>- *supported* если `geomaps_tile_provider` установлен в пустую строку |
| geomaps_max_zoom | integer | Максимальный уровень масштабирования геокарт.<br><br>Возможные значения: 0-30.<br><br>Поведение свойства:<br>- *supported* если `geomaps_tile_provider` установлен в пустую строку |
| geomaps_attribution | string | Текст атрибуции геокарт.<br><br>Поведение свойства:<br>- *supported* если `geomaps_tile_provider` установлен в пустую строку |
| vault_provider | integer | Поставщик Vault.<br><br>Возможные значения:<br>0 - *(по умолчанию)* HashiCorp Vault;<br>1 - CyberArk Vault. |
| proxy_secrets_provider | integer | Разрешать значения секретных макросов следующим образом:<br><br>Возможные значения:<br>0 - *(по умолчанию)* только сервер;<br>1 - сервер и прокси независимо. |

| Property | Type | Description |
|---|---|---|
| timeout_zabbix_agent | string | Тратить на обработку не более `timeout_*` секунд.<br>Принимает секунды или единицы времени с суффиксом (например, 30s, 1m). Также принимает пользовательские макросы.<br><br>Диапазон возможных значений: 1-600s.<br><br>По умолчанию: 3s.<br>По умолчанию для `timeout_browser`: 60s.<br><br>Поведение свойства:<br>- *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 | | |

**settings.get**

Описание

`object settings.get(  )`

Метод позволяет получить объект настроек по заданным параметры.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает только один параметр.

| Parameter | Type | Description |
|---|---|---|
| output | query | Этот параметр описан в справочном комментарии. |

Возвращаемые значения

`(object)` Возвращает объект настроек..

Примеры

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "settings.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Ответ:


```
{
    "jsonrpc": "2.0",
    "result": {
        "default_theme": "blue-theme",
        "search_limit": "1000",
        "max_in_table": "50",
        "server_check_interval": "10",
        "work_period": "1-5,09:00-18:00",
        "show_technical_errors": "0",
        "history_period": "24h",
        "period_default": "1h",
        "max_period": "2y",
        "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": "           ",
        "severity_name_1": "      ",
        "severity_name_2": "        ",
        "severity_name_3": "    ",
        "severity_name_4": "    ",
        "severity_name_5": "       ",
        "custom_color": "0",
        "ok_period": "5m",
        "blink_period": "2m",
        "problem_unack_color": "CC0000",
        "problem_ack_color": "CC0000",
        "ok_unack_color": "009900",
        "ok_ack_color": "009900",
        "problem_unack_style": "1",
        "problem_ack_style": "1",
        "ok_unack_style": "1",
        "ok_ack_style": "1",
        "discovery_groupid": "5",
        "default_inventory_mode": "-1",
        "alert_usrgrpid": "7",
        "snmptrap_logging": "1",
        "default_lang": "en_US",
        "default_timezone": "system",
        "login_attempts": "5",
        "login_block": "30s",
        "validate_uri_schemes": "1",
        "uri_valid_schemes": "http,https,ftp,file,mailto,tel,ssh",
        "x_frame_options": "SAMEORIGIN",
        "iframe_sandboxing_enabled": "1",
        "iframe_sandboxing_exceptions": "",
        "max_overview_table_size": "50",
        "connect_timeout": "3s",
        "socket_timeout": "3s",
        "media_type_test_timeout": "65s",
        "script_timeout": "60s",
        "item_test_timeout": "60s",
        "url": "",
        "report_test_timeout": "60s",
        "auditlog_enabled": "1",
        "auditlog_mode": "1",
        "ha_failover_delay": "1m",
        "geomaps_tile_provider": "OpenStreetMap.Mapnik",
        "geomaps_tile_url": "",
```

```
        "geomaps_max_zoom": "0",
        "geomaps_attribution": "",
        "vault_provider": "0",
        "timeout_zabbix_agent": "3s",
        "timeout_simple_check": "3s",
        "timeout_snmp_agent": "3s",
        "timeout_external_check": "3s",
        "timeout_db_monitor": "3s",
        "timeout_http_agent": "3s",
        "timeout_ssh_agent": "3s",
        "timeout_telnet_agent": "3s",
        "timeout_script": "3s"
    },
    "id": 1
}
```


Источник

CSettings::get() в *ui/include/classes/api/services/CSettings.php*.

**settings.update**

Описание

`object settings.update( )`

Этот метод позволяет обновить существующие общие настройки.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Свойства настроек, которые необходимо обновить.

Возвращаемые значения

`(array)` Возвращает массив с именами обновленных параметров.

Примеры

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "settings.update",
    "params": {
        "login_attempts": "1",
        "login_block": "1m"
    },
    "id": 1
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": [
        "login_attempts",
        "login_block"
    ],
    "id": 1
}
```


Источник

CSettings::update() в *ui/include/classes/api/services/CSettings.php*.

#### Обнаруженная служба

Этот класс предназначен для работы с обнаруженными службами.

Ссылки на объекты:

- Обнаруженная служба

Доступные методы:

- dservice.get - получить обнаруженные службы

#### Обнаруженный объект службы

Следующие объекты напрямую связаны с API `dservice`.

Обнаруженная служба

> **Note:**
> Обнаруженные службы создаются сервером Zabbix и не могут быть изменены через API.

Объект обнаруженной службы содержит информацию о службе, обнаруженной правилом сетевого обнаружения на узле сети. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dserviceid | ID | ID обнаруженной службы. |
| dcheckid | ID | ID проверки обнаружения, использованной для обнаружения службы. |
| dhostid | ID | ID обнаруженного узла сети, на котором работает служба. |
| dns | string | DNS узла сети, на котором работает служба. |
| ip | string | IP-адрес узла сети, на котором работает служба. |
| lastdown | timestamp | Время, когда обнаруженная служба в последний раз была недоступна. |
| lastup | timestamp | Время, когда обнаруженная служба в последний раз была доступна. |
| port | integer | Номер порта службы. |
| status | integer | Состояние службы.<br><br>Возможные значения:<br>0 - служба доступна;<br>1 - служба недоступна. |
| value | string | Значение, возвращаемое службой при выполнении проверки обнаружения Zabbix agent, SNMPv1, SNMPv2 или SNMPv3. |

#### dservice.get

Описание

`integer/array dservice.get(          )`

Этот метод позволяет получать обнаруженные сервисы в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dserviceids | ID/array | Возвращать только обнаруженные службы с указанными ID. |
| dhostids | ID/array | Возвращать только обнаруженные службы, которые принадлежат указанным обнаруженным узлам сети. |
| dcheckids | ID/array | Возвращать только обнаруженные службы, которые были обнаружены указанными проверками обнаружения. |
| druleids | ID/array | Возвращать только обнаруженные службы, которые были обнаружены указанными правилами обнаружения. |
| selectDRules | query | Возвращать свойство `drules` с массивом правил обнаружения, которые обнаружили службу. |
| selectDHosts | query | Возвращать свойство `dhosts` с массивом обнаруженных узлов сети, к которым принадлежит служба. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети с тем же IP-адресом и прокси, что и у службы.<br><br>Поддерживает `count`. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам:<br>`selectHosts` - результат будет отсортирован по `hostid`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `dserviceid`, `dhostid`, `ip`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получить службы, обнаруженные на узле сети

Получить все обнаруженные службы, выявленные на обнаруженном узле сети ”11”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dservice.get",
    "params": {
        "output": "extend",
        "dhostids": "11"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


См. также

- Обнаруженный узел сети
- Проверка обнаружения
- Узел сети

Источник

CDService::get() в *ui/include/classes/api/services/CDService.php*.

#### Обнаруженный узел сети

Этот класс предназначен для работы с обнаруженными узлами сети.

Ссылки на объекты:

- Обнаруженный узел сети

Доступные методы:

- dhost.get - получить обнаруженные узлы сети

#### Объект обнаруженного узла сети

Следующие объекты напрямую связаны с API `dhost`.

Обнаруженный узел сети

> **Note:**
> Обнаруженные узлы сети создаются сервером Zabbix и не могут быть изменены через API.

Объект обнаруженного узла сети содержит информацию об узле сети, обнаруженном правилом сетевого обнаружения. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dhostid | ID | ID обнаруженного узла сети. |
| druleid | ID | ID правила обнаружения, которое обнаружило узел сети. |

| Property | Type | Description |
|---|---|---|
| lastdown | timestamp | Время, когда обнаруженный узел сети в последний раз переходил в состояние недоступности. |
| lastup | timestamp | Время, когда обнаруженный узел сети в последний раз переходил в состояние доступности. |
| status | integer | Указывает, доступен обнаруженный узел сети или нет. Узел сети считается доступным, если у него есть хотя бы один активный обнаруженный сервис.<br><br>Возможные значения:<br>0 - узел сети доступен;<br>1 - узел сети недоступен. |

**dhost.get**

Описание

`integer/array dhost.get(  )`

Метод позволяет получить обнаруженные узлы сети по заданным параметрам.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dhostids | ID/array | Возвращать только обнаруженные узлы сети с указанными ID. |
| druleids | ID/array | Возвращать только обнаруженные узлы сети, созданные указанными правилами обнаружения. |
| dserviceids | ID/array | Возвращать только обнаруженные узлы сети, на которых запущены указанные службы. |
| selectDRules | query | Возвращать свойство `drules` с массивом правил обнаружения, которые обнаружили узел сети. |
| selectDServices | query | Возвращать свойство `dservices` с обнаруженными службами, запущенными на узле сети.<br><br>Поддерживает `count`. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам:<br>`selectDServices` - результаты будут отсортированы по `dserviceid`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `dhostid`, `druleid`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |

| Parameter | Type | Description |
|---|---|---|
| startSearch | boolean | |

Возвращаемые значения

(integer/array) Возвращает:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр countOutput.

Примеры

Получение обнаруженных узлов сети по правилу обнаружения

Получить все узлы сети и обнаруженные службы, которые они запускают и которые были обнаружены правилом обнаружения "4".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dhost.get",
    "params": {
        "output": "extend",
        "selectDServices": "extend",
        "druleids": "4"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Обнаруженная служба
- Правило обнаружения

Источник

CDHost::get() in *ui/include/classes/api/services/CDHost.php*.

#### Обслуживание

Этот класс предназначен для работы с обслуживанием.

Ссылки на объекты:

- Обслуживание

Доступные методы:

- housekeeping.get - получить данные обслуживания
- housekeeping.update - обновить данные обслуживания

#### Объект housekeeping

Следующие объекты напрямую связаны с API `housekeeping`.

Обслуживание

Объект настроек имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| hk_events_mode | integer | Включить внутреннее обслуживание для событий и оповещений.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| hk_events_trigger | string | Период хранения данных триггеров. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 365d. |
| hk_events_service | string | Период хранения данных сервисов. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 1d. |
| hk_events_internal | string | Период хранения внутренних данных. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 1d. |
| hk_events_discovery | string | Период хранения данных обнаружения сети. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 1d. |
| hk_events_autoreg | string | Период хранения данных авто-регистрации. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 1d. |
| hk_services_mode | integer | Включить внутреннее обслуживание для сервисов.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |

| Property | Type | Description |
|---|---|---|
| hk_services | string | Период хранения данных сервисов. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 365d. |
| hk_audit_mode | integer | Включить внутреннее обслуживание для аудита.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| hk_audit | string | Период хранения данных аудита. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 31d. |
| hk_sessions_mode | integer | Включить внутреннее обслуживание для сеансов.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| hk_sessions | string | Период хранения данных сеансов. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 31d. |
| hk_history_mode | integer | Включить внутреннее обслуживание для истории.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| hk_history_global | integer | Переопределить период истории элемента данных.<br><br>Возможные значения:<br>0 - Не переопределять;<br>1 - *(по умолчанию)* Переопределить. |
| hk_history | string | Период хранения данных истории. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 31d. |
| hk_trends_mode | integer | Включить внутреннее обслуживание для трендов.<br><br>Возможные значения:<br>0 - Отключить;<br>1 - *(по умолчанию)* Включить. |
| hk_trends_global | integer | Переопределить период трендов элемента данных.<br><br>Возможные значения:<br>0 - Не переопределять;<br>1 - *(по умолчанию)* Переопределить. |
| hk_trends | string | Период хранения данных трендов. Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 365d. |
| db_extension | string | Флаг конфигурации расширения БД. Если для этого флага задано значение ”timescaledb”, сервер изменяет свое поведение для обслуживания и удаления элементов данных.<br><br>Поведение свойства:<br>- *только для чтения* |

| Property | Type | Description |
|---|---|---|
| compression_availability | integer | Поддерживается ли сжатие данных базой данных (или ее расширением).<br><br>Возможные значения:<br>0 - Недоступно;<br>1 - Доступно.<br><br>Поведение свойства:<br>- *только для чтения* |
| compression_status | integer | Включить сжатие TimescaleDB для истории и трендов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Выкл.;<br>1 - Вкл. |
| compress_older | string | Сжимать записи истории и трендов, старше указанного периода.<br>Принимает секунды и единицу времени с суффиксом.<br><br>По умолчанию: 7d. |

**housekeeping.get**

Описание

`object housekeeping.get( )`

Метод позволяет получить объект обслуживания в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает только один параметр.

| Parameter | Type | Description |
|---|---|---|
| output | query | Этот параметр описан в справочном комментарии. |

Возвращаемые значения

`(object)` Возвращает объект обслуживания данных.

Примеры

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "housekeeping.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Источник

CHousekeeping ::get() в *ui/include/classes/api/services/CHousekeeping.php*.

**housekeeping.update**

Описание

`object housekeeping.update(  )`

Этот метод позволяет обновить существующие настройки управления.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Свойства очистки, которые необходимо обновить.

Возвращаемые значения

`(array)` Возвращает массив с именами обновленных параметров.

Примеры

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        "hk_events_mode",
        "hk_events_trigger",
        "hk_events_internal",
        "hk_events_discovery"
    ],
    "id": 1
}
```

Источник

CHousekeeping::update() в *ui/include/classes/api/services/CHousekeeping.php*.

### Обслуживание

Этот класс предназначен для работы с обслуживанием.

Ссылки на объекты:

- Обслуживание
- Период времени
- Тег проблемы

Доступные методы:

- maintenance.create - создать новые обслуживания
- maintenance.delete - удалить обслуживания
- maintenance.get - получить обслуживания
- maintenance.update - обновить обслуживания

### Объект обслуживания

Следующие объекты напрямую связаны с API `maintenance`.

Обслуживание

Объект обслуживания имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| maintenanceid | ID | ID обслуживания.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя обслуживания.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| active_since | timestamp | Время, когда обслуживание становится активным (включительно).<br><br>Указанное значение будет округлено вниз до минут.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| active_till | timestamp | Время, когда обслуживание перестает быть активным (исключительно).<br><br>Указанное значение будет округлено вниз до минут.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| description | string | Описание обслуживания. |

| Property | Type | Description |
|---|---|---|
| maintenance_type | integer | Тип обслуживания.<br><br>Возможные значения:<br>0 - *(по умолчанию)* с сбором данных;<br>1 - без сбора данных. |
| tags_evaltype | integer | Метод оценки тегов проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* And/Or;<br>2 - Or. |

Период времени

Объект периода времени используется для определения периодов, когда обслуживание должно вступить в силу. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| period | integer | Длительность периода обслуживания в секундах.<br><br>Указанное значение будет округлено вниз до минут.<br><br>Допустимый диапазон значений: 300 - 86399940.<br><br>По умолчанию: 3600. |
| timeperiod_type | integer | Тип периода времени.<br><br>Возможные значения:<br>0 - *(по умолчанию)* только один раз;<br>2 - ежедневно;<br>3 - еженедельно;<br>4 - ежемесячно. |
| start_date | timestamp | Дата, когда период обслуживания должен вступить в силу.<br>Указанное значение будет округлено вниз до минут.<br><br>По умолчанию: текущая дата.<br><br>Property behavior:<br>- *supported* если `timeperiod_type` установлено в ”one time only” |
| start_time | integer | Время суток, когда обслуживание начинается, в секундах.<br>Указанное значение будет округлено вниз до минут.<br><br>По умолчанию: 0.<br><br>Property behavior:<br>- *supported* если `timeperiod_type` установлено в ”daily”, ”weekly” или ”monthly” |

| Property | Type | Description |
|---|---|---|
| every | integer | Для ежедневных и еженедельных периодов `every` определяет интервалы в днях или неделях, через которые обслуживание должно вступить в силу.<br>Значение по умолчанию, если `timeperiod_type` установлено в ”daily” или ”weekly”: 1.<br><br>Для ежемесячных периодов, когда задан `day`, свойство `every` определяет день месяца, когда обслуживание должно вступить в силу.<br>Значение по умолчанию, если `timeperiod_type` установлено в ”monthly” и задан `day`: 1.<br><br>Для ежемесячных периодов, когда задан `dayofweek`, свойство `every` определяет неделю месяца, когда обслуживание должно вступить в силу.<br>Возможные значения, если `timeperiod_type` установлено в ”monthly” и задан `dayofweek`:<br>1 - *(по умолчанию)* первая неделя;<br>2 - вторая неделя;<br>3 - третья неделя;<br>4 - четвертая неделя;<br>5 - последняя неделя.<br><br>Property behavior:<br>- *supported* если `timeperiod_type` установлено в ”daily”, ”weekly” или ”monthly” |
| dayofweek | integer | Дни недели, когда обслуживание должно вступить в силу.<br><br>Возможные битовые значения:<br>1 - понедельник;<br>2 - вторник;<br>4 - среда;<br>8 - четверг;<br>16 - пятница;<br>32 - суббота;<br>64 - воскресенье.<br><br>Это поле является битовой маской; допустима любая сумма возможных битовых значений (например, 21 для понедельника, среды и пятницы).<br><br>Property behavior:<br>- *required* если `timeperiod_type` установлено в ”weekly” или если `timeperiod_type` установлено в ”monthly” и `day` не задан |
| day | integer | День месяца, когда обслуживание должно вступить в силу.<br><br>Property behavior:<br>- *required* если `timeperiod_type` установлено в ”monthly” и `dayofweek` не задан |

| Property | Type | Description |
|---|---|---|
| month | integer | Месяцы, когда обслуживание должно вступить в силу.<br><br>Возможные битовые значения:<br>1 - январь;<br>2 - февраль;<br>4 - март;<br>8 - апрель;<br>16 - май;<br>32 - июнь;<br>64 - июль;<br>128 - август;<br>256 - сентябрь;<br>512 - октябрь;<br>1024 - ноябрь;<br>2048 - декабрь.<br><br>Это поле является битовой маской; допустима любая сумма возможных битовых значений (например, 585 для января, апреля, июля и октября).<br><br>Property behavior:<br>- *required* если `timeperiod_type` установлено в ”monthly” |

Тег проблемы

Объект тега проблемы используется для определения, какие проблемы должны быть подавлены, когда вступает в силу обслуживание. Теги можно указывать только в том случае, если `maintenance_type` объекта Maintenance установлен в ”with data collection”. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега проблемы.<br><br>Поведение свойства:<br>- *required* |
| operator | integer | Оператор условия.<br><br>Возможные значения:<br>0 - Равно;<br>2 - *(по умолчанию)* Содержит. |
| value | string | Значение тега проблемы. |

**maintenance.create**

Описание

`object maintenance.create( / maintenances)`

Этот метод позволяет создавать новые обслуживания.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Технические обслуживания для создания.

В дополнение к стандартным свойствам технического обслуживания, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы узлов сети, которые будут переведены на техническое обслуживание.<br><br>Группы узлов сети должны содержать только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* if `hosts` is not set |
| hosts | object/array | Узлы сети, которые будут переведены на техническое обслуживание.<br><br>Узлы сети должны содержать только свойство `hostid`.<br><br>Поведение параметра:<br>- *required* if `groups` is not set |
| timeperiods | object/array | Периоды времени технического обслуживания.<br><br>Поведение параметра:<br>- *required* |
| tags | object/array | Теги проблем.<br><br>Определяет, какие проблемы должны быть подавлены.<br>Если теги не указаны, будут подавлены все активные проблемы узлов сети, находящихся на техническом обслуживании.<br><br>Поведение параметра:<br>- *supported* if `maintenance_type` of Maintenance object is set to "with data collection" |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных обслуживаний, указанных в свойстве `maintenanceids`. Порядок возвращаемых ID совпадает с порядком переданных обслуживаний.

Примеры

Создание обслуживания

Создайте обслуживание со сбором данных для группы узлов сети с ID "2" и с тегами проблем **service:mysqld** и **error**. Оно должно быть активно с 17.03.2026 по 17.03.2027, вступать в силу каждое воскресенье в 18:00 и длиться один час.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.create",
    "params": {
        "name": "Sunday maintenance",
        "active_since": 1773720240,
        "active_till": 1805256240,
        "tags_evaltype": 0,
        "groups": [
            {"groupid": "2"}
        ],
        "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3"
        ]
    },
    "id": 1
}
```

См. также

- Период времени

Источник

CMaintenance::create() в *ui/include/classes/api/services/CMaintenance.php*.

**maintenance.delete**

Описание

`object maintenance.delete(    maintenanceIds)`

Этот метод позволяет удалять периоды обслуживания.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых периодов обслуживания.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных периодов обслуживания, указанных в свойстве `maintenanceids`.

Примеры

Удаление нескольких периодов обслуживания

Удалите два периода обслуживания.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.delete",
    "params": [
        "3",
        "1"
```

```
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3",
            "1"
        ]
    },
    "id": 1
}
```

Источник

CMaintenance::delete() в *ui/include/classes/api/services/CMaintenance.php*.

### maintenance.get

Описание

`integer/array maintenance.get(object parameters)`

Метод позволяет получать обслуживания в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groupids | ID/array | Возвращать только обслуживания, назначенные указанным группам узлов сети. |
| hostids | ID/array | Возвращать только обслуживания, назначенные указанным узлам сети. |
| maintenanceids | ID/array | Возвращать только обслуживания с указанными ID. |
| selectHostGroups | query | Возвращать свойство `hostgroups` с группами узлов сети, назначенными обслуживанию. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, назначенными обслуживанию. |
| selectTags | query | Возвращать свойство `tags` с тегами проблем обслуживания. |
| selectTimeperiods | query | Возвращать свойство `timeperiods` с периодами времени обслуживания. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `maintenanceid`, `name`, `maintenance_type`, `active_since`, `active_till`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |

| Parameter | Type | Description |
|---|---|---|
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использован параметр `countOutput`.

Примеры

Получение обслуживаний

Получите все настроенные обслуживания, а также данные о назначенных группах узлов сети, определенных временных периодах и тегах проблем.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.get",
    "params": {
        "output": "extend",
        "selectHostGroups": "extend",
        "selectTimeperiods": "extend",
        "selectTags": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "maintenanceid": "3",
            "name": "Sunday maintenance",
            "maintenance_type": "0",
            "description": "",
            "active_since": "1358844540",
            "active_till": "1390466940",
            "tags_evaltype": "0",
            "hostgroups": [
                {
                    "groupid": "4",
                    "name": "Zabbix servers",
                    "flags": "0",
                    "uuid": "6f6799aa69e844b4b3918f779f2abf08"
                }
            ],
            "timeperiods": [
                {
                    "timeperiod_type": "3",
                    "every": "1",
                    "month": "0",
                    "dayofweek": "1",
                    "day": "0",
                    "start_time": "64800",
                    "period": "3600",
                    "start_date": "2147483647"
                }
            ],
```

```
            "tags": [
                {
                    "tag": "service",
                    "operator": "0",
                    "value": "mysqld",
                },
                {
                    "tag": "error",
                    "operator": "2",
                    "value": ""
                }
            ]
        }
    ],
    "id": 1
}
```

См. также

- Узел сети
- Группа узлов сети
- Период времени

Источник

CMaintenance::get() в *ui/include/classes/api/services/CMaintenance.php*.

**maintenance.update**

Описание

`object maintenance.update(  /  maintenances)`

Этот метод позволяет обновлять существующие обслуживания.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства обслуживания, которые необходимо обновить.

Свойство `maintenanceid` должно быть определено для каждого обслуживания, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам обслуживания метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы узлов сети, которые заменят текущие группы.<br><br>Группы узлов сети должны иметь определенным только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* если `hosts` не задан |
| hosts | object/array | Узлы сети, которые заменят текущие узлы сети.<br><br>У узлов сети должно быть определено только свойство `hostid`.<br><br>Поведение параметра:<br>- *required* если `groups` не задан |
| timeperiods | object/array | Периоды времени обслуживания, которые заменят текущие периоды. |

| Parameter | Type | Description |
|---|---|---|
| tags | object/array | Теги проблемы, которые заменят текущие теги.<br><br>Поведение параметра:<br>- *supported* если `maintenance_type` объекта обслуживания установлен в "with data collection" |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных обслуживаний, указанных в свойстве `maintenanceids`.

Примеры

Назначение других узлов сети

Замените узлы сети, которые в настоящее время назначены на обслуживание, двумя другими.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.update",
    "params": {
        "maintenanceid": "3",
        "hosts": [
            {"hostid": "10085"},
            {"hostid": "10084"}
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3"
        ]
    },
    "id": 1
}
```

См. также

- Период времени

Источник

CMaintenance::update() в *ui/include/classes/api/services/CMaintenance.php*.

**Отчет**

Этот класс предназначен для работы с запланированными отчетами.

Ссылки на объекты:

- Отчет
- Пользователи
- Группы пользователей

Доступные методы:

- report.create - создать новые запланированные отчеты
- report.delete - удалить запланированные отчеты
- report.get - получить запланированные отчеты
- report.update - обновить запланированные отчеты

**Объект отчета**

Следующие объекты напрямую связаны с API `report`.

Отчет

Объект отчета имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| reportid | ID | ID отчета.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| userid | ID | ID пользователя, создавшего отчет.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Уникальное имя отчета.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| dashboardid | ID | ID панели, на основе которой формируется отчет.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| period | integer | Период, за который будет подготовлен отчет.<br><br>Возможные значения:<br>0 - *(по умолчанию)* предыдущий день;<br>1 - предыдущая неделя;<br>2 - предыдущий месяц;<br>3 - предыдущий год. |
| cycle | integer | Периодичность повторения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* ежедневно;<br>1 - еженедельно;<br>2 - ежемесячно;<br>3 - ежегодно. |
| start_time | integer | Время суток в секундах, когда отчет будет подготовлен к отправке.<br><br>По умолчанию: 0. |
| weekdays | integer | Дни недели для отправки отчета.<br><br>Возможные значения битовой маски:<br>1 - понедельник;<br>2 - вторник;<br>4 - среда;<br>8 - четверг;<br>16 - пятница;<br>32 - суббота;<br>64 - воскресенье.<br><br>Это поле является битовой маской; допустима любая сумма возможных значений битовой маски (например, 21 для понедельника, среды и пятницы).<br><br>По умолчанию: 0.<br><br>Поведение свойства:<br>- *обязательно*, если `cycle` установлен в ”weekly”. |

| Property | Type | Description |
|---|---|---|
| active_since | string | Дата начала.<br><br>Возможные значения:<br>пустая строка - *(по умолчанию)* не указано (хранится как 0);<br>конкретная дата в формате YYYY-MM-DD (хранится как временная метка начала дня (00:00:00)). |
| active_till | string | Дата окончания.<br><br>Возможные значения:<br>пустая строка - *(по умолчанию)* не указано (хранится как 0);<br>конкретная дата в формате YYYY-MM-DD (хранится как временная метка конца дня (23:59:59)). |
| subject | string | Тема сообщения отчета. |
| message | string | Текст сообщения отчета. |
| status | integer | Включен отчет или отключен.<br><br>Возможные значения:<br>0 - Отключен;<br>1 - *(по умолчанию)* Включен. |
| description | text | Описание отчета. |
| state | integer | Состояние отчета.<br><br>Возможные значения:<br>0 - *(по умолчанию)* отчет еще не был обработан;<br>1 - отчет был сгенерирован и успешно отправлен всем получателям;<br>2 - не удалось сгенерировать отчет; ”info” содержит информацию об ошибке;<br>3 - отчет был сгенерирован, но отправка некоторым (или всем) получателям не удалась; ”info” содержит информацию об ошибке.<br><br>Поведение свойства:<br>- *только для чтения* |
| lastsent | timestamp | Unix timestamp последнего успешно отправленного отчета.<br><br>Поведение свойства:<br>- *только для чтения* |
| info | string | Описание ошибки или дополнительная информация.<br><br>Поведение свойства:<br>- *только для чтения* |

Пользователи

Объект users имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| userid | ID | ID пользователя, которому нужно отправить отчет.<br><br>Поведение свойства:<br>- *required* |
| access_userid | ID | ID пользователя, от имени которого будет сгенерирован отчет.<br><br>0 - *(default)* Сгенерировать отчет от имени получателя. |
| exclude | integer | Исключать ли пользователя из списка рассылки.<br><br>Возможные значения:<br>0 - *(default)* Включить;<br>1 - Исключить. |

Группы пользователей

Объект групп пользователей имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| usrgrpid | ID | ID группы пользователей, которой будет отправлен отчет.<br><br>Поведение свойства:<br>- *required* |
| access_userid | ID | ID пользователя, от имени которого будет сгенерирован отчет.<br><br>0 - *(по умолчанию)* Сгенерировать отчет от имени получателя. |

**report.create**

Описание

`object report.create(  /  reports)`

Этот метод позволяет создавать новые запланированные отчеты.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Запланированные отчеты для создания.

Помимо стандартных свойств запланированного отчета, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| users | object/array | Пользователи, которым отправлять отчет.<br><br>Поведение параметра:<br>- *required* если `user_groups` не задан |
| user_groups | object/array | Группы пользователей, которым отправлять отчет.<br><br>Поведение параметра:<br>- *required* если `users` не задан |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных запланированных отчетов в свойстве `reportids`. Порядок возвращаемых идентификаторов соответствует порядку переданных запланированных отчетов.

Примеры

Создание запланированного отчета

Создайте еженедельный отчет, который будет формироваться за предыдущую неделю каждый понедельник-пятницу в 12:00 с 2021-04-01 по 2021-08-31.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "report.create",
    "params": {
        "userid": "1",
        "name": "Weekly report",
        "dashboardid": "1",
        "period": "1",
        "cycle": "1",
```

```
        "start_time": "43200",
        "weekdays": "31",
        "active_since": "2021-04-01",
        "active_till": "2021-08-31",
        "subject": "Weekly report",
        "message": "Report accompanying text",
        "status": "1",
        "description": "Report description",
        "users": [
            {
                "userid": "1",
                "access_userid": "1",
                "exclude": "0"
            },
            {
                "userid": "2",
                "access_userid": "0",
                "exclude": "1"
            }
        ],
        "user_groups": [
            {
                "usrgrpid": "7",
                "access_userid": "0"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

См. также

- Пользователи
- Группы пользователей

Источник

CReport::create() в *ui/include/classes/api/services/CReport.php*.

**report.delete**

Описание

`object report.delete(    reportids)`

Этот метод позволяет удалять запланированные отчеты.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(array) Идентификаторы запланированных отчетов для удаления.

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы удаленных запланированных отчетов в свойстве reportids.

Примеры

Удаление нескольких запланированных отчетов

Удаление двух запланированных отчетов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "report.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Источник

CReport::delete() в *ui/include/classes/api/services/CReport.php*.

**report.get**

Описание

integer/array report.get( )

Метод позволяет получать запланированные отчеты по заданным параметрам.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object) Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| reportids | ID/array | Возвращать только запланированные отчеты с указанными ID отчетов. |
| expired | boolean | Если установлено значение true, возвращать только просроченные запланированные отчеты. |
| selectUsers | query | Возвращать свойство users с пользователями, которым настроена отправка отчета. |
| selectUserGroups | query | Возвращать свойство user_groups с группами пользователей, которым настроена отправка отчета. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `reportid`, `name`, `status`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение данных отчета

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "report.get",
    "params": [
        "output": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "reportids": ["1", "2"]
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "reportid": "1",
            "userid": "1",
            "name": "Weekly report",
            "dashboardid": "1",
            "period": "1",
            "cycle": "1",
            "start_time": "43200",
            "weekdays": "31",
            "active_since": "2021-04-01",
            "active_till": "2021-08-31",
            "subject": "Weekly report",
            "message": "Report accompanying text",
            "status": "1",
            "description": "Report description",
            "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": "Monthly report",
            "dashboardid": "2",
            "period": "2",
            "cycle": "2",
            "start_time": "0",
            "weekdays": "0",
            "active_since": "2021-05-01",
            "active_till": "",
            "subject": "Monthly report",
            "message": "Report accompanying text",
            "status": "1",
            "description": "",
            "state": "0",
            "lastsent": "0",
            "info": "",
            "users": [
                {
                    "userid": "1",
                    "access_userid": "1",
                    "exclude": "0"
                }
            ],
            "user_groups": []
        }
    ],
    "id": 1
}
```


См. также

- Пользователи
- Группы пользователей

Источник

CReport::get() в *ui/include/classes/api/services/CReport.php*.

**report.update**

Описание

`object report.update( / reports)`

Этот метод позволяет обновлять существующие запланированные отчеты.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства запланированного отчета, которые необходимо обновить.

Свойство `reportid` должно быть определено для каждого запланированного отчета, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам запланированного отчета метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| users | object/array | Пользователи, которые заменят текущих пользователей, назначенных запланированному отчету.<br><br>Поведение параметра:<br>- *required* если `user_groups` не задан |
| user_groups | object/array | Группы пользователей, которые заменят текущие группы пользователей, назначенные запланированному отчету.<br><br>Поведение параметра:<br>- *required* если `users` не задан |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных запланированных отчетов в свойстве `reportids`.

Примеры

Отключение запланированного отчета

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "report.update",
    "params": {
        "reportid": "1",
        "status": "0"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

См. также

- Пользователи
- Группы пользователей

Источник

CReport::update() в *ui/include/classes/api/services/CReport.php*.

### Панель мониторинга

Этот класс предназначен для работы с панелями мониторинга.

Ссылки на объекты:

- Панель мониторинга
- Страница панели мониторинга
  - Виджет панели мониторинга
    - Поле виджета панели мониторинга
- Группа пользователей панели мониторинга
- Пользователь панели мониторинга

Доступные методы:

- dashboard.create - создать новые панели мониторинга
- dashboard.delete - удалить панели мониторинга
- dashboard.get - получить панели мониторинга
- dashboard.update - обновить панели мониторинга

### Объект панели

Следующие объекты напрямую связаны с API панелей `dashboard`.

Панель мониторинга

Объект панели мониторинга имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dashboardid | ID | ID панели мониторинга.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| name | string | Имя панели мониторинга.<br><br>Property behavior:<br>- *required* for create operations |
| userid | ID | ID пользователя, который является владельцем панели мониторинга. |
| private | integer | Тип общего доступа к панели мониторинга.<br><br>Possible values:<br>0 - общедоступная панель мониторинга;<br>1 - *(default)* частная панель мониторинга. |
| display_period | integer | Период отображения страницы по умолчанию (в секундах).<br><br>Possible values: 10, 30, 60, 120, 600, 1800, 3600.<br><br>Default: 30. |
| auto_start | integer | Автоматически запускать слайд-шоу.<br><br>Possible values:<br>0 - не запускать слайд-шоу автоматически;<br>1 - *(default)* автоматически запускать слайд-шоу. |

Страница панели

Объект страницы панели имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dashboard_pageid | ID | ID страницы панели.<br><br>Property behavior:<br>- *read-only* |
| name | string | Имя страницы панели.<br><br>По умолчанию: пустая строка. |
| display_period | integer | Период отображения страницы панели (в секундах).<br><br>Допустимые значения: 0, 10, 30, 60, 120, 600, 1800, 3600.<br><br>По умолчанию: 0 (будет использоваться период отображения страницы по умолчанию). |
| widgets | array | Массив объектов dashboard widget. |

Виджет панели

Объект виджета панели имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| widgetid | ID | ID виджета панели.<br><br>Property behavior:<br>- *read-only* |

| Property | Type | Description |
|---|---|---|
| type | string | Тип виджета панели.<br><br>Возможные значения:<br>actionlog - Журнал действий;<br>clock - Часы;<br>discovery - Состояние обнаружения;<br>favgraphs - Избранные графики;<br>favmaps - Избранные карты;<br>gauge - Индикатор;<br>geomap - Геокарта;<br>graph - График (классический);<br>graphprototype - Прототип графика;<br>honeycomb - Honeycomb;<br>hostavail - Доступность узла сети;<br>hostcard - Карточка узла сети;<br>hostnavigator - Навигатор узлов сети;<br>itemcard - Карточка элемента данных;<br>itemhistory - История элемента данных;<br>itemnavigator - Навигатор элементов данных;<br>item - Значение элемента данных;<br>map - Карта;<br>navtree - Дерево навигации по карте;<br>piechart - Круговая диаграмма;<br>problemhosts - Узлы сети с проблемами;<br>problems - Проблемы;<br>problemsbysv - Проблемы по уровню серьезности;<br>slareport - Отчет SLA;<br>svggraph - График;<br>systeminfo - Системная информация;<br>tophosts - Топ узлов сети;<br>topitems - Топ элементов данных;<br>toptriggers - Топ триггеров;<br>trigover - Обзор триггеров;<br>url - URL;<br>web - Мониторинг веб-сайтов.<br><br>Property behavior:<br>- *required* |
| name | string | Пользовательское имя виджета. |
| x | integer | Горизонтальная позиция от левой стороны панели.<br><br>Возможные значения: от 0 до 71. |
| y | integer | Вертикальная позиция от верхней стороны панели.<br><br>Возможные значения: от 0 до 63. |
| width | integer | Ширина виджета.<br><br>Возможные значения: от 1 до 72. |
| height | integer | Высота виджета.<br><br>Возможные значения: от 1 до 64. |
| view_mode | integer | Режим отображения виджета.<br><br>Возможные значения:<br>0 - *(по умолчанию)* стандартный вид виджета;<br>1 - со скрытым заголовком; |
| fields | array | Массив объектов поле виджета панели.<br><br>Property behavior:<br>- см. отдельные виджеты в Поля виджета панели |

Поле виджета панели

Объект поля виджета панели имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип поля виджета.<br><br>Возможные значения:<br>0 - Integer;<br>1 - String;<br>2 - Группа узлов сети;<br>3 - Узел сети;<br>4 - Элемент данных;<br>5 - Прототип элемента данных;<br>6 - График;<br>7 - Прототип графика;<br>8 - Карта;<br>9 - Сервис;<br>10 - SLA;<br>11 - Пользователь;<br>12 - Действие;<br>13 - Тип оповещения.<br><br>Поведение свойства:<br>- *required* |
| name | string | Имя поля виджета.<br><br>Возможные значения: см. Поля виджета панели.<br><br>Поведение свойства:<br>- *required* |
| value | mixed | Значение поля виджета в зависимости от типа.<br><br>Возможные значения: см. Поля виджета панели.<br><br>Поведение свойства:<br>- *required* |

Группа пользователей панели мониторинга

Список прав доступа к панели мониторинга на основе групп пользователей. Имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| usrgrpid | ID | ID группы пользователей.<br><br>Property behavior:<br>- *required* |
| permission | integer | Тип уровня доступа.<br><br>Возможные значения:<br>2 - только чтение;<br>3 - чтение и запись.<br><br>Property behavior:<br>- *required* |

Пользователь панели

Список прав доступа к панели на основе пользователей. Имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| userid | ID | ID пользователя.<br><br>Property behavior:<br>- *required* |
| permission | integer | Тип уровня прав доступа.<br><br>Возможные значения:<br>2 - только чтение;<br>3 - чтение и запись.<br><br>Property behavior:<br>- *required* |

**dashboard.create**

Описание

`object dashboard.create( / dashboards)`

Этот метод позволяет создавать новые панели.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Панели мониторинга для создания.

В дополнение к стандартным свойствам панели мониторинга, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| pages | array | Страницы панели мониторинга, которые нужно создать для панели мониторинга. Страницы панели мониторинга будут упорядочены в том же порядке, в котором они указаны.<br><br>Поведение параметра:<br>- *required* |
| users | array | Пользователь панели мониторинга общие доступы, которые нужно создать на панели мониторинга. |
| userGroups | array | Группа пользователей панели мониторинга общие доступы, которые нужно создать на панели мониторинга. |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID созданных панелей, указанных в свойстве `dashboardids`. Порядок возвращаемых ID совпадает с порядком переданных панелей.

Примеры

Создание дашборда

Создайте дашборд с именем ”My dashboard” с одним виджетом Problems с тегами и с использованием двух типов общего доступа (группа пользователей и пользователь) на одной странице дашборда.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
```

```
    "id": 1
}
```

См. также

- Страница панели мониторинга
- Виджет панели мониторинга
- Поле виджета панели мониторинга
- Пользователь панели мониторинга
- Группа пользователей панели мониторинга

Источник

CDashboard::create() в *ui/include/classes/api/services/CDashboard.php*.

### dashboard.delete

Описание

`object dashboard.delete(    dashboardids)`

Этот метод позволяет удалять панели.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых панелей.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных панелей, указанных в свойстве `dashboardids`.

Примеры

Удаление нескольких панелей мониторинга

Удаление двух панелей мониторинга.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.delete",
    "params": [
        "2",
        "3"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2",
            "3"
        ]
    },
    "id": 1
}
```

Источник

CDashboard::delete() в *ui/include/classes/api/services/CDashboard.php*.

**dashboard.get**

Описание

`integer/array dashboard.get(    parameters)`

Этот метод позволяет получать панели в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dashboardids | ID/array | Возвращать только панели с указанными ID. |
| selectPages | query | Возвращать свойство `pages` со страницами панели, правильно отсортированными. |
| selectUsers | query | Возвращать свойство `users` с пользователями, с которыми расшарена панель. |
| selectUserGroups | query | Возвращать свойство `userGroups` с группами пользователей, с которыми расшарена панель. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `dashboardid`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(  /  )` Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение панели по ID

Получение всех данных о панелях ”1” и ”2”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "dashboardids": [
```

```
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


См. также

- Страница панели
- Виджет панели
- Поле виджета панели
- Пользователь панели
- Группа пользователей панели

Источник

CDashboard::get() в *ui/include/classes/api/services/CDashboard.php*.

### dashboard.update

Описание

`object dashboard.update(  /  dashboards)`

Этот метод позволяет обновлять существующие панели.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства дашборда, которые необходимо обновить.

Для каждого дашборда должно быть определено свойство `dashboardid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств дашборда, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| pages | array | Страницы дашборда для замены существующих страниц дашборда.<br><br>Страницы дашборда обновляются по свойству `dashboard_pageid`. Новые страницы дашборда будут созданы для объектов без свойства `dashboard_pageid`, а существующие страницы дашборда будут удалены, если не будут повторно использованы. Страницы дашборда будут упорядочены в том же порядке, в котором они указаны. Будут обновлены только указанные свойства страниц дашборда. |
| users | array | Общие элементы пользователя дашборда для замены существующих элементов. |
| userGroups | array | Общие элементы группы пользователей дашборда для замены существующих элементов. |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных панелей, указанных в свойстве `dashboardids`.

Примеры

Переименование панели мониторинга

Переименуйте панель мониторинга в "SQL server status".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "name": "SQL server status"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Обновление страниц панели

Переименуйте первую страницу панели, замените виджеты на второй странице панели и добавьте новую страницу в качестве третьей. Удалите все остальные страницы панели.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Изменение владельца панели мониторинга

Доступно только администраторам и супер-администраторам.

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "userid": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

См. также

- Страница панели
- Виджет панели
- Поле виджета панели
- Пользователь панели
- Группа пользователей панели

Источник

CDashboard::update() в *ui/include/classes/api/services/CDashboard.php*.

#### Поля виджета панели мониторинга

Эта страница содержит навигационные ссылки к параметрам виджетов панели и возможным значениям свойств для соответствующих объектов поля виджета панели.

Чтобы посмотреть параметры и значения свойств для каждого виджета, перейдите на отдельные страницы виджетов:

- Журнал действий
- Часы
- Статус обнаружения
- Избранные графики
- Избранные карты
- Шкала
- Геокарта
- График
- График (классический)
- Шаблон графика
- Соты
- Доступность узлов сети
- Карточка узла сети
- Навигатор узлов сети
- Карточка элемента данных
- История элемента данных
- Навигатор элементов данных
- Значение элемента данных
- Карта
- Дерево навигации по карте
- Круговая диаграмма
- Проблемные узлы сети
- Проблемы
- Отчет SLA
- Системная информация
- Проблемы по уровню серьезности
- Топ узлов сети

- Топ элементов данных
- Топ триггеров
- Обзор триггеров
- URL
- Веб-мониторинг

**Attention:**
Устаревшие виджеты будут удалены в следующем major-релизе.

1. Журнал действий

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Action log* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Action log*, см. поведение параметров, описанное в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Action log*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Recipients* | 11 | userids.0 | ID пользователя.<br><br>Примечание: чтобы настроить нескольких пользователей, создайте объект поля виджета панели для каждого пользователя с увеличенным номером в имени свойства. |
| *Actions* | 12 | actionids.0 | ID действия.<br><br>Примечание: чтобы настроить несколько действий, создайте объект поля виджета панели для каждого действия с увеличенным номером в имени свойства. |
| *Media types* | 13 | mediatypeids.0 | ID типа оповещения.<br><br>Примечание: чтобы настроить несколько типов оповещений, создайте объект поля виджета панели для каждого типа оповещения с увеличенным номером в имени свойства. |
| *Status* | 0 | statuses.0 | 0 - В процессе;<br>1 - Отправлено/выполнено;<br>2 - Ошибка.<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения с увеличенным номером в имени свойства. |
| *Search string* | 1 | message | Любое строковое значение. |

| Parameter | type | name | value |
|---|---|---|---|
| *Time period* | 1 | time_period._referenc | DASHBOARD._timeperiod - использовать селектор *Time period* в качестве источника данных;<br>ABCDE._timeperiod - использовать совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных.<br><br>По умолчанию: DASHBOARD._timeperiod<br><br>Кроме того, можно задать период времени только в параметрах *From* и *To*. |
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или в относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported*, если *Time period* не задан<br>- *required*, если задан time_period.to |
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или в относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported*, если *Time period* не задан<br>- *required*, если задан time_period.from |
| *Sort entries by* | 0 | sort_triggers | 3 - Время (по возрастанию);<br>4 - *(по умолчанию)* Время (по убыванию);<br>5 - Тип (по возрастанию);<br>6 - Тип (по убыванию);<br>7 - Статус (по возрастанию);<br>8 - Статус (по убыванию);<br>11 - Получатель (по возрастанию);<br>12 - Получатель (по убыванию). |
| *Show lines* | 0 | show_lines | Возможные значения: от 1 до 100.<br><br>По умолчанию: 25. |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *Action log*. Дополнительную информацию о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Журнал действий*

Настройте виджет *Журнал действий*, который отображает 10 записей с подробностями операций действий, отсортированных по времени (по возрастанию). Кроме того, отображайте подробности только для тех операций действий, которые пытались отправить письмо пользователю "1", но не были выполнены успешно.

Запрос:

```
{
    "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
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

2. Часы

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Clock* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Clock*, ознакомьтесь с поведением параметров, описанным в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Clock*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |
| *Time type* | 0 | time_type | 0 - *(по умолчанию)* Местное время;<br>1 - Время сервера;<br>2 - Время узла сети. |
| *Clock type* | 0 | clock_type | 0 - *(по умолчанию)* Аналоговые;<br>1 - Цифровые. |

Следующие параметры поддерживаются, если для *Time type* установлено значение ”Host time”.

| Parameter | type | name | value |
|---|---|---|---|
| *Item* | 4 | itemid.0 | ID элемента данных.<br><br>Поведение параметра:<br>*- required* |

Следующие параметры поддерживаются, если для *Clock type* установлено значение ”Digital”.

| Parameter | type | name | value |
|---|---|---|---|
| *Show* | 0 | show.0 | 1 - Дата;<br>2 - *(по умолчанию)* Время;<br>3 - Часовой пояс.<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения с увеличенным номером в имени свойства. |

Расширенная конфигурация

Следующие параметры расширенной конфигурации поддерживаются, если для *Clock type* установлено значение ”Digital”.

| Parameter | type | name | value |
|---|---|---|---|
| *Background color* | 1 | bg_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |

Дата

Следующие параметры расширенной конфигурации поддерживаются, если для *Clock type* установлено значение ”Digital”, а для *Show* — ”Date”.

| Parameter | type | name | value |
|---|---|---|---|
| *Bold* | 0 | date_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Color* | 1 | date_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |

Время

Следующие параметры расширенной конфигурации поддерживаются, если для *Clock type* задано значение ”Digital”, а для *Show* — значение ”Time”.

| Parameter | type | name | value |
|---|---|---|---|
| *Bold* | 0 | time_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Color* | 1 | time_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Seconds* | 0 | time_sec | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Format* | 0 | time_format | 0 - *(по умолчанию)* 24-часовой формат;<br>1 - 12-часовой формат. |

Часовой пояс

Следующие параметры расширенной конфигурации поддерживаются, если для *Clock type* установлено значение ”Digital”, а для *Show* — ”Time zone”.

| Parameter | type | name | value |
|---|---|---|---|
| *Bold* | 0 | tzone_bold | 0 - *(default)* Disabled;<br>1 - Enabled. |
| *Color* | 1 | tzone_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Time zone* | 1 | tzone_timezone | Допустимая строка часового пояса (например, Europe/Riga, system, UTC и т. д.). Полный список поддерживаемых часовых поясов см. в документации PHP.<br><br>По умолчанию: local.<br><br>Поведение параметра:<br>- *supported* if *Time type* is set to ”Local time” or ”Server time” |
| *Format* | 0 | tzone_format | 0 - *(default)* Short;<br>1 - Full.<br><br>Поведение параметра:<br>- *supported* if *Time type* is set to ”Local time” or ”Server time” |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *Clock*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Clock*

Настройте виджет *Clock*, который отображает местную дату, время и часовой пояс в виде настраиваемых цифровых часов.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

3 Статус обнаружения

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели мониторинга позволяют настраивать виджет *Discovery status* в методах `dashboard.create` и `dashboard.update`.

Параметры

Для виджета *Discovery status* поддерживаются следующие параметры.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *Discovery status*. Дополнительные сведения о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Discovery status*

Настройте виджет *Discovery status* с интервалом обновления 15 минут.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

4 Избранные графики

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели управления позволяют настраивать виджет *Избранные графики* в методах dashboard.create и dashboard.update.

Параметры

Следующие параметры поддерживаются для виджета *Избранные графики*.

| Parameter | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Favorite graphs*. Дополнительную информацию о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Избранные графики*

Настройте виджет *Избранные графики* с интервалом обновления 10 минут.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "favgraphs",
                        "name": "Favorite graphs",
                        "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

5 Избранные карты

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Избранные карты* в методах `dashboard.create` и `dashboard.update`.

Параметры

Следующие параметры поддерживаются для виджета *Избранные карты*.

| Parameter | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Favorite maps*. Дополнительные сведения о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Избранные карты*

Настройте виджет *Избранные карты* с интервалом обновления, установленным на 10 минут.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

6 Шкала

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Gauge* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Gauge*, ознакомьтесь с поведением параметров, описанным в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Gauge*.

| Параметр | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Item* | 4 | itemid.0 | ID элемента данных.<br><br>Поведение параметра:<br>- *required* если *Item (Widget)* не задан |
| *Item (Widget)* | 1 | itemid._reference | Вместо ID элемента данных:<br>ABCDE._itemid - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для элементов данных. |
| *Min* | 1 | min | Поведение параметра:<br>- *required* если *Item* не задан<br>Любое числовое значение. Поддерживаются суффиксы (например, "1d", "2w", "4K", "8G").<br><br>По умолчанию: "0". |
| *Max* | 1 | max | Любое числовое значение. Поддерживаются суффиксы (например, "1d", "2w", "4K", "8G").<br><br>По умолчанию: "100". |

| Параметр | type | name | value |
|---|---|---|---|
| *Value arc* | 1 | value_arc_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Arc background* | 1 | empty_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Background* | 1 | bg_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Show* | 0 | show.0 | 1 - Описание;<br>2 - Значение;<br>3 - Стрелка;<br>4 - Шкала;<br>5 - Дуга значения.<br><br>Примечание: Чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения с увеличивающимся номером в имени свойства.<br><br>По умолчанию: 1, 2, 4, 5.<br><br>Значения ”Стрелка” и ”Шкала” не поддерживаются, если одновременно:<br>- для *Show* не задан объект поля виджета панели со значением ”Value arc”;<br>- параметр дополнительной настройки *Show arc* установлен в ”Disabled”.<br><br>Параметры дополнительной настройки для опций *Show* не поддерживаются, если не заданы объекты полей виджета панели с соответствующими значениями. |
| *Override host* | 1 | override_hostid._reference | `ABCDE._hostid` - задайте совместимый виджет (с параметром *Reference*, установленным в ”ABCDE”) в качестве источника данных для узлов сети;<br>`DASHBOARD._hostid` - задайте селектор *Host* панели в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

Расширенная настройка

Следующие параметры расширенной настройки поддерживаются для виджета *Gauge*.

**Note:**
Число в имени свойства *Thresholds* (например, thresholds.0.color) указывает на место порога в списке, отсортированном по возрастанию. Однако если пороги настроены в другом порядке, значения будут отсортированы по возрастанию после обновления конфигурации виджета во веб-интерфейсе Zabbix (например, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

| Parameter | type | name | value |
|---|---|---|---|
| *Angle* | 0 | angle | Возможные значения: 180 *(по умолчанию)* или 270. |
| ***Description*** | | | |
| *Description* | 1 | description | Любое строковое значение, включая макросы.<br>Поддерживаемые макросы: {HOST.*}, {ITEM.*}, {INVENTORY.*}, пользовательские макросы.<br><br>По умолчанию: {ITEM.NAME}. |

| Parameter | type | name | value |
|---|---|---|---|
| *Size* | 0 | desc_size | Возможные значения от 1 до 100.<br><br>По умолчанию: 15. |
| *Vertical position* | 0 | desc_v_pos | 0 - Сверху;<br>1 - *(по умолчанию)* Снизу. |
| *Bold* | 0 | desc_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Color* | 1 | desc_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| ***Value*** | | | |
| *Decimal places* | 0 | decimal_places | Возможные значения от 1 до 10.<br><br>По умолчанию: 2. |
| *Size* | 0 | value_size | Возможные значения от 1 до 100.<br><br>По умолчанию: 25. |
| *Bold* | 0 | value_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Color* | 1 | value_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| ***Units*** | | | |
| *Units* (checkbox) | 0 | units_show | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Units* (value) | 1 | units | Любое строковое значение.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Units* (checkbox) установлено значение ”Enabled” |
| *Size* | 0 | units_size | Возможные значения от 1 до 100.<br><br>По умолчанию: 25.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Units* (checkbox) установлено значение ”Enabled” |
| *Bold* | 0 | units_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Units* (checkbox) установлено значение ”Enabled” |
| *Position* | 0 | units_pos | 0 - Перед значением;<br>1 - Над значением;<br>2 - *(по умолчанию)* После значения;<br>3 - Под значением.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Units* (checkbox) установлено значение ”Enabled”<br><br>Этот параметр игнорируется, если для него задано одно из следующих единиц, связанных со временем: unixtime, uptime, s. |
| *Color* | 1 | units_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| ***Value arc*** | | | |
| *Arc size* | 0 | value_arc_size | Возможные значения от 1 до 100.<br><br>По умолчанию: 20. |

| Parameter | type | name | value |
|---|---|---|---|
| ***Needle*** | | | |
| *Color* | 1 | needle_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто).<br><br>Поведение параметра:<br>- *поддерживается*, если для объекта поля виджета панели управления *Show* со значением ”Value arc” задано значение, или *Show arc* установлено в ”Enabled” |
| ***Scale*** | | | |
| *Show units* | 0 | scale_show_units | 0 - Отключено;<br>1 - *(по умолчанию)* Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Units* (checkbox) установлено значение ”Enabled” и либо для объекта поля виджета панели управления *Show* со значением ”Value arc” задано значение, либо *Show arc* установлено в ”Enabled” |
| *Size* | 0 | scale_size | Возможные значения от 1 до 100.<br><br>По умолчанию: 15.<br><br>Поведение параметра:<br>- *поддерживается*, если для объекта поля виджета панели управления *Show* со значением ”Value arc” задано значение, либо *Show arc* установлено в ”Enabled” |
| *Decimal places* | 0 | scale_decimal_places | Возможные значения от 1 до 10.<br><br>По умолчанию: 0.<br><br>Поведение параметра:<br>- *поддерживается*, если для объекта поля виджета панели управления *Show* со значением ”Value arc” задано значение, либо *Show arc* установлено в ”Enabled” |
| ***Thresholds*** | | | |
| *Color* | 1 | thresholds.0.color | Шестнадцатеричный код цвета (например, FF0000). |
| *Threshold* | 1 | thresholds.0.threshold | Любое числовое значение. Поддерживаются суффиксы (например, ”1d”, ”2w”, ”4K”, ”8G”). |
| *Show labels* | 0 | th_show_labels | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если пороги заданы и либо для объекта поля виджета панели управления *Show* со значением ”Value arc” задано значение, либо *Show arc* установлено в ”Enabled” |
| *Show arc* | 0 | th_show_arc | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если пороги заданы |
| *Arc size* | 0 | th_arc_size | Возможные значения от 1 до 100.<br><br>По умолчанию: 5.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Show arc* установлено значение ”Enabled” |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для

виджета *Gauge*. Дополнительную информацию о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Gauge*

Настройте виджет *Gauge*, который отображает значение элемента данных для элемента данных "44474" (Interface enp0s3: Bits sent). Кроме того, визуально доработайте виджет с помощью нескольких расширенных параметров, включая пороговые значения.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create

- dashboard.update

7 Геокарта

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Geomap* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Geomap*, см. описание поведения параметров в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Geomap*.

| Параметр | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Группы узлов сети* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Группы узлов сети (виджет)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>ABCDE._hostgroupids - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Узлы сети* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Группы узлов сети* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Узлы сети (виджет/панель)* | 1 | hostids._reference | Вместо ID узла сети:<br>DASHBOARD.hostids - задайте селектор *Узел сети* в качестве источника данных для узлов сети;<br>ABCDE._hostids - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Параметр | type | name | value |
|---|---|---|---|
| *Теги* | | | |
| *Тип оценки* | 0 | evaltype | 0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Имя тега* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраиваются *Теги*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Оператор* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраиваются *Теги*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Значение тега* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраиваются *Теги*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Начальный вид* | 1 | default_view | Разделенные запятыми *широта*, *долгота*, *уровень масштабирования* (*необязательно*, возможные значения от 0 до 30).<br>Пример: `40.6892494,-74.0466891,10`. |
| *Ссылка* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, которой принадлежит виджет.<br><br>Поведение параметра:<br>- *обязательно* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Geomap*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Geomap*

Настройте виджет *Geomap*, который отображает узлы сети из групп узлов сети ”2” и ”22” на основе следующей конфигурации тегов: тег с именем ”component” содержит значение ”node”, а тег с именем ”location” равен значению ”New York”. Кроме того, задайте начальный вид карты с координатами ”40.6892494” (широта), ”-74.0466891” (долгота) и

уровнем масштабирования ”10”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

8 График

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Graph* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы гарантировать успешное создание или обновление виджета *Graph*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Graph*.

| Parameter | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Ссылка* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах веб-интерфейса, к которому принадлежит виджет.<br><br>Поведение параметра:<br>- *required* |

Набор данных

Следующие параметры поддерживаются для настройки *Набора данных*.

**Note:**
Первое число в имени свойства (например, ds.0.hosts.0, ds.0.items.0) обозначает конкретный набор данных, а второе число, если оно присутствует, обозначает настроенный узел сети или элемент данных.

| Параметр | type | name | value |
|---|---|---|---|
| *Тип набора данных* | 0 | ds.0.dataset_type | 0 - Список элементов данных;<br>1 - *(по умолчанию)* Шаблоны элементов данных. |
| *Элементы данных* | 4 | ds.0.itemids.0 | ID элемента данных.<br><br>При настройке виджета на панели шаблона следует указывать только элементы данных, настроенные на шаблоне.<br><br>Примечание: чтобы настроить несколько элементов данных, создайте объект поля виджета панели для каждого элемента, увеличивая номер в имени свойства.<br><br>Поведение параметра:<br>- *обязателен*, если *Тип набора данных* установлен в ”Список элементов данных” и *Элементы данных (Виджет)* не задан |
| *Элементы данных (Виджет)* | 1 | ds.0.itemids.0._reference | Вместо ID элемента данных:<br>`ABCDE._itemid` - укажите совместимый виджет (с параметром *Reference*, установленным в ”`ABCDE`”) в качестве источника данных для элементов данных.<br><br>Примечание: чтобы настроить несколько виджетов, создайте объект поля виджета панели для каждого виджета, увеличивая номер в имени свойства.<br><br>Поведение параметра:<br>- *обязателен*, если *Тип набора данных* установлен в ”Список элементов данных” и *Элементы данных* не заданы |
| *Цвет* | 1 | ds.0.color.0 | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>Поведение параметра:<br>- *обязателен*, если *Тип набора данных* установлен в ”Список элементов данных” |

| Параметр | type | name | value |
|---|---|---|---|
| *Шаблоны узлов сети* | 1 | ds.0.hosts.0 | Имя или шаблон узла сети (например, ”Zabbix*”).<br><br>Поведение параметра:<br>- *обязателен*, если *Тип набора данных* установлен в ”Шаблоны элементов данных”<br><br>Этот параметр не поддерживается при настройке виджета на панели шаблона. |
| *Шаблоны элементов данных* | 1 | ds.0.items.0 | Имя или шаблон элемента данных (например, ”*: Number of processed *values per second”).<br><br>При настройке виджета на панели шаблона следует указывать только шаблоны для элементов данных, настроенных на шаблоне.<br><br>Поведение параметра:<br>- *обязателен*, если *Тип набора данных* установлен в ”Шаблоны элементов данных” |
| *Цвет* | 1 | ds.0.color | Шестнадцатеричный код цвета (например, FF0000).<br><br>Поведение параметра:<br>- *поддерживается*, если *Тип набора данных* установлен в ”Шаблоны элементов данных” и *Палитра цветов* не задана |
| *Палитра цветов* | 0 | ds.0.color_palette | Индекс палитры цветов.<br><br>Допустимые значения: 0-11.<br><br>По умолчанию: 0.<br><br>Поведение параметра:<br>- *поддерживается*, если *Тип набора данных* установлен в ”Шаблоны элементов данных” и *Цвет* не задан |
| *Отрисовка* | 0 | ds.0.type | 0 - *(по умолчанию)* Линия;<br>1 - Точки;<br>2 - Ступенчатая линия;<br>3 - Столбцы. |
| *С накоплением* | 0 | ds.0.stacked | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если *Отрисовка* установлена в ”Линия”, ”Ступенчатая линия” или ”Столбцы” |
| *Ширина* | 0 | ds.0.width | Допустимые значения: 1-10.<br><br>По умолчанию: 1.<br><br>Поведение параметра:<br>- *поддерживается*, если *Отрисовка* установлена в ”Линия” или ”Ступенчатая линия” |
| *Размер точки* | 0 | ds.0.pointsize | Допустимые значения: 1-10.<br><br>По умолчанию: 3.<br><br>Поведение параметра:<br>- *поддерживается*, если *Отрисовка* установлена в ”Точки” |
| *Прозрачность* | 0 | ds.0.transparency | Допустимые значения: 1-10.<br><br>По умолчанию: 5. |

| Параметр | type | name | value |
|---|---|---|---|
| *Заливка* | 0 | ds.0.fill | Допустимые значения: 1-10.<br><br>По умолчанию: 3.<br><br>Поведение параметра:<br>- *поддерживается*, если *Отрисовка* установлена в ”Линия” или ”Ступенчатая линия” |
| *Отсутствующие данные* | 0 | ds.0.missingdatafunc | 0 - *(по умолчанию)* Нет;<br>1 - Соединять;<br>2 - Считать как 0;<br>3 - Последнее известное.<br><br>Поведение параметра:<br>- *поддерживается*, если *Отрисовка* установлена в ”Линия” или ”Ступенчатая линия” |
| *Переопределить узел сети* | 1 | ds.0.override_hostid | `ABCDE` - укажите совместимый виджет (с параметром ”Reference”, установленным в ”ABCDE”) в качестве источника данных для узлов сети;<br>`DASHBOARD._hostid` - укажите селектор узлов сети на панели в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на панели шаблона. |
| *Ось Y* | 0 | ds.0.axisy | 0 - *(по умолчанию)* Левая;<br>1 - Правая. |
| *Сдвиг времени* | 1 | ds.0.timeshift | Допустимая строка времени (например, `3600`, `1h` и т. д.). Можно использовать суффиксы времени. Также допускаются отрицательные значения.<br><br>По умолчанию: "" (пусто). |
| *Функция агрегации* | 0 | ds.0.aggregate_function | 0 - *(по умолчанию)* не используется;<br>1 - min;<br>2 - max;<br>3 - avg;<br>4 - count;<br>5 - sum;<br>6 - first;<br>7 - last. |
| *Интервал агрегации* | 1 | ds.0.aggregate_interval | Допустимая строка времени (например, `3600`, `1h` и т. д.). Можно использовать суффиксы времени.<br><br>По умолчанию: `1h`. |
| *Агрегировать* | 0 | ds.0.aggregate_grouping | 0 - *(по умолчанию)* Каждый элемент данных;<br>1 - Набор данных.<br><br>Поведение параметра:<br>- *поддерживается*, если *Функция агрегации* установлена в ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first” или ”last” |
| *Аппроксимация* | 0 | ds.0.approximation | 1 - min;<br>2 - *(по умолчанию)* avg;<br>4 - max;<br>7 - all. |
| *Метка набора данных* | 1 | ds.0.data_set_label | Любое строковое значение.<br><br>По умолчанию: "" (пусто). |

Параметры отображения

Следующие параметры поддерживаются для настройки *Параметров отображения*.

| Параметр | type | name | value |
|---|---|---|---|
| *Выбор исторических данных* | 0 | source | 0 - *(по умолчанию)* Авто;<br>1 - История;<br>2 - Тренды. |
| *Простые триггеры* | 0 | simple_triggers | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Рабочее время* | 0 | working_time | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Линия перцентиля (слева)* | | | |
| *Состояние* | 0 | percentile_left | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если *Ось Y* (в конфигурации *Набор данных*) установлена в ”Left” |
| *Значение* | 0 | percentile_left_value | Допустимые значения: от 1 до 100.<br><br>Поведение параметра:<br>- *поддерживается*, если *Ось Y* (в конфигурации *Набор данных*) установлена в ”Left” |
| *Линия перцентиля (справа)* | | | |
| *Состояние* | 0 | percentile_right | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если *Ось Y* (в конфигурации *Набор данных*) установлена в ”Right” |
| *Значение* | 0 | percentile_right_value | Допустимые значения: от 1 до 100.<br><br>Поведение параметра:<br>- *поддерживается*, если *Ось Y* (в конфигурации *Набор данных*) установлена в ”Right” |

Период времени

Следующие параметры поддерживаются для настройки *Периода времени*.

| Parameter | type | name | value |
|---|---|---|---|
| *Период времени* | 1 | time_period._reference | `DASHBOARD._timeperiod` - задает селектор *Период времени* в качестве источника данных;<br>`ABCDE._timeperiod` - задает совместимый виджет (с параметром *Reference*, установленным в значение ”`ABCDE`”) в качестве источника данных.<br><br>По умолчанию: `DASHBOARD._timeperiod`<br><br>Кроме того, вы можете задать период времени только в параметрах *From* и *To*. |
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном (`YYYY-MM-DD hh:mm:ss`) или относительном формате времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Поведение параметра:<br>- *supported*, если *Период времени* не задан<br>- *required*, если задан `time_period.to` |

| Parameter | type | name | value |
|---|---|---|---|
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном (`YYYY-MM-DD hh:mm:ss`) или относительном формате времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Поведение параметра:<br>- *supported*, если *Период времени* не задан<br>- *required*, если задан `time_period.from` |

Оси

Для настройки *Оси* поддерживаются следующие параметры.

| Parameter | type | name | value |
|---|---|---|---|
| *Left Y* | 0 | lefty | 0 - Отключено;<br>1 - *(по умолчанию)* Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если *Y-axis* (в конфигурации *Data set*) установлено в "Left" |
| *Right Y* | 0 | righty | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если *Y-axis* (в конфигурации *Data set*) установлено в "Right" |
| *Scale* | 0 | lefty_scale<br>righty_scale | 0 - *(по умолчанию)* Линейная;<br>1 - Логарифмическая. |
| *Min* | 1 | lefty_min<br>righty_min | Любое числовое значение.<br><br>По умолчанию: "" (пусто). |
| *Max* | 1 | lefty_max<br>righty_max | Любое числовое значение.<br><br>По умолчанию: "" (пусто). |
| *Units* (type) | 0 | lefty_units<br>righty_units | 0 - *(по умолчанию)* Авто;<br>1 - Статический. |
| *Units* (value) | 1 | lefty_static_units<br>righty_static_units | Любое строковое значение.<br><br>По умолчанию: "" (пусто). |
| *X-Axis* | 0 | xaxis | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |

Легенда

Следующие параметры поддерживаются для настройки *Легенда*.

| Parameter | type | name | value |
|---|---|---|---|
| *Показывать легенду* | 0 | legend | 0 - Отключено;<br>1 - *(по умолчанию)* Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение "Включено" |

| Parameter | type | name | value |
|---|---|---|---|
| *Показывать min/avg/max* | 0 | legend_statistic | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Показывать функцию агрегации* | 0 | legend_aggregation | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Строки* | 0 | legend_lines_mode | 0 - *(по умолчанию)* Фиксированное;<br>1 - Переменное.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Количество строк/ Максимальное количество строк* | 0 | legend_lines | Допустимые значения: 1-10.<br><br>По умолчанию: 1.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Количество столбцов* | 0 | legend_columns | Допустимые значения: 1-4.<br><br>По умолчанию: 4.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено”, а для *Показывать min/avg/max* установлено значение ”Отключено” |

Проблемы

Следующие параметры поддерживаются для настройки *Problems*.

| Parameter | type | name | value |
|---|---|---|---|
| *Show problems* | 0 | show_problems | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Selected items only* | 0 | graph_item_problems | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Problem hosts* | 1 | problemhosts.0 | Имя узла сети.<br><br>Примечание: число в имени свойства указывает на настроенный узел сети.<br>Чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая число в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Severity* | 0 | severities.0 | 0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>По умолчанию: пусто (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая число в имени свойства. |
| *Problem* | 1 | problem_name | Имя события проблемы (без учета регистра, полное имя или его часть). |
| *Problem tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает на порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Problem tags* |
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает на порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Problem tags* |
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает на порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Problem tags* |

Переопределения

Следующие параметры поддерживаются для настройки *Переопределений*.

**Note:**
Первое число в имени свойства (например, or.0.hosts.0, or.0.items.0) представляет конкретный набор данных, а второе число, если оно присутствует, представляет настроенный узел сети или элемент данных.

| Parameter | type | name | value |
|---|---|---|---|
| *Шаблоны узлов сети* | 1 | or.0.hosts.0 | Узел сети name or pattern (e.g. `Zabbix*`).<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели.<br><br>Поведение параметра:<br>- *required* if configuring *Overrides* |
| *Шаблоны элементов данных* | 1 | or.0.items.0 | Элемент данных name or pattern (e.g. `*: Number of processed *values per second`).<br>При настройке виджета на шаблонной панели следует задавать только шаблоны для элементов данных, настроенных на шаблоне.<br><br>Поведение параметра:<br>- *required* if configuring *Overrides* |
| *Базовый цвет* | 1 | or.0.color | Шестнадцатеричный код цвета (например, FF0000). |
| *Ширина* | 0 | or.0.width | Возможные значения: от 1 до 10. |
| *Отрисовка* | 0 | or.0.type | 0 - Линия;<br>1 - Точки;<br>2 - Лестница;<br>3 - Столбец. |
| *Прозрачность* | 0 | or.0.transparency | Возможные значения: от 1 до 10. |
| *Заливка* | 0 | or.0.fill | Возможные значения: от 1 до 10. |
| *Размер точки* | 0 | or.0.pointsize | Возможные значения: от 1 до 10. |
| *Отсутствующие данные* | 0 | or.0.missingdatafunc | 0 - Нет;<br>1 - Связанные;<br>2 - Считать как 0;<br>3 - Последнее известное. |
| *Ось Y* | 0 | or.0.axisy | 0 - Левая;<br>1 - Правая. |
| *Сдвиг времени* | 1 | or.0.timeshift | Допустимая строка времени (например, 3600, 1h и т. д.).<br>Можно использовать суффиксы времени. Отрицательные значения допускаются. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Graph*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Graph*

Настройте виджет *Graph* следующим образом:

- 2 набора данных в общей сложности для 9 элементов данных на 1 узле сети.
- Первый набор данных имеет тип "Item list" и состоит из 3 элементов, которые отображаются линиями разного цвета, но одинаковой ширины, прозрачности и заливки.
- Второй набор данных имеет тип "Item patterns", состоит из 6 элементов, имеет настроенную агрегацию и отображается линией с пользовательскими цветом, шириной, прозрачностью и заливкой.
- Второй набор данных также имеет пользовательскую метку набора данных.
- Данные на графике отображаются за период последних 3 часов.
- Проблемы на графике отображаются только для настроенных элементов данных.
- График имеет две оси Y, при этом правая ось Y отображает значения только для второго набора данных.
- Легенда графика отображает настроенные элементы данных в 4 строках, а также минимальные, максимальные и средние значения наборов данных.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
```

```
{
    "widgets": [
        {
            "type": "svggraph",
            "name": "Graph",
            "x": 0,
            "y": 0,
            "width": 36,
            "height": 5,
            "view_mode": 0,
            "fields": [
                {
                    "type": 0,
                    "name": "ds.0.dataset_type",
                    "value": 0
                },
                {
                    "type": 4,
                    "name": "ds.0.itemids.0",
                    "value": 23264
                },
                {
                    "type": 1,
                    "name": "ds.0.color.0",
                    "value": "FF0000"
                },
                {
                    "type": 4,
                    "name": "ds.0.itemids.1",
                    "value": 23269
                },
                {
                    "type": 1,
                    "name": "ds.0.color.1",
                    "value": "BF00FF"
                },
                {
                    "type": 4,
                    "name": "ds.0.itemids.2",
                    "value": 23257
                },
                {
                    "type": 1,
                    "name": "ds.0.color.2",
                    "value": "0040FF"
                },
                {
                    "type": 0,
                    "name": "ds.0.width",
                    "value": 3
                },
                {
                    "type": 0,
                    "name": "ds.0.transparency",
                    "value": 3
                },
                {
                    "type": 0,
                    "name": "ds.0.fill",
                    "value": 1
                },
                {
```

```
                "type": 1,
                "name": "ds.1.hosts.0",
                "value": "Zabbix server"
            },
            {
                "type": 1,
                "name": "ds.1.items.0",
                "value": "*: Number of processed *values per second"
            },
            {
                "type": 1,
                "name": "ds.1.color",
                "value": "000000"
            },
            {
                "type": 0,
                "name": "ds.1.transparency",
                "value": 0
            },
            {
                "type": 0,
                "name": "ds.1.fill",
                "value": 0
            },
            {
                "type": 0,
                "name": "ds.1.axisy",
                "value": 1
            },
            {
                "type": 0,
                "name": "ds.1.aggregate_function",
                "value": 3
            },
            {
                "type": 1,
                "name": "ds.1.aggregate_interval",
                "value": "1m"
            },
            {
                "type": 0,
                "name": "ds.1.aggregate_grouping",
                "value": 1
            },
            {
                "type": 1,
                "name": "ds.1.data_set_label",
                "value": "Number of processed values per second"
            },
            {
                "type": 0,
                "name": "graph_time",
                "value": 1
            },
            {
                "type": 1,
                "name": "time_period.from",
                "value": "now-3h"
            },
            {
                "type": 1,
                "name": "time_period.to",
```

```
                                "value": "now"
                            },
                            {
                                "type": 0,
                                "name": "legend_statistic",
                                "value": 1
                            },
                            {
                                "type": 0,
                                "name": "legend_lines",
                                "value": 4
                            },
                            {
                                "type": 0,
                                "name": "show_problems",
                                "value": 1
                            },
                            {
                                "type": 1,
                                "name": "reference",
                                "value": "YZABC"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

9 График (классический)

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели мониторинга позволяют настраивать виджет *Graph (classic)* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели мониторинга. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Graph (classic)*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Graph (classic)*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Source* | 0 | source_type | 0 - *(по умолчанию)* Graph;<br>1 - Simple graph. |
| *Graph* | 6 | graphid.0 | ID Graph.<br><br>Поведение параметра:<br>- *required* если *Source* установлено в ”Graph” |
| *Graph (Widget)* | 1 | graphid._reference | Вместо ID Graph:<br>`ABCDE._graphid` - задайте совместимый виджет (с параметром *Reference*, установленным в ”ABCDE”) в качестве источника данных для графиков.<br><br>Поведение параметра:<br>- *required* если *Source* установлено в ”Simple graph” и *Graph* не задан |
| *Item* | 4 | itemid.0 | ID элемента данных.<br><br>Поведение параметра:<br>- *required* если *Source* установлено в ”Simple graph” и *Item (Widget)* не задан |
| *Item (Widget)* | 1 | itemid._reference | Вместо ID элемента данных:<br>`ABCDE._itemid` - задайте совместимый виджет (с параметром *Reference*, установленным в ”ABCDE”) в качестве источника данных для элементов данных.<br><br>Поведение параметра:<br>- *required* если *Source* установлено в ”Simple graph” и *Item* не задан |
| *Time period* | 1 | time_period._reference | `DASHBOARD._timeperiod` - задайте селектор *Time period* в качестве источника данных;<br>`ABCDE._timeperiod` - задайте совместимый виджет (с параметром *Reference*, установленным в ”ABCDE”) в качестве источника данных.<br><br>По умолчанию: `DASHBOARD._timeperiod`<br><br>Кроме того, можно задать период времени только в параметрах *From* и *To*. |

| Parameter | type | name | value |
|---|---|---|---|
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported* если *Time period* не задан<br>- *required* если задан time_period.to |
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported* если *Time period* не задан<br>- *required* если задан time_period.from |
| *Show legend* | 0 | show_legend | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Override host* | 1 | override_hostid._reference | ABCDE._hostid - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для узлов сети;<br>DASHBOARD._hostid - задайте селектор *Host* на панели управления в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели управления. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, ABCDE или JBPNL). Это значение должно быть уникальным в пределах панели управления, которой принадлежит виджет.<br><br>Поведение параметра:<br>- *required* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Graph (classic)*. Дополнительные сведения о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Graph (classic)*

Настройте виджет *Graph (classic)*, который отображает простой график для элемента данных "42269".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

10 Прототип графика

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели мониторинга позволяют настраивать виджет *Graph prototype* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели мониторинга. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Graph prototype*, см. описание поведения параметров в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Graph prototype*.

| Parameter | | type | name |
|---|---|---|---|
| *Refresh interval* | | 0 | rf_rate |
| *Source* | | 0 | source_type |
| *Graph prototype* | | 7 | graphid.0 |
| *Item prototype* | | 5 | itemid.0 |
| *Time period* | | 1 | time_period._reference |
| | *From* | 1 | time_period.from |
| | *To* | 1 | time_period.to |
| *Show legend* | | 0 | show_legend |
| *Override host* | | 1 | override_hostid._reference |
| *Columns* | | 0 | columns |
| *Rows* | | 0 | rows |
| *Reference* | | 1 | reference |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Graph prototype*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Graph prototype*

Настройте виджет *Graph prototype*, который отображает сетку из 3 графиков (3 столбца, 1 строка), созданных из прототипа элемента данных (ID: ”42316”) с помощью низкоуровневого обнаружения.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

11 Honeycomb

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Honeycomb* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Honeycomb*, см. поведение параметров, описанное в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Honeycomb*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host tags* | | | |
| *Evaluation type* | 0 | evaltype_host | 0 - *(по умолчанию)* And/Or;<br>2 - Or.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Tag name* | 1 | host_tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Operator* | 0 | host_tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Tag value* | 1 | host_tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Item pat-terns* | 1 | items.0 | Имя или шаблон элемента данных.<br><br>Примечание: чтобы настроить несколько шаблонов элементов данных, создайте объект поля виджета панели для каждого шаблона элемента данных, увеличивая номер в имени свойства.<br><br>Поведение параметра:<br>- *обязательно* |
| *Item tags* | | | |
| *Evaluation type* | 0 | evaltype_item | 0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| *Tag name* | 1 | item_tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Item tags* |

| Parameter | type | name | value |
|---|---|---|---|
| *Operator* | 0 | item_tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Item tags* |
| *Tag value* | 1 | item_tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Item tags* |
| *Show hosts in maintenance* | 0 | maintenance | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Show* | 0 | show.0 | 1 - Основная метка;<br>2 - Вторичная метка.<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства.<br><br>По умолчанию: 1, 2. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *обязательно* |

Расширенная настройка

Следующие параметры расширенной настройки поддерживаются для виджета *Honeycomb*.

**Note:**
Число в имени свойства *Thresholds* (например, thresholds.0.color) указывает на позицию порога в списке, отсортированном по возрастанию. Однако если пороги настроены в другом порядке, значения будут отсортированы по возрастанию после обновления конфигурации виджета во веб-интерфейсе Zabbix (например, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

| Parameter | type | name | value |
|---|---|---|---|
| ***Primary label*** | | | |
| *Type* | 0 | primary_label_type | 0 - *(по умолчанию)* Текст;<br>1 - Значение. |

| Parameter | type | name | value |
|---|---|---|---|
| *Text* | 1 | primary_label | Любое строковое значение, включая макросы.<br>Поддерживаемые макросы: {HOST.*}, {ITEM.*}, {INVENTORY.*}, пользовательские макросы.<br><br>По умолчанию: {HOST.NAME}<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение "Text" |
| *Decimal places* | 0 | primary_label_decimal_places | Допустимые значения: от 0 до 6.<br><br>По умолчанию: 2.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение "Value" |
| *Size* (type) | 0 | primary_label_size_type | 0 - *(по умолчанию)* Авто;<br>1 - Пользовательский. |
| *Size* | 0 | primary_label_size | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 20.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Size* (type) задано значение "Custom" |
| *Bold* | 0 | primary_label_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Color* | 1 | primary_label_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: зависит от `theme` в объекте Settings и объекте User:<br>1F2C33 для "blue-theme" или "hc-light";<br>EEEEEE для "dark-theme" или "hc-dark". |
| *Units* (checkbox) | 0 | primary_label_units_show | 0 - Отключено;<br>1 - *(по умолчанию)* Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение "Value" |
| *Units* (value) | 1 | primary_label_units | Любое строковое значение.<br><br>"" (пусто)<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение "Value" и для *Units* (checkbox) задано значение "Enabled" |
| *Position* | 0 | primary_label_units_pos | 0 - Перед значением;<br>1 - *(по умолчанию)* После значения.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение "Value" и для *Units* (checkbox) задано значение "Enabled"<br><br>Этот параметр игнорируется, если для него задано одно из следующих единиц времени: unixtime, uptime, s. |
| ***Secondary label*** | | | |
| *Type* | 0 | secondary_label_type | 0 - Текст;<br>1 - *(по умолчанию)* Значение. |
| *Text* | 1 | secondary_label | Любое строковое значение, включая макросы.<br>Поддерживаемые макросы: {HOST.*}, {ITEM.*}, {INVENTORY.*}, пользовательские макросы.<br><br>По умолчанию: {{ITEM.LASTVALUE}.fmtnum(2)}<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение "Text" |

| Parameter | type | name | value |
|---|---|---|---|
| *Decimal places* | 0 | secondary_label_decimal_places | Допустимые значения: от 0 до 6.<br><br>По умолчанию: 2.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение ”Value” |
| *Size* (type) | 0 | secondary_label_size_type | 0 - *(по умолчанию)* Авто;<br>1 - Пользовательский. |
| *Size* | 0 | secondary_label_size | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 30.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Size* (type) задано значение ”Custom” |
| *Bold* | 0 | secondary_label_bold | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Color* | 1 | secondary_label_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: зависит от theme в объекте Settings и объекте User:<br>1F2C33 для ”blue-theme” или ”hc-light”;<br>EEEEEE для ”dark-theme” или ”hc-dark”. |
| *Units* (checkbox) | 0 | secondary_label_units_show | 0 - Отключено;<br>1 - *(по умолчанию)* Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение ”Value” |
| *Units* (value) | 1 | secondary_label_units | Любое строковое значение.<br><br>"" (пусто)<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение ”Value” и для *Units* (checkbox) задано значение ”Enabled” |
| *Position* | 0 | secondary_label_position | 0 - Перед значением;<br>1 - *(по умолчанию)* После значения.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Type* задано значение ”Value” и для *Units* (checkbox) задано значение ”Enabled”<br><br>Этот параметр игнорируется, если для него задано одно из следующих единиц времени: unixtime, uptime, s. |
| ***Background color*** | | | |
| *Background color* | 1 | bg_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: зависит от theme в объекте Settings и объекте User:<br>D9E7ED для ”blue-theme”;<br>3D5059 для ”dark-theme”;<br>AAD7E9 для ”hc-light”;<br>335463 для ”hc-dark”. |
| ***Thresholds*** | | | |
| *Color interpolation* | 0 | interpolation | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Color* | 1 | thresholds.0.color | Шестнадцатеричный код цвета (например, FF0000). |
| *Threshold* | 1 | thresholds.0.threshold | Любое числовое значение. Поддерживаются суффиксы (например, ”1d”, ”2w”, ”4K”, ”8G”). |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга

для виджета *Honeycomb*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Honeycomb*

Настройте виджет *Honeycomb*, который отображает загрузку процессов сервера Zabbix. Кроме того, измените основную метку ячеек honeycomb и визуально настройте виджет с помощью пороговых значений.

Запрос:

```
{
    "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
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

12 Доступность узла сети

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Доступность узла сети* в методах `dashboard.create` и `dashboard.update`.

Параметры

Следующие параметры поддерживаются для виджета *Доступность узлов сети*.

| Параметр | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |
| *Группы узлов сети* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Группы узлов сети (виджет)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - укажите совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Тип интерфейса* | 0 | interface_type.0 | 0 - Нет;<br>1 - Zabbix agent (пассивные проверки);<br>2 - SNMP;<br>3 - IPMI;<br>4 - JMX;<br>5 - Zabbix agent (активные проверки).<br><br>По умолчанию: 1, 2, 3, 4, 5 (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства. |
| *Макет* | 0 | layout | 0 - *(по умолчанию)* Горизонтальный;<br>1 - Вертикальный. |
| *Показывать узлы сети в обслуживании* | 0 | maintenance | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Показывать только итоги* | 0 | only_totals | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Host availability*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Host availability*

Настройте виджет *Host availability*, который отображает информацию о доступности (в вертикальном расположении) для узлов сети в группе узлов сети "4" с настроенными интерфейсами "Zabbix agent" и "SNMP".

Request:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
```

```
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

13. Карточка узла сети

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Host card* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Host card*, пожалуйста, ознакомьтесь с поведением параметров, описанным в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Host card*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Не обновлять;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host* | 3 | hostid.0 | ID узла сети.<br><br>Поведение параметра:<br>- *required* если *Host (Widget/Dashboard)* не задан<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host (Widget/Dashboard)* | 1 | hostid._reference | Вместо ID узла сети:<br>`DASHBOARD.hostid` - задает селектор *Host* в качестве источника данных для узла сети;<br>`ABCDE._hostid` - задает совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для узла сети.<br><br>Поведение параметра:<br>- *required* если *Host* не задан<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Show suppressed problems* | 0 | show_suppressed | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |

| Parameter | type | name | value |
|---|---|---|---|
| *Show* | 0 | sections.0 | 0 - Группы узлов сети;<br>1 - Описание;<br>2 - Мониторинг;<br>3 - Доступность;<br>4 - Контролируется;<br>5 - Шаблоны;<br>6 - Инвентаризация;<br>7 - Теги.<br><br>Примечание: Число в имени свойства указывает порядок раздела в списке разделов. Чтобы настроить несколько разделов, создайте объект поля виджета панели для каждого раздела, увеличивая число в имени свойства. |

Следующие параметры поддерживаются, если для *Show* установлено значение ”Inventory”.

| Parameter | type | name | value |
|---|---|---|---|
| *Inventory fields* | 0 | inventory.0 | ID инвентаризации.<br><br>Примечание: Чтобы настроить несколько полей инвентаризации, создайте объект поля виджета панели для каждого поля инвентаризации, увеличивая число в имени свойства. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Host card*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Host card*

Настройте виджет *Host card*, который отображает следующие разделы: ”Monitoring”, ”Availability”, ”Monitored by”, ”Inventory” и ”Tags”.

Request:

```
{
    "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
}
```


Response:


```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

14. Навигатор по узлам сети

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Host navigator* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Host navigator*, см. поведение параметров, описанное в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Host navigator*.

| Параметр | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Группы узлов сети* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Группы узлов сети (виджет)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Параметр | type | name | value |
|---|---|---|---|
| *Шаблоны узлов сети* | 1 | hosts.0 | Имя или шаблон узла сети.<br><br>Примечание: чтобы настроить несколько шаблонов узлов сети, создайте объект поля виджета панели для каждого шаблона узла сети, увеличивая номер в имени свойства. Для нескольких шаблонов узлов сети параметр *Группы узлов сети* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат узлы сети, соответствующие настроенным шаблонам узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Состояние узла сети* | 0 | status | -1 - *(по умолчанию)* Любое;<br>0 - Включен;<br>1 - Отключен.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Теги узлов сети* | | | |
| *Тип оценки* | 0 | host_tags_evaltype | 0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Имя тега* | 1 | host_tags.0.tag | Любое строковое значение.<br><br>Примечание: номер в имени свойства соответствует порядку тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Теги узлов сети*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Оператор* | 0 | host_tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: номер в имени свойства соответствует порядку тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Теги узлов сети*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Значение тега* | 1 | host_tags.0.value | Любое строковое значение.<br><br>Примечание: номер в имени свойства соответствует порядку тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Теги узлов сети*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Параметр | type | name | value |
|---|---|---|---|
| *Серьезность* | 0 | severities.0 | 0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>По умолчанию: пусто (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства. |
| *Показывать узлы сети в обслуживании* | 0 | maintenance | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Показывать проблемы* | 0 | show_problems | 0 - Все;<br>1 - *(по умолчанию)* Без подавленных;<br>2 - Нет. |
| *Группировать по* | | | |
| *Атрибут* | 0 | group_by.0.attribute | 0 - Группа узлов сети;<br>1 - Значение тега;<br>2 - Серьезность.<br><br>Примечание: номер в имени свойства соответствует порядку атрибута в списке атрибутов группировки.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Группировать по* |
| *Значение* | 1 | group_by.0.tag_name | Любое строковое значение.<br><br>Примечание: номер в имени свойства соответствует атрибуту группировки, заданному в параметре *Атрибут*.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Группировать по*, если для *Атрибут* установлено значение ”Значение тега” |
| *Лимит узлов сети* | 0 | show_lines | Допустимые значения: от 1 до 9999.<br><br>По умолчанию: 100.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Ссылка* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *обязательно* |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *Host navigator*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Host navigator*

Настройте виджет *Host navigator*, который отображает узлы сети, сгруппированные сначала по их группе узлов сети, а затем по значению тега ”city”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

15 Карточка элемента данных

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Item card* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Item card*, обратитесь к описанию поведения параметров в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Item card*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Item* | 4 | itemid.0 | ID элемента данных.<br><br>Поведение параметра:<br>- *required* если *Item (Widget)* не задан |
| *Item (Widget)* | 1 | itemid._reference | Вместо ID элемента данных:<br>`ABCDE._itemid` - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для элемента данных.<br><br>Поведение параметра:<br>- *required* если *Item* не задан |

| Parameter | type | name | value |
|---|---|---|---|
| *Show* | 0 | sections.0 | 0 - Описание;<br>1 - Текст ошибки;<br>2 - Метрики;<br>3 - Последние данные;<br>4 - Тип информации;<br>5 - Триггеры;<br>6 - Интерфейс узла сети;<br>7 - Тип;<br>8 - Инвентаризация узла сети;<br>9 - Теги.<br><br>Примечание: число в имени свойства соответствует порядку раздела в списке разделов. Чтобы настроить несколько разделов, создайте объект поля виджета панели для каждого раздела, увеличивая число в имени свойства. |
| *Override host* | 1 | override_hostid._reference | `ABCDE._hostid` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети;<br>`DASHBOARD._hostid` - задайте селектор *Host* панели в качестве источника данных для узлов сети.<br><br>По умолчанию: `""` (пусто)<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

Спарклайн

Следующие параметры поддерживаются, если для *Show* установлено значение "Latest data".

| Parameter | type | name | value |
|---|---|---|---|
| *Width* | 0 | sparkline.width | Допустимые значения: от 0 до 10.<br><br>По умолчанию: 1. |
| *Fill* | 0 | sparkline.fill | Допустимые значения: от 0 до 10.<br><br>По умолчанию: 3. |
| *Color* | 1 | sparkline.color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>По умолчанию: `42A5F5`. |
| *Time period* | 1 | sparkline.time_period._reference | `DASHBOARD._timeperiod` - задает селектор *Time period* в качестве источника данных;<br>`ABCDE._timeperiod` - задает совместимый виджет (с параметром `reference`, равным `ABCDE`) в качестве источника данных.<br><br>По умолчанию: `""` (пусто)<br><br>Кроме того, вы можете задать период времени только в параметрах *From* и *To*. |
| *From* | 1 | sparkline.time_period.from | Допустимая строка времени в абсолютном формате (`YYYY-MM-DD hh:mm:ss`) или относительном синтаксисе времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>По умолчанию: now-1h.<br><br>Поведение параметра:<br>- *supported*, если *Time period* не задан<br>- *required*, если задан `sparkline.time_period.to` |

| Parameter | type | name | value |
|---|---|---|---|
| *To* | 1 | sparkline.time_period.to | Допустимая строка времени в абсолютном формате (`YYYY-MM-DD hh:mm:ss`) или относительном синтаксисе времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>По умолчанию: now.<br><br>Поведение параметра:<br>- *supported*, если *Time period* не задан<br>- *required*, если задан `time_period.from` |
| *History data* | 0 | sparkline.history | 0 - *(по умолчанию)* Авто;<br>1 - История;<br>2 - Тренды. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Item card*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Карточка элемента данных*

Настройте виджет *Карточка элемента данных*, который отображает следующие разделы: ”Description”, ”Latest data”, ”Triggers” и ”Tags”.

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

См. также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

16 История элемента данных

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Item history* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Item history*, см. описание поведения параметров в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Item history*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Layout* | 0 | layout | 0 - *(по умолчанию)* Горизонтально;<br>1 - Вертикально. |
| *Columns* (see below) | | | |
| *Show lines* | 0 | show_lines | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 25. |
| *Override host* | 1 | override_hostid._reference | `ABCDE._hostid` - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для узлов сети;<br>`DASHBOARD._hostid` - задайте селектор *Host* на панели в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Advanced configuration* (see below) | | | |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, которой принадлежит виджет.<br><br>Поведение параметра:<br>- *required* |

Столбцы

Столбцы имеют общие параметры и дополнительные параметры в зависимости от настройки параметра *Item*.

**Note:**
Для всех параметров, связанных со столбцами, число в имени свойства (например, columns.0.name) указывает на столбец, для которого настроен параметр.

Следующие параметры поддерживаются для всех столбцов.

| Parameter | type | name | value |
|---|---|---|---|
| *Name* | 1 | columns.0.name | Любое строковое значение.<br><br>Поведение параметра:<br>- *required* |
| *Item* | 4 | columns.0.itemid | ID элемента данных.<br><br>При настройке виджета на шаблонной панели следует указывать только элементы данных, настроенные на шаблоне.<br><br>Поведение параметра:<br>- *required* |
| *Base color* | 1 | columns.0.base_color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>По умолчанию: "" (пусто). |

Следующие параметры столбца поддерживаются, если настроенный *элемент данных* является элементом числового типа.

| Parameter | type | name | value |
|---|---|---|---|
| *Display* | 0 | columns.0.display | 1 - *(по умолчанию)* Как есть;<br>2 - Bar;<br>3 - Indicators. |
| *Min* | 1 | columns.0.min | Любое числовое значение.<br><br>Поведение параметра:<br>- *supported* если *Display* установлено в ”Bar” или ”Indicators” |
| *Max* | 1 | columns.0.max | Любое числовое значение.<br><br>Поведение параметра:<br>- *supported* если *Display* установлено в ”Bar” или ”Indicators” |
| *Thresholds* | | | |
| *Color* | 1 | columns.0.thresholds.color | Шестнадцатеричный код цвета (например, FF0000). |
| *Threshold* | 1 | columns.0.thresholds.threshold | Любое числовое значение. Поддерживаются суффиксы (например, ”1d”, ”2w”, ”4K”, ”8G”). |
| *History data* | 0 | columns.0.history | 0 - *(по умолчанию)* Авто;<br>1 - История;<br>2 - Тренды. |

Поддерживаются следующие параметры столбца, если настроенный *элемент данных* имеет тип character, text или log.

| Parameter | type | name | value |
|---|---|---|---|
| *Highlights* | | | |
| *Highlight* | 1 | columns.0.highlights.color | Шестнадцатеричный код цвета (например, FF0000). |
| *Threshold* | 1 | columns.0.highlights.pattern | Любое регулярное выражение. |
| *Display* | 0 | columns.0.display | 1 - *(по умолчанию)* Как есть;<br>4 - HTML;<br>5 - Одна строка. |
| *Single line* | 0 | columns.0.max_length | Возможные значения: от 1 до 500.<br><br>По умолчанию: 100.<br><br>Поведение параметра:<br>- *supported*, если для *Display* установлено значение ”Single line” |
| *Use monospace font* | 0 | columns.0.monospace_font | 0 - *(по умолчанию)* Использовать шрифт по умолчанию;<br>1 - Использовать моноширинный шрифт. |
| *Display local time* | 0 | columns.0.local_time | 0 - *(по умолчанию)* Отображать метку времени;<br>1 - Отображать местное время.<br><br>Поведение параметра:<br>- *supported*, если для *Item* установлено значение элемента данных типа log, а для *Show timestamp* установлено значение ”Enabled” |

Поддерживаются следующие параметры столбца, если настроенный *элемент данных* является элементом данных двоичного типа.

| Parameter | type | name | value |
|---|---|---|---|
| *Show thumbnail* | 1 | columns.0.show_thumbnail | 0 - *(default)* Disabled;<br>1 - Enabled. |

Расширенная настройка

Следующие параметры расширенной настройки поддерживаются для виджета *Item history*.

| Parameter | type | name | value |
|---|---|---|---|
| *New values* | 0 | sortorder | 0 - *(по умолчанию)* Сверху;<br>1 - Снизу. |
| *Show timestamp* | 0 | show_timestamp | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Show column header* | 0 | show_column_header | 0 - Выкл.;<br>1 - Горизонтально;<br>2 - *(по умолчанию)* Вертикально. |
| *Time period* | 1 | time_period._reference | DASHBOARD._timeperiod - задает *селектор периода времени* в качестве источника данных;<br>ABCDE._timeperiod - задает совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных.<br><br>По умолчанию: DASHBOARD._timeperiod<br><br>Кроме того, вы можете задать период времени только в параметрах *From* и *To*. |
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если *Time period* не задан<br>- *обязателен*, если задан time_period.to |
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если *Time period* не задан<br>- *обязателен*, если задан time_period.from |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Item history*. Дополнительную информацию о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Item history*

Настройте виджет *Item history*, который отображает последние данные для двух числовых элементов данных "42269" и "42270". Кроме того, настройте отображение столбцов элементов данных по вертикали, а имен столбцов — по горизонтали; ограничьте отображение 15 строками данных и добавьте отдельный столбец с временной меткой.

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

17 Навигатор элементов данных

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Item navigator* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Item navigator*, см. описание поведения параметров в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Item navigator*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostid` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostid` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host tags* | | | |
| *Evaluation type* | 0 | host_tags_evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Tag name* | 1 | host_tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Operator* | 0 | host_tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Tag value* | 1 | host_tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Item patterns* | 1 | items.0 | Имя или шаблон элемента данных.<br><br>Примечание: чтобы настроить несколько шаблонов элементов данных, создайте объект поля виджета панели для каждого шаблона элемента данных, увеличивая номер в имени свойства. |
| *Item tags* | | | |
| *Evaluation type* | 0 | item_tags_evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| *Tag name* | 1 | item_tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Item tags* |
| *Operator* | 0 | item_tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Item tags* |
| *Tag value* | 1 | item_tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Item tags* |
| *State* | 0 | state | -1 - *(по умолчанию)* Все;<br>0 - Нормальное;<br>1 - Не поддерживается. |
| *Show problems* | 0 | show_problems | 0 - Все;<br>1 - *(по умолчанию)* Без подавленных;<br>2 - Нет. |
| *Group by* | | | |

| Parameter | type | name | value |
|---|---|---|---|
| *Attribute* | 0 | group_by.0.attribute | 0 - Группа узлов сети;<br>1 - Имя узла сети;<br>2 - Значение тега узла сети;<br>3 - Значение тега элемента данных.<br><br>Примечание: число в имени свойства указывает порядок атрибута в списке атрибутов группировки.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Group by* |
| *Value* | 1 | group_by.0.tag_name | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает на атрибут группировки, заданный в параметре *Attribute*.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Group by* и для *Attribute* задано значение ”Host tag value” или ”Item tag value” |
| *Item limit* | 0 | show_lines | Допустимые значения: от 1 до 9999.<br><br>По умолчанию: 100. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *обязательно* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Item navigator*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Item navigator*

Настройте виджет *Item navigator*, который отображает до 1000 элементов данных, сгруппированных по узлу сети, а затем по значению тега элемента данных ”component”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

18 Значение элемента данных

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Item value* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Item value*, ознакомьтесь с поведением параметров, описанным в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Item value*.

| Parameter | | type | name |
|---|---|---|---|
| *Refresh interval* | | 0 | rf_rate |
| *Item* | | 4 | itemid.0 |
| | *Item (Widget)* | 1 | itemid._reference |
| *Show* | | 0 | show.0 |
| *Override host* | | 1 | override_hostid._reference |

Расширенная настройка

Следующие параметры расширенной настройки поддерживаются для виджета *Значение элемента данных*.

**Note:**
Число в имени свойства *Thresholds* (например, thresholds.0.color) указывает на позицию порога в списке, отсортированном по возрастанию. Однако если пороги настроены в другом порядке, значения будут отсортированы по возрастанию после обновления конфигурации виджета во веб-интерфейсе Zabbix (например, `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

| Parameter | type | name | value |
|---|---|---|---|
| *Цвет фона* | 1 | bg_color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>По умолчанию: `""` (пусто). |
| *Пороги* | | | |
| *Цвет* | 1 | thresholds.0.color | Шестнадцатеричный код цвета (например, `FF0000`). |
| *Порог* | 1 | thresholds.0.threshold | Любое строковое значение. |
| *Функция агрегации* | 0 | aggregate_function | 0 - *(по умолчанию)* не используется;<br>1 - min;<br>2 - max;<br>3 - avg;<br>4 - count;<br>5 - sum;<br>6 - first;<br>7 - last. |
| *Период времени* | 1 | time_period._reference | `DASHBOARD._timeperiod` - использовать *селектор периода времени* в качестве источника данных;<br>`ABCDE._timeperiod` - использовать совместимый виджет (со значением параметра `reference`, равным `ABCDE`) в качестве источника данных.<br><br>По умолчанию: `DASHBOARD._timeperiod`<br><br>Кроме того, можно задать период времени только в параметрах *From* и *To*.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Функция агрегации* установлено значение ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” |

| Parameter | type | name | value |
|---|---|---|---|
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если *Период времени* не задан и для *Функция агрегации* установлено значение "min", "max", "avg", "count", "sum", "first", "last"<br>- *обязательно*, если задан time_period.to |
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если *Период времени* не задан и для *Функция агрегации* установлено значение "min", "max", "avg", "count", "sum", "first", "last"<br>- *обязательно*, если задан time_period.from |
| *Исторические данные* | 0 | history | 0 - *(по умолчанию)* Auto;<br>1 - History;<br>2 - Trends. |

Описание

Следующие параметры расширенной конфигурации поддерживаются, если для *Show* установлено значение "Description".

| Parameter | type | name | value |
|---|---|---|---|
| *Description* | 1 | description | Любое строковое значение, включая макросы.<br>Поддерживаемые макросы: {HOST.*}, {ITEM.*}, {INVENTORY.*}, пользовательские макросы.<br><br>По умолчанию: {ITEM.NAME}. |
| *Horizontal position* | 0 | desc_h_pos | 0 - Слева;<br>1 - *(по умолчанию)* По центру;<br>2 - Справа. |
| *Vertical position* | 0 | desc_v_pos | Два или более элементов (Description, Value, Time) не могут иметь одинаковые *Horizontal position* и *Vertical position*.<br>0 - Сверху;<br>1 - Посередине;<br>2 - *(по умолчанию)* Снизу. |
| *Size* | 0 | desc_size | Два или более элементов (Description, Value, Time) не могут иметь одинаковые *Horizontal position* и *Vertical position*.<br>Возможные значения: от 1 до 100.<br><br>По умолчанию: 15. |
| *Bold* | 0 | desc_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Color* | 1 | desc_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |

Значение

Следующие параметры расширенной конфигурации поддерживаются, если для *Show* установлено значение "Value".

| Parameter | type | name | value |
|---|---|---|---|
| *Десятичные знаки* | | | |
| *Десятичные знаки* | 0 | decimal_places | Допустимые значения: от 1 до 10.<br><br>По умолчанию: 2. |
| *Размер* | 0 | decimal_size | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 35. |
| *Положение* | | | |
| *Горизонтальное положение* | 0 | value_h_pos | 0 - Слева;<br>1 - *(по умолчанию)* По центру;<br>2 - Справа.<br><br>Два или более элементов (Description, Value, Time) не могут иметь одинаковые значения *Horizontal position* и *Vertical position*. |
| *Вертикальное положение* | 0 | value_v_pos | 0 - Сверху;<br>1 - *(по умолчанию)* По центру;<br>2 - Снизу.<br><br>Два или более элементов (Description, Value, Time) не могут иметь одинаковые значения *Horizontal position* и *Vertical position*. |
| *Размер* | 0 | value_size | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 45. |
| *Полужирный* | 0 | value_bold | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Цвет* | 1 | value_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Единицы* | | | |
| *Единицы* (флажок) | 0 | units_show | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Единицы* (значение) | 1 | units | Любое строковое значение. |
| *Положение* | 0 | units_pos | 0 - Перед значением;<br>1 - Над значением;<br>2 - *(по умолчанию)* После значения;<br>3 - Под значением. |
| *Размер* | 0 | units_size | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 35. |
| *Полужирный* | 0 | units_bold | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Цвет* | 1 | units_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |

Время

Следующие параметры расширенной настройки поддерживаются, если для *Show* установлено значение ”Time”.

| Parameter | type | name | value |
|---|---|---|---|
| *Горизонтальное положение* | 0 | time_h_pos | 0 - Слева;<br>1 - *(по умолчанию)* По центру;<br>2 - Справа.<br><br>Два или более элементов (Description, Value, Time) не могут иметь одинаковые *Горизонтальное положение* и *Вертикальное положение*. |

| Parameter | type | name | value |
|---|---|---|---|
| *Вертикальное положение* | 0 | time_v_pos | 0 - *(по умолчанию)* Сверху;<br>1 - Посередине;<br>2 - Снизу.<br><br>Два или более элементов (Description, Value, Time) не могут иметь одинаковые *Горизонтальное положение* и *Вертикальное положение*. |
| *Размер* | 0 | time_size | Допустимые значения находятся в диапазоне от 1 до 100.<br><br>По умолчанию: 15. |
| *Полужирный* | 0 | time_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Цвет* | 1 | time_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |

Индикатор изменения

Следующие параметры расширенной конфигурации поддерживаются, если для *Показывать* установлено значение ”Индикатор изменения”.

| Parameter | type | name | value |
|---|---|---|---|
| *Цвет индикатора изменения* ↑ | 1 | up_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Цвет индикатора изменения* ↓ | 1 | down_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |
| *Цвет индикатора изменения* ↕ | 1 | updown_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: "" (пусто). |

Sparkline

Следующие параметры расширенной настройки поддерживаются, если для *Show* установлено значение ”Sparkline”.

| Parameter | type | name | value |
|---|---|---|---|
| *Width* | 0 | sparkline.width | Допустимые значения: от 0 до 10.<br><br>По умолчанию: 1. |
| *Fill* | 0 | sparkline.fill | Допустимые значения: от 0 до 10.<br><br>По умолчанию: 3. |
| *Color* | 1 | sparkline.color | Шестнадцатеричный код цвета (например, FF0000).<br><br>По умолчанию: 42A5F5. |
| *Time period* | 1 | sparkline.time_period.reference | DASHBOARD._timeperiod - использовать селектор *Time period* в качестве источника данных;<br>ABCDE._timeperiod - использовать совместимый виджет (с параметром reference, равным ABCDE) в качестве источника данных.<br><br>По умолчанию: DASHBOARD._timeperiod<br><br>Кроме того, можно задать период времени только в параметрах *From* и *To*. |

| Parameter | type | name | value |
|---|---|---|---|
| *From* | 1 | sparkline.time_period.from | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или в относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *required*, если задан sparkline.time_period.to |
| *To* | 1 | sparkline.time_period.to | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или в относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *required*, если задан sparkline.time_period.from |
| *History data* | 0 | sparkline.history | 0 - *(по умолчанию)* Auto;<br>1 - History;<br>2 - Trends. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Item value*. Дополнительные сведения о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Item value*

Настройте виджет *Item value*, который отображает значение элемента данных для элемента данных "42266" (доступность агента Zabbix). Кроме того, визуально доработайте виджет с помощью нескольких расширенных параметров, включая динамический цвет фона, который изменяется в зависимости от статуса доступности агента Zabbix.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
```

```
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

19 Карта

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Map* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Map*, ознакомьтесь с поведением параметров, описанным в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Map*.

| Параметр | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |
| *Карта* | 8 | sysmapid.0 | ID карты.<br><br>Поведение параметра:<br>- *обязателен*, если *Map (Widget)* не задан |
| *Map (Widget)* | 1 | sysmapid._reference | ABCDE._mapid - задайте виджет *Map navigation tree* (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для карт.<br><br>Поведение параметра:<br>- *обязателен*, если *Map* не задан |
| *Ссылка* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, ABCDE или JBPNL). Это значение должно быть уникальным в пределах веб-интерфейса, к которому принадлежит виджет.<br><br>Поведение параметра:<br>- *обязателен* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Map*. Дополнительную информацию о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Map*

Настройте виджет *Map*, который отображает карту "1".

Request:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "map",
                        "name": "Map",
                        "x": 0,
                        "y": 0,
                        "width": 54,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 8,
                                "name": "sysmapid.0",
                                "value": 1
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Настройка связанного виджета *Map*

Настройте виджет *Map*, который связан с виджетом *Map navigation tree*.

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "map",
                        "name": "Map",
                        "x": 0,
                        "y": 5,
                        "width": 54,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "sysmapid._reference",
                                "value": "ABCDE._mapid"
                            }
                        ]
                    },
                    {
                        "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": "ABCDE"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update
- Дерево навигации карт сетей

20 Дерево навигации по карте

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Map navigation tree* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Map navigation tree*, см. поведение параметров, описанное в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Map navigation tree*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Не обновлять;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |
| *Show unavailable maps* | 1 | show_unavailable | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах веб-интерфейса, к которому принадлежит виджет.<br><br>Parameter behavior:<br>- *required* |

Следующие параметры поддерживаются для настройки элементов дерева навигации по картам.

| Parameter | type | name | value |
|---|---|---|---|
| *Name* | 1 | navtree.1.name | Любое строковое значение.<br><br>Примечание: число в имени свойства задает номер элемента. |
| *Linked map* | 8 | navtree.1.sysmapid | ID Map.<br><br>Примечание: число в имени свойства указывает на элемент, с которым связана карта. |
| *Parameters for creating element hierarchy* | 0 | navtree.1.parent | Номер родительского элемента.<br><br>Примечание: число в имени свойства указывает на дочерний элемент. Значение свойства указывает на родительский элемент. |
| | 0 | navtree.1.order | Положение элемента в дереве навигации по картам.<br><br>Примечание: число в имени свойства указывает на номер элемента. Значение свойства указывает на положение элемента в дереве навигации по картам. Положение родительского элемента определяется в пределах всего дерева навигации по картам. Положение дочернего элемента определяется в пределах родительского элемента. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Map navigation tree*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Map navigation tree*

Настройте виджет *Map navigation tree*, который отображает следующее дерево навигации по карте:

- Элемент A
  - Элемент A1
  - Элемент A2
- Элемент B
  - Элемент B1 (содержит связанную карту "1", которую можно отобразить в связанном виджете *Map widget*)
  - Элемент B2
- Элемент C

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`
- Карта

21 Круговая диаграмма

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Pie chart* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Pie chart*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Для виджета *Pie chart* поддерживаются следующие параметры.

| Параметр | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |

Набор данных

Следующие параметры поддерживаются для настройки *набора данных*.

**Note:**
Первое число в имени свойства (например, ds.0.hosts.0, ds.0.items.0) обозначает конкретный набор данных, а второе число, если оно присутствует, обозначает настроенный узел сети или элемент данных.

| Параметр | type | name | value |
|---|---|---|---|
| *Тип набора данных* | 0 | ds.0.dataset_type | 0 - Список элементов данных;<br>1 - *(по умолчанию)* Шаблоны элементов данных. |
| *Элементы данных* | 4 | ds.0.itemids.0 | ID элемента данных.<br>При настройке виджета на шаблонной панели следует указывать только элементы данных, настроенные на шаблоне.<br><br>Примечание: чтобы настроить несколько элементов данных, создайте объект поля виджета панели для каждого элемента, увеличивая номер в имени свойства.<br><br>Поведение параметра:<br>- *обязательно*, если *Тип набора данных* установлен в "Список элементов данных" и *Элементы данных (Виджет)* не задан |
| *Элементы данных (Виджет)* | 1 | ds.0.itemids.0._reference | Вместо ID элемента данных:<br>`ABCDE._itemid` - укажите совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для элементов данных.<br><br>Примечание: чтобы настроить несколько виджетов, создайте объект поля виджета панели для каждого виджета, увеличивая номер в имени свойства.<br><br>Поведение параметра:<br>- *обязательно*, если *Тип набора данных* установлен в "Список элементов данных" и *Элементы данных* не заданы |
| *Цвет* | 1 | ds.0.color.0 | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>Поведение параметра:<br>- *обязательно*, если *Тип набора данных* установлен в "Список элементов данных" |
| *Тип элемента данных* | 0 | ds.0.type.0 | 0 - *(по умолчанию)* Обычный;<br>1 - Итоговый.<br><br>Значение "Итоговый" можно задать только для одного элемента во всей диаграмме.<br><br>Поведение параметра:<br>- *поддерживается*, если *Тип набора данных* установлен в "Список элементов данных" |
| *Шаблоны узлов сети* | 1 | ds.0.hosts.0 | Имя или шаблон узла сети (например, "Zabbix*").<br><br>Поведение параметра:<br>- *обязательно*, если *Тип набора данных* установлен в "Шаблоны элементов данных"<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Параметр | type | name | value |
|---|---|---|---|
| *Шаблоны элементов данных* | 1 | ds.0.items.0 | Имя или шаблон элемента данных (например, ”*: Number of processed *values per second”).<br><br>При настройке виджета на шаблонной панели следует указывать только шаблоны для элементов данных, настроенных на шаблоне.<br><br>Поведение параметра:<br>- *обязательно*, если *Тип набора данных* установлен в ”Шаблоны элементов данных” |
| *Цвет* | 1 | ds.0.color | Шестнадцатеричный код цвета (например, FF0000).<br><br>Поведение параметра:<br>- *поддерживается*, если *Тип набора данных* установлен в ”Шаблоны элементов данных” и *Цветовая палитра* не задана |
| *Цветовая палитра* | 0 | ds.0.color_palette | Индекс цветовой палитры.<br><br>Допустимые значения: от 0 до 11.<br><br>По умолчанию: 0.<br><br>Поведение параметра:<br>- *поддерживается*, если *Тип набора данных* установлен в ”Шаблоны элементов данных” и *Цвет* не задан |
| *Функция агрегации* | 0 | ds.0.aggregate_function | 1 - min;<br>2 - max;<br>3 - avg;<br>4 - count;<br>5 - sum;<br>6 - first;<br>7 - *(по умолчанию)* last. |
| *Агрегация набора данных* | 0 | ds.0.dataset_aggregation | 0 - *(по умолчанию)* none;<br>1 - min;<br>2 - max;<br>3 - avg;<br>4 - count;<br>5 - sum.<br><br>Поведение параметра:<br>- *поддерживается*, если *Тип элемента данных* установлен в ”Итоговый” |
| *Метка набора данных* | 1 | ds.0.data_set_label | Любое строковое значение.<br><br>По умолчанию: "" (пусто). |

Параметры отображения

Следующие параметры поддерживаются для настройки *Параметров отображения*.

| Параметр | type | имя | значение |
|---|---|---|---|
| *Выбор исторических данных* | 0 | source | 0 - *(по умолчанию)* Авто;<br>1 - История;<br>2 - Тренды. |
| *Отрисовка* | 0 | draw_type | 0 - *(по умолчанию)* Круговая диаграмма;<br>1 - Пончик. |

| Параметр | type | имя | значение |
|---|---|---|---|
| *Ширина* | 0 | width | 20 - 20% от радиуса;<br>30 - 30% от радиуса;<br>40 - 40% от радиуса;<br>50 - *(по умолчанию)* 50% от радиуса.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отрисовка* установлено значение ”Пончик” |
| *Ширина линии* | 0 | stroke | Возможные значения: от 0 до 10.<br><br>По умолчанию: 0.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отрисовка* установлено значение ”Пончик” |
| *Показывать итоговое значение* | 0 | total_show | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отрисовка* установлено значение ”Пончик” |
| *Размер* | 0 | value_size_type | 0 - *(по умолчанию)* Авто;<br>1 - Пользовательский.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать итоговое значение* установлено значение ”Включено” |
| *Размер* (значение для пользовательского размера) | 0 | value_size | Возможные значения: от 1 до 100.<br><br>По умолчанию: 20.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать итоговое значение* установлено значение ”Включено” |
| *Количество знаков после запятой* | 0 | decimal_places | Возможные значения: от 0 до 6.<br><br>По умолчанию: 2.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать итоговое значение* установлено значение ”Включено” |
| *Единицы* (флажок) | 0 | units_show | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать итоговое значение* установлено значение ”Включено” |
| *Единицы* (значение) | 1 | units | Любая строка.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Единицы* (флажок) установлено значение ”Включено” |
| *Полужирный* | 0 | value_bold | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать итоговое значение* установлено значение ”Включено” |

| Параметр | type | имя | значение |
|---|---|---|---|
| *Цвет* | 1 | value_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать итоговое значение* установлено значение "Включено" |
| *Промежуток между секторами* | 0 | space | Возможные значения: от 0 до 10.<br><br>По умолчанию: 1. |
| *Объединять сектора меньше N%* (флажок) | 0 | merge | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Объединять сектора меньше N%* (значение) | 0 | merge_percent | Возможные значения: от 1 до 10.<br><br>По умолчанию: 1.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Объединять сектора меньше N%* (флажок) установлено значение "Включено" |
| *Объединять сектора меньше N%* (цвет) | 1 | merge_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Объединять сектора меньше N%* (флажок) установлено значение "Включено" |

Период времени

Следующие параметры поддерживаются для настройки *Периода времени*.

| Parameter | type | name | value |
|---|---|---|---|
| *Период времени* | 1 | time_period._reference | DASHBOARD._timeperiod - задает селектор *Период времени* в качестве источника данных;<br>ABCDE._timeperiod - задает совместимый виджет (с параметром *Reference*, установленным в значение "ABCDE") в качестве источника данных.<br><br>По умолчанию: DASHBOARD._timeperiod<br><br>Кроме того, вы можете задать период времени только в параметрах *From* и *To*. |
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported* если *Период времени* не задан<br>- *required* если задан time_period.to |
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported* если *Период времени* не задан<br>- *required* если задан time_period.from |

Легенда

Следующие параметры поддерживаются для настройки *Legend*.

| Parameter | type | name | value |
|---|---|---|---|
| *Показывать легенду* | 0 | legend | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Показывать значение* | 0 | legend_value | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Показывать функцию агрегации* | 0 | legend_aggregation | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Строки* | 0 | legend_lines_mode | 0 - *(по умолчанию)* Фиксированное;<br>1 - Переменное.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Количество строк/ Максимальное количество строк* | 0 | legend_lines | Возможные значения: от 1 до 10.<br><br>По умолчанию: 1.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено” |
| *Количество столбцов* | 0 | legend_columns | Возможные значения: от 1 до 4.<br><br>По умолчанию: 4.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Показывать легенду* установлено значение ”Включено”, а для *Показывать значение* установлено значение ”Отключено” |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Pie chart*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *круговая диаграмма*

Настройте виджет *круговая диаграмма* следующим образом:

- 2 набора данных, всего 9 элементов данных на 1 узел сети.
- Первый набор данных имеет тип ”Item list” и состоит из 3 элементов, все с типом ”Normal” и отображаются разными цветами.
- Второй набор данных имеет тип ”Item patterns”, состоит из 6 элементов, для каждого элемента данных настроена агрегация и отображается пользовательским цветом.
- Второй набор данных также имеет пользовательскую метку набора данных.
- Данные на круговой диаграмме отображаются как кольцевая диаграмма с пользовательской шириной и общим значением с единицами измерения в центре.
- Данные на круговой диаграмме отображаются и агрегируются за пользовательский период времени — последние 3 часа.
- Легенда круговой диаграммы отображает настроенные элементы в 4 строках.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

22 Проблемные узлы сети

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Проблемные узлы сети* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Проблемные узлы сети*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Проблемные узлы сети*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Exclude host groups* | 2 | exclude_groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы исключить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem* | 1 | problem | Имя события проблемы (без учета регистра, полное имя или его часть). |

| Parameter | type | name | value |
|---|---|---|---|
| *Severity* | 0 | severities.0 | 0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>По умолчанию: пусто (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства. |
| *Problem tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Problem tags* |
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Problem tags* |
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно*, если настраивается *Problem tags* |
| *Show suppressed problems* | 0 | show_suppressed | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Hide groups without problems* | 0 | hide_empty_groups | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem display* | 0 | ext_ack | 0 - *(по умолчанию)* Все;<br>1 - Только не подтвержденные;<br>2 - Раздельно. |

| Parameter | type | name | value |
|---|---|---|---|
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *обязательно* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Проблемные узлы сети*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Problem hosts*

Настройте виджет *Problem hosts*, который отображает узлы сети из групп узлов сети "2" и "4", у которых есть проблемы с названием, содержащим строку "CPU", и со следующими уровнями серьезности: "Warning", "Average", "High", "Disaster".

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

23 проблемы

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Problems* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Problems*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Problems*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Show* | 0 | show | 1 - *(по умолчанию)* Текущие проблемы;<br>2 - История;<br>3 - Проблемы. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Exclude host groups* | 2 | exclude_groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы исключить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem* | 1 | problem | Имя события проблемы (без учета регистра, полное имя или его часть). |

| Parameter | type | name | value |
|---|---|---|---|
| *Severity* | 0 | severities.0 | 0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>По умолчанию: пусто (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства. |
| *Problem tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: номер в имени свойства соответствует порядку тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* при настройке *Problem tags* |
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: номер в имени свойства соответствует порядку тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* при настройке *Problem tags* |
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: номер в имени свойства соответствует порядку тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* при настройке *Problem tags* |
| *Show tags* | 0 | show_tags | 0 - *(по умолчанию)* Нет;<br>1 - 1;<br>2 - 2;<br>3 - 3. |
| *Tag name (format)* | 0 | tag_name_format | 0 - *(по умолчанию)* Полное;<br>1 - Сокращенное;<br>2 - Нет.<br><br>Поведение параметра:<br>- *supported* если *Show tags* установлено в ”1”, ”2” или ”3” |
| *Tag display priority* | 1 | tag_priority | Список тегов, разделенный запятыми.<br><br>Поведение параметра:<br>- *supported* если *Show tags* установлено в ”1”, ”2” или ”3” |

| Parameter | type | name | value |
|---|---|---|---|
| *Show operational data* | 0 | show_opdata | 0 - *(по умолчанию)* Нет;<br>1 - Отдельно;<br>2 - Вместе с именем проблемы. |
| *Show suppressed problems* | 0 | show_suppressed | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Acknowledgement status* | 0 | acknowledgement_status | 0 - *(по умолчанию)* все;<br>1 - Не подтверждено;<br>2 - Подтверждено. |
| *By me* | 0 | acknowledged_by_me | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Sort entries by* | 0 | sort_triggers | 1 - Уровень серьезности (по убыванию);<br>2 - Узел сети (по возрастанию);<br>3 - Время (по возрастанию);<br>4 - *(по умолчанию)* Время (по убыванию);<br>13 - Уровень серьезности (по возрастанию);<br>14 - Узел сети (по убыванию);<br>15 - Проблема (по возрастанию);<br>16 - Проблема (по убыванию).<br><br>Для всех значений, кроме ”Время (по убыванию)” и ”Время (по возрастанию)”, параметр *Show timeline* должен быть установлен в ”Disabled”.<br><br>Значения ”Узел сети (по возрастанию)” и ”Узел сети (по убыванию)” не поддерживаются при настройке виджета на шаблонной панели. |
| *Show timeline* | 0 | show_timeline | 0 - Отключено;<br>1 - *(по умолчанию)* Включено.<br><br>Поведение параметра:<br>- *supported* если *Sort entries by* установлено в ”Время (по убыванию)” или ”Время (по возрастанию)” |
| *Highlight whole row* | 0 | highlight_row | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Этот параметр не поддерживается в темах с высокой контрастностью.<br><br>Эта опция поддерживается начиная с Zabbix 7.4.3. |
| *Show lines* | 0 | show_lines | Допустимые значения: от 1 до 100.<br><br>По умолчанию: 25. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *required* |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *Problems*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Проблемы*

Настройте виджет *Проблемы*, который отображает проблемы для группы узлов сети ”4”, удовлетворяющие следующим условиям:

- Проблемы, у которых есть тег с именем ”scope”, содержащий значения ”performance” или ”availability”, или ”capacity”.
- Проблемы со следующими уровнями серьезности: ”Warning”, ”Average”, ”High”, ”Disaster”.

Кроме того, настройте виджет так, чтобы он отображал теги и оперативные данные.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

24 Проблемы по уровню серьезности

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Проблемы по степени важности* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Проблемы по степени важности*, пожалуйста, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Problems by severity*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Exclude host groups* | 2 | exclude_groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы исключить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem* | 1 | problem | Имя события проблемы (без учета регистра, полное имя или его часть). |
| *Severity* | 0 | severities.0 | 0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>По умолчанию: пусто (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства. |
| *Problem tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* при настройке *Problem tags* |

| Parameter | type | name | value |
|---|---|---|---|
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* при настройке *Problem tags* |
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* при настройке *Problem tags* |
| *Show* | 0 | show_type | 0 - *(по умолчанию)* Группы узлов сети;<br>1 - Итого.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели, и по умолчанию установлен в ”Итого”. |
| *Layout* | 0 | layout | 0 - *(по умолчанию)* Горизонтальный;<br>1 - Вертикальный.<br><br>Поведение параметра:<br>- *supported* если *Show* установлен в ”Totals” |
| *Show operational data* | 0 | show_opdata | 0 - *(по умолчанию)* Нет;<br>1 - Отдельно;<br>2 - Вместе с именем проблемы. |
| *Show suppressed problems* | 0 | show_suppressed | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Hide groups without problems* | 0 | hide_empty_groups | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *supported* если *Show* установлен в ”Host groups”<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem display* | 0 | ext_ack | 0 - *(по умолчанию)* Все;<br>1 - Только не подтвержденные;<br>2 - Раздельно. |
| *Show timeline* | 0 | show_timeline | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *required* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Проблемы по степени серьезности*. Дополнительные сведения о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Проблемы по степени важности*

Настройте виджет *Проблемы по степени важности*, который отображает общее количество проблем для всех групп узлов сети.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
```

```
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

25 Отчет SLA

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *SLA report* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *SLA report*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *SLA report*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - *(по умолчанию)* Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *SLA* | 10 | slaid.0 | ID SLA.<br><br>Поведение параметра:<br>- *required* |
| *Service* | 9 | serviceid.0 | ID Service. |
| *Show periods* | 0 | show_periods | Возможные значения: от 1 до 100.<br><br>По умолчанию: 20. |
| *From* | 1 | date_from | Допустимая строка даты в формате `YYYY-MM-DD`.<br>Поддерживаются относительные даты с модификаторами `d`, `w`, `M`, `y` (например, `now`, `now/d`, `now/w-1w` и т. д.). |
| *To* | 1 | date_to | Допустимая строка даты в формате `YYYY-MM-DD`.<br>Поддерживаются относительные даты с модификаторами `d`, `w`, `M`, `y` (например, `now`, `now/d`, `now/w-1w` и т. д.). |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *SLA report*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *SLA report*

Настройте виджет *SLA report*, который отображает отчет SLA для SLA "4" и сервиса "2" за период последних 30 дней.

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

26 Системная информация

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *System Information* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *System information*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *System Information*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - *(по умолчанию)* 15 минут. |
| *Show* | 0 | info_type | 0 - *(по умолчанию)* Статистика системы;<br>1 - Узлы сети высокой доступности. |
| *Show software update check details* | 0 | show_software_update_check_details | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Поведение параметра:<br>- *supported* если `AllowSoftwareUpdateCheck` в конфигурации сервера Zabbix включен и для *Show* установлено значение "System stats" |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *System information*. Дополнительные сведения о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *System information*

Настройте виджет *System information*, который отображает системную статистику с интервалом обновления 10 минут и включенной проверкой обновлений программного обеспечения.

Request:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Top hosts* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Top hosts*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Top hosts*.

| Параметр | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |

| Параметр | type | name | value |
|---|---|---|---|
| *Host tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |
| *Show hosts in maintenance* | 0 | maintenance | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Columns* (see below) | | | |
| *Order by* | 0 | column | Числовое значение столбца из настроенных столбцов. |
| *Order* | 0 | order | 2 - *(по умолчанию)* Top N;<br>3 - Bottom N. |

| Параметр | type | name | value |
|---|---|---|---|
| *Host limit* | 0 | show_lines | Допустимые значения: от 1 до 1000.<br><br>По умолчанию: 10.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели. |

Столбцы

Столбцы имеют общие параметры и дополнительные параметры в зависимости от настройки параметра *Data*.

**Note:**
Для всех параметров, связанных со столбцами, число в имени свойства (например, columns.0.name) указывает на столбец, для которого настроен параметр.

Следующие параметры поддерживаются для всех столбцов.

| Parameter | type | name | value |
|---|---|---|---|
| *Имя* | 1 | columns.0.name | Любое строковое значение.<br><br>Поведение параметра:<br>- *обязательный* |
| *Данные* | 0 | columns.0.data | 1 - значение элемента данных;<br>2 - имя узла сети;<br>3 - текст.<br><br>Поведение параметра:<br>- *обязательный* |
| *Основной цвет* | 1 | columns.0.base_color | Шестнадцатеричный код цвета (например, FF0000).<br><br>Поведение параметра:<br>- *обязательный* |

Значение элемента данных

Следующие параметры поддерживаются, если для *Data* задано значение ”Item value”.

**Note:**
Первое число в имени свойства *Thresholds* (например, columnsthresholds.0.color.0) указывает на столбец, для которого настроены пороговые значения, а второе число указывает позицию порогового значения в списке, отсортированном по возрастанию. Однако если пороговые значения настроены в другом порядке, после обновления конфигурации виджета в веб-интерфейсе Zabbix значения будут отсортированы по возрастанию (например, `"threshold.0.threshold":"5"` → `"threshold.0.threshold":"1"`; `"threshold.1.threshold":"1"` → `"threshold.1.threshold": "5"`).

| Parameter | type | name | value |
|---|---|---|---|
| *Item* | 1 | columns.0.item | Допустимое имя элемента данных.<br><br>При настройке виджета на шаблонной панели следует указывать только элементы данных, настроенные на шаблоне. |
| *Display item value as* | 0 | columns.0.display_value_as | 0 - *(по умолчанию)* Числовое значение;<br>1 - Текст;<br>2 - Двоичное. |

| Parameter | type | name | value |
|---|---|---|---|
| *Display* | 0 | columns.0.display | 1 - *(по умолчанию)* Как есть;<br>2 - Полоса;<br>3 - Индикаторы;<br>6 - Sparkline.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” |
| *Min* | 1 | columns.0.min | Любое числовое значение.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Bar” or ”Indicators” |
| *Max* | 1 | columns.0.max | Любое числовое значение.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Bar” or ”Indicators” |
| *Decimal places* | 0 | columns.0.decimal_places | Допустимый диапазон значений: 0-10.<br><br>По умолчанию: 2.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” |
| *Sparkline* | | | |
| *Width* | 0 | columns.0.sparkline.width | Допустимый диапазон значений: 0-10.<br><br>По умолчанию: 1.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline” |
| *Fill* | 0 | columns.0.sparkline.fill | Допустимый диапазон значений: 0-10.<br><br>По умолчанию: 3.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline” |
| *Color* | 1 | columns.0.sparkline.color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>По умолчанию: `42A5F5`.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline” |
| *Time period* | 1 | columns.0.sparkline.time_period | `DASHBOARD._timeperiod` - задать селектор *Time period* в качестве источника данных;<br>`ABCDE._timeperiod` - задать совместимый виджет (с параметром `reference`, равным `ABCDE`) в качестве источника данных.<br><br>По умолчанию: `DASHBOARD._timeperiod`<br><br>Кроме того, можно задать период времени только в параметрах *From* и *To*.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline” |

| Parameter | type | name | value |
|---|---|---|---|
| *From* | 1 | columns.0.sparkline.time_period.from | Дата и время в строка времени в абсолютном (`YYYY-MM-DD hh:mm:ss`) или относительном формате времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline”<br>- *required* if `columns.0.sparkline.time_period.to` is set |
| *To* | 1 | columns.0.sparkline.time_period.to | Дата и время в строка времени в абсолютном (`YYYY-MM-DD hh:mm:ss`) или относительном формате времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline”<br>- *required* if `columns.0.sparkline.time_period.from` is set |
| *History data* | 0 | columns.0.sparkline.history | 0 - *(по умолчанию)* Авто;<br>1 - История;<br>2 - Тренды.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” and *Display* is set to ”Sparkline” |
| *Thresholds* | | | |
| *Color* | 1 | columnsthresholds.0.color | Цвет в шестнадцатеричный код цвета (например, `FF0000`).<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” |
| *Threshold* | 1 | columnsthresholds.0.threshold | Пороговое значение строка.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” |
| *Highlights* | | | |
| *Color* | 1 | columns.0.highlights.0.color | Цвет в шестнадцатеричный код цвета (например, `FF0000`).<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Text” |
| *Pattern* | 1 | columns.0.highlights.0.pattern | Шаблон строка.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Text” |
| *Show thumb-nail* | 0 | columns.0.show_thumbnail | 0 - *(по умолчанию)* Отключено;<br>1 - Включено.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Binary” |
| *Aggregation func-tion* | 0 | columns.0.aggregate_function | 0 - *(по умолчанию)* не используется;<br>1 - min;<br>2 - max;<br>3 - avg;<br>4 - count;<br>5 - sum;<br>6 - first;<br>7 - last. |

| Parameter | type | name | value |
|---|---|---|---|
| *Time period* | 1 | columns.0.time_period.name | DASHBOARD._timeperiod - задать селектор *Time period* в качестве источника данных;<br>ABCDE._timeperiod - задать совместимый виджет (с параметром reference, равным ABCDE) в качестве источника данных.<br><br>По умолчанию: DASHBOARD._timeperiod<br><br>Кроме того, можно задать период времени только в параметрах *From* и *To*.<br><br>Parameter behavior:<br>- *supported* if *Aggregation function* is set to ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” |
| *From* | 1 | columns.0.time_period.from | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Parameter behavior:<br>- *supported* if *Time period* is not set and *Aggregation function* is set to ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last”<br>- *required* if columns.0.time_period.to is set |
| *To* | 1 | columns.0.time_period.to | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Parameter behavior:<br>- *supported* if *Time period* is not set and *Aggregation function* is set to ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last”<br>- *required* if columns.0.time_period.from is set |
| *History data* | 0 | columns.0.history | 0 - *(по умолчанию)* Авто;<br>1 - История;<br>2 - Тренды.<br><br>Parameter behavior:<br>- *supported* if *Display item value as* is set to ”Numeric” |
| *Reference* | 1 | reference | Любая строка из 5 символов (например, ABCDE или JBPNL). Это значение должно быть уникальным на панели, к которой принадлежит виджет.<br><br>Parameter behavior:<br>- *required* |

Текст

Следующие параметры поддерживаются, если для параметра *Data* установлено значение ”Текст”.

| Параметр | тип | имя | значение |
|---|---|---|---|
| *Текст* | 1 | columns.0.text | Любое строковое значение, включая макросы.<br>Поддерживаемые макросы: {HOST.*}, {INVENTORY.*}.<br><br>Поведение параметра:<br>- *обязательно*, если для *Data* установлено значение ”Текст” |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета информационной панели для виджета *Топ узлов сети*. Для получения дополнительной информации о настройке информационной панели см. dashboard.create.

Настройка виджета *Top hosts*

Настройте виджет *Top hosts*, который отображает узлы сети с наибольшей загрузкой CPU в группе узлов сети "4". Кроме того, настройте следующие пользовательские столбцы: "Host name", "CPU utilization in %", "1m avg", "5m avg", "15m avg", "Processes".

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

28 Лучшие элементы данных

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настроить виджет *Top items* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Top items*, см. поведение параметров, описанное в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Top items*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host tags* | | | |
| *Evaluation type* | 0 | host_tags_evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Tag name* | 1 | host_tags.0.tag | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Operator* | 0 | host_tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Tag value* | 1 | host_tags.0.value | Любое строковое значение.<br><br>Примечание: число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязательно* при настройке *Host tags*<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Layout* | 0 | layout | 0 - *(по умолчанию)* Горизонтальный;<br>1 - Вертикальный. |
| *Show problems* | 0 | show_problems | 0 - Все;<br>1 - *(по умолчанию)* Без подавленных;<br>2 - Нет. |
| *Items* (see below) | | | |

Расширенная настройка

Следующие параметры расширенной настройки поддерживаются для виджета *Top items*.

| Parameter | type | name | value |
|---|---|---|---|
| ***Порядок узлов сети*** | | | |
| *Сортировать по* | 0 | host_ordering_order_by | 0 - *(по умолчанию)* имя узла сети;<br>3 - значение элемента данных. |

| Parameter | type | name | value |
|---|---|---|---|
| *Шаблоны элементов данных* | 1 | host_ordering_item.0 | Имя или шаблон элемента данных (например, ”*: Number of processed *values per second”).<br><br>Примечание: чтобы настроить несколько шаблонов элементов данных, создайте объект поля виджета панели для каждого шаблона элемента данных, увеличивая номер в имени свойства.<br><br>При настройке виджета на шаблонной панели следует задавать только шаблоны для элементов данных, настроенных на шаблоне.<br><br>Поведение параметра:<br>- *обязательно*, если для *Сортировать по* задано значение ”Значение элемента данных” |
| *Порядок* | 0 | host_ordering_order | 2 - *(по умолчанию)* Top N;<br>3 - Bottom N. |
| *Ограничение* | 0 | host_ordering_limit | Допустимый диапазон значений: от 1 до 100.<br><br>По умолчанию: 10. |
| ***Порядок элементов данных*** | | | |
| *Сортировать по* | 0 | item_ordering_order_by | 1 - узел сети;<br>2 - имя элемента данных;<br>3 - *(по умолчанию)* значение элемента данных. |
| *Шаблоны узлов сети* | 1 | item_ordering_host.0 | Имя или шаблон узла сети.<br><br>Примечание: чтобы настроить несколько шаблонов узлов сети, создайте объект поля виджета панели для каждого шаблона узла сети, увеличивая номер в имени свойства.<br><br>Поведение параметра:<br>- *обязательно*, если для *Сортировать по* задано значение ”Узел сети” |
| *Порядок* | 0 | item_ordering_order | 2 - *(по умолчанию)* Top N;<br>3 - Bottom N. |
| *Ограничение* | 0 | item_ordering_limit | Допустимый диапазон значений: от 1 до 100.<br><br>По умолчанию: 10. |

Столбцы

**Note:**
Первое число в имени свойства (например, columns.0.items.0, columns.0.item_tags_evaltype) обозначает конкретный столбец, а второе число, если оно присутствует, обозначает настраиваемую сущность (например, шаблон элемента данных, тег).

| Parameter | type | name | value |
|---|---|---|---|
| *Шаблоны элементов данных* | 1 | columns.0.items.0 | Имя или шаблон элемента данных (например, ”*: Number of processed *values per second”).<br><br>Примечание: чтобы настроить несколько шаблонов элементов данных, создайте объект поля виджета панели для каждого шаблона элемента данных, увеличивая второе число в имени свойства.<br><br>При настройке виджета на шаблонной панели следует задавать только шаблоны для элементов данных, настроенных на шаблоне.<br><br>Поведение параметра:<br>- *обязателен* |
| *Теги элементов данных* | | | |
| *Тип оценки* | 0 | columns.0.item_tags_evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| *Имя тега* | 1 | columns.0.item_tags.0.tag | Любая строка.<br><br>Примечание: второе число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязателен* при настройке *Теги элементов данных* |
| *Оператор* | 0 | columns.0.item_tags.0.operator | 0 - Contains;<br>1 - Equals;<br>2 - Does not contain;<br>3 - Does not equal;<br>4 - Exists;<br>5 - Does not exist.<br><br>Примечание: второе число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязателен* при настройке *Теги элементов данных* |
| *Значение тега* | 1 | columns.0.item_tags.0.value | Любая строка.<br><br>Примечание: второе число в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *обязателен* при настройке *Теги элементов данных* |
| *Базовый цвет* | 1 | columns.0.base_color | Шестнадцатеричный код цвета (например, FF0000). |
| *Отображать значение как* | 0 | columns.0.display_value_as | 1 - *(по умолчанию)* Числовое;<br>2 - Текст. |
| *Отображение* | 0 | columns.0.display | 1 - *(по умолчанию)* Как есть;<br>2 - Полоса;<br>3 - Индикаторы;<br>6 - Sparkline. |
| *Мин.* | 1 | columns.0.min | Любое числовое значение. Поддерживаются суффиксы (например, ”1d”, ”2w”, ”4K”, ”8G”).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано ”Числовое”, а для *Отображение* задано ”Полоса” или ”Индикаторы” |

| Parameter | type | name | value |
|---|---|---|---|
| *Макс.* | 1 | columns.0.max | Любое числовое значение. Поддерживаются суффиксы (например, ”1d”, ”2w”, ”4K”, ”8G”).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано ”Числовое”, а для *Отображение* задано ”Полоса” или ”Индикаторы” |
| *Sparkline* | | | |
| *Ширина* | 0 | columns.0.sparkline.width | Возможные значения: от 0 до 10.<br><br>По умолчанию: 1.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline” |
| *Заливка* | 0 | columns.0.sparkline.fill | Возможные значения: от 0 до 10.<br><br>По умолчанию: 3.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline” |
| *Цвет* | 1 | columns.0.sparkline.color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>По умолчанию: `42A5F5`.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline” |
| *Период времени* | 1 | columns.0.sparkline.time_period.reference | `DASHBOARD._timeperiod` - задать *селектор периода времени* в качестве источника данных;<br>`ABCDE._timeperiod` - задать совместимый виджет (с параметром `reference`, равным `ABCDE`) в качестве источника данных.<br><br>По умолчанию: `DASHBOARD._timeperiod`<br><br>Кроме того, период времени можно задать только в параметрах *From* и *To*.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline” |
| *From* | 1 | columns.0.sparkline.time_period.from | Допустимая строка времени в абсолютном формате (`YYYY-MM-DD hh:mm:ss`) или относительном синтаксисе времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline”<br>- *обязателен*, если задан `columns.0.sparkline.time_period.to` |
| *To* | 1 | columns.0.sparkline.time_period.to | Допустимая строка времени в абсолютном формате (`YYYY-MM-DD hh:mm:ss`) или относительном синтаксисе времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline”<br>- *обязателен*, если задан `columns.0.sparkline.time_period.from` |
| *Исторические данные* | 0 | columns.0.sparkline.history | 0 - *(по умолчанию)* Auto;<br>1 - History;<br>2 - Trends.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображение* задано ”Sparkline” |

| Parameter | type | name | value |
|---|---|---|---|
| *Пороги* | | | |
| *Цвет* | 1 | columns.0.thresholds.color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано "Числовое" |
| *Порог* | 1 | columns.0.thresholds.threshold | Числовое значение. Поддерживаются суффиксы (например, "1d", "2w", "4K", "8G").<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано "Числовое" |
| *Подсветка* | | | |
| *Подсветка* | 1 | columns.0.highlights.color | Шестнадцатеричный код цвета (например, `FF0000`).<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано "Текст" |
| *Порог* | 1 | columns.0.highlights.pattern | Любое регулярное выражение.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано "Текст" |
| *Десятичные знаки* | 0 | columns.0.decimal_places | Возможные значения: от 0 до 10.<br><br>По умолчанию: 2. |
| ***Дополнительная настройка*** | | | |
| *Функция агрегации* | 0 | columns.0.aggregate_function | Если для *Отображать значение как* задано "Числовое":<br>0 - *(по умолчанию)* не используется;<br>1 - min;<br>2 - max;<br>3 - avg;<br>4 - count;<br>5 - sum;<br>6 - first;<br>7 - last.<br><br>Если для *Отображать значение как* задано "Текст":<br>0 - *(по умолчанию)* не используется;<br>4 - count;<br>6 - first;<br>7 - last. |
| *Период времени* | 1 | columns.0.time_period.name | `DASHBOARD._timeperiod` - задать *селектор периода времени* в качестве источника данных;<br>`ABCDE._timeperiod` - задать совместимый виджет (с параметром `reference`, равным `ABCDE`) в качестве источника данных.<br><br>По умолчанию: `DASHBOARD._timeperiod`<br><br>Кроме того, период времени можно задать только в параметрах *From* и *To*.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Функция агрегации* заданы значения "min", "max", "avg", "count", "sum", "first", "last" |

| Parameter | type | name | value |
|---|---|---|---|
| *From* | 1 | columns.0.time_period.from | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если *Период времени* не задан и для *Функция агрегации* заданы значения "min", "max", "avg", "count", "sum", "first", "last"<br>- *обязателен*, если задан columns.0.time_period.to |
| *To* | 1 | columns.0.time_period.to | Допустимая строка времени в абсолютном формате (YYYY-MM-DD hh:mm:ss) или относительном синтаксисе времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *поддерживается*, если *Период времени* не задан и для *Функция агрегации* заданы значения "min", "max", "avg", "count", "sum", "first", "last"<br>- *обязателен*, если задан columns.0.time_period.from |
| *Исторические данные* | 0 | columns.0.history | 0 - *(по умолчанию)* Auto;<br>1 - History;<br>2 - Trends.<br><br>Поведение параметра:<br>- *поддерживается*, если для *Отображать значение как* задано "Числовое" |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Top items*. Дополнительные сведения о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *Top items*

Настройте виджет *Top items*, который отображает данные для узла сети "10084" и только для тех элементов данных, у которых тег с именем "component" содержит значение "cpu". Кроме того, отображайте данные так, чтобы узлы сети были вверху, и используйте цветную полосу gauge для представления ячеек.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

29 Топ триггеров

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Top triggers* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Top triggers*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Top triggers*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - *(default)* Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem* | 1 | problem | Имя события проблемы (без учета регистра, полное имя или его часть). |

| Parameter | type | name | value |
|---|---|---|---|
| *Severity* | 0 | severities.0 | Серьезность триггера.<br><br>0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Чрезвычайная.<br><br>По умолчанию: пусто (все включены).<br><br>Примечание: чтобы настроить несколько значений, создайте объект поля виджета панели для каждого значения, увеличивая номер в имени свойства. |
| *Problem tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(default)* И/ИЛИ;<br>2 - ИЛИ. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Problem tags* |
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Problem tags* |
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Problem tags* |
| *Time period* | 1 | time_period._referenceDASHBOARD._timeperiod - задает селектор *Time period* в качестве источника данных;<br>ABCDE._timeperiod - задает совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных.<br><br>По умолчанию: DASHBOARD._timeperiod<br><br>Кроме того, вы можете задать период времени только в параметрах *From* и *To*. | |
| *From* | 1 | time_period.from | Допустимая строка времени в абсолютном (YYYY-MM-DD hh:mm:ss) или относительном формате времени (now, now/d, now/w-1w и т. д.).<br><br>Поведение параметра:<br>- *supported* если *Time period* не задан<br>- *required* если задан time_period.to |

| Parameter | type | name | value |
|---|---|---|---|
| *To* | 1 | time_period.to | Допустимая строка времени в абсолютном (`YYYY-MM-DD hh:mm:ss`) или относительном формате времени (`now`, `now/d`, `now/w-1w` и т. д.).<br><br>Поведение параметра:<br>- *supported* если *Time period* не задан<br>- *required* если задан `time_period.from` |
| *Trigger limit* | 0 | show_lines | Возможные значения: от 1 до 100.<br><br>По умолчанию: 10. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Top triggers*. Дополнительные сведения о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Top triggers*

Настройте виджет *Top triggers*, который отображает 5 наиболее значимых триггеров для группы узлов сети "4" с количеством всех проблем для каждого триггера. Виджет отображает только триггеры с уровнями серьезности "Warning", "Average", "High" или "Disaster", а также проблемы, у которых есть тег с именем "scope" и значением "performance" или "availability", или "capacity".

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

30 Обзор триггеров

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели мониторинга позволяют настраивать виджет *Trigger overview* в методах `dashboard.create` и `dashboard.update`.

**Attention:**
Свойства `fields` виджета не проверяются при создании или обновлении панели мониторинга. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Trigger overview*, см. поведение параметров, описанное в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Trigger Overview*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Не обновлять;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Show* | 0 | show | 1 - *(по умолчанию)* Recent problems;<br>2 - Any;<br>3 - Problems. |

| Parameter | type | name | value |
|---|---|---|---|
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети, увеличивая номер в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети, увеличивая номер в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Problem tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настроен *Problem tags* |
| *Operator* | 0 | tags.0.operator | 0 - Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настроен *Problem tags* |

| Parameter | type | name | value |
|---|---|---|---|
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настроен *Problem tags* |
| *Show suppressed problems* | 0 | show_suppressed | 0 - *(по умолчанию)* Отключено;<br>1 - Включено. |
| *Layout* | 0 | layout | 0 - *(по умолчанию)* Горизонтально;<br>1 - Вертикально. |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Trigger overview*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Обзор триггеров*

Настройте виджет *Обзор триггеров*, который отображает состояния триггеров для всех групп узлов сети, у которых есть триггеры с тегом, имя которого равно "scope" и значение содержит "availability".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

31 URL

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *URL* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *URL*, обратитесь к поведению параметров, описанному в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *URL*.

| Parameter | type | name | value |
|---|---|---|---|
| *Интервал обновления* | 0 | rf_rate | 0 - *(по умолчанию)* Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |

| Parameter | type | name | value |
|---|---|---|---|
| *URL* | 1 | url | Допустимая строка URL.<br><br>Поведение параметра:<br>- *обязательно* |
| *Переопределить узел сети* | 1 | override_hostid._reference | ABCDE._hostid - задайте совместимый виджет (с параметром *Reference*, установленным в "ABCDE") в качестве источника данных для узлов сети;<br>DASHBOARD._hostid - задайте *Host* selector на панели управления в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблоне панели управления. |

Примеры

Следующие примеры предназначены только для описания настройки объектов полей виджета панели мониторинга для виджета *URL*. Дополнительную информацию о настройке панели мониторинга см. в dashboard.create.

Настройка виджета *URL*

Настройте виджет *URL*, который отображает главную страницу руководства Zabbix.

Запрос:

```
{
    "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/7.4/en"
                            }
                        ]
                    }
                ]
            }
        ],
        "userGroups": [
            {
                "usrgrpid": 7,
                "permission": 2
            }
        ],
        "users": [
            {
                "userid": 1,
                "permission": 3
```

```
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- dashboard.create
- dashboard.update

32 Веб-мониторинг

Описание

Эти параметры и возможные значения свойств для соответствующих объектов полей виджета панели позволяют настраивать виджет *Web monitoring* в методах dashboard.create и dashboard.update.

**Attention:**
Свойства fields виджета не проверяются при создании или обновлении панели. Это позволяет пользователям изменять встроенные виджеты и создавать пользовательские виджеты, но также создает риск некорректного создания или обновления виджетов. Чтобы обеспечить успешное создание или обновление виджета *Web monitoring*, ознакомьтесь с поведением параметров, описанным в таблицах ниже.

Параметры

Следующие параметры поддерживаются для виджета *Web monitoring*.

| Parameter | type | name | value |
|---|---|---|---|
| *Refresh interval* | 0 | rf_rate | 0 - Без обновления;<br>10 - 10 секунд;<br>30 - 30 секунд;<br>60 - *(по умолчанию)* 1 минута;<br>120 - 2 минуты;<br>600 - 10 минут;<br>900 - 15 минут. |
| *Host groups* | 2 | groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы настроить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети с увеличенным номером в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |

| Parameter | type | name | value |
|---|---|---|---|
| *Host groups (Widget)* | 1 | groupids._reference | Вместо ID группы узлов сети:<br>`ABCDE._hostgroupids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для групп узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Exclude host groups* | 2 | exclude_groupids.0 | ID группы узлов сети.<br><br>Примечание: чтобы исключить несколько групп узлов сети, создайте объект поля виджета панели для каждой группы узлов сети с увеличенным номером в имени свойства.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts* | 3 | hostids.0 | ID узла сети.<br><br>Примечание: чтобы настроить несколько узлов сети, создайте объект поля виджета панели для каждого узла сети с увеличенным номером в имени свойства. Для нескольких узлов сети параметр *Host groups* должен либо вообще не быть настроен, либо быть настроен как минимум с одной группой узлов сети, к которой принадлежат настроенные узлы сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Hosts (Widget/Dashboard)* | 1 | hostids._reference | Вместо ID узла сети:<br>`DASHBOARD.hostids` - задайте селектор *Host* в качестве источника данных для узлов сети;<br>`ABCDE._hostids` - задайте совместимый виджет (с параметром *Reference*, установленным в "`ABCDE`") в качестве источника данных для узлов сети.<br><br>Этот параметр не поддерживается при настройке виджета на шаблонной панели. |
| *Scenario tags* | | | |
| *Evaluation type* | 0 | evaltype | 0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| *Tag name* | 1 | tags.0.tag | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Scenario tags* |
| *Operator* | 0 | tags.0.operator | 0 - Contains;<br>1 - Equals;<br>2 - Does not contain;<br>3 - Does not equal;<br>4 - Exists;<br>5 - Does not exist.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Scenario tags* |

| Parameter | type | name | value |
|---|---|---|---|
| *Tag value* | 1 | tags.0.value | Любое строковое значение.<br><br>Примечание: номер в имени свойства указывает порядок тега в списке оценки тегов.<br><br>Поведение параметра:<br>- *required* если настраивается *Scenario tags* |
| *Show hosts in maintenance* | 0 | maintenance | 0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| *Reference* | 1 | reference | Любое строковое значение, состоящее из 5 символов (например, `ABCDE` или `JBPNL`). Это значение должно быть уникальным в пределах панели, к которой принадлежит виджет.<br><br>Поведение параметра:<br>- *required* |

Примеры

Следующие примеры предназначены только для описания конфигурации объектов полей виджета панели мониторинга для виджета *Web monitoring*. Дополнительную информацию о настройке панели мониторинга см. в `dashboard.create`.

Настройка виджета *Web monitoring*

Настройте виджет *Web monitoring*, который отображает сводку состояния активных сценариев веб-мониторинга для группы узлов сети "4".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Смотрите также

- Поле виджета панели
- `dashboard.create`
- `dashboard.update`

#### Пользователь

Этот класс предназначен для работы с пользователями.

Ссылки на объекты:

- User
- Media

Доступные методы:

- user.checkauthentication - проверка и продление пользовательских сеансов
- user.create - создание новых пользователей
- user.delete - удаление пользователей
- user.get - получение пользователей
- user.login - вход в API
- user.logout - выход из API
- user.provision - подготовка пользователей LDAP
- user.resettotp - сброс секретов TOTP пользователей
- user.unblock - разблокировка пользователей
- user.update - обновление пользователей

#### Объект user

Следующие объекты напрямую связаны с API `user`.

Пользователь

Объект пользователя имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| userid | ID | ID пользователя.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| username | string | Имя пользователя.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для подготовленных пользователей, если пользователь связан с каталогом пользователей (`userdirectoryid` имеет допустимое значение, отличное от ”0”), включено состояние подготовки каталога пользователей (`provision_status` объекта User directory object установлено в ”1”), и включено состояние аутентификации для всей подготовки LDAP или SAML (`ldap_jit_status` объекта Authentication object установлено в ”Enabled for configured LDAP IdPs” или `saml_jit_status` объекта Authentication object установлено в ”Enabled for configured SAML IdPs”) |
| passwd | string | Пароль пользователя.<br><br>Значение этого параметра может быть пустой строкой, если пользователь связан с каталогом пользователей.<br><br>Поведение свойства:<br>- *только для записи* |
| roleid | ID | ID роли пользователя.<br><br>Обратите внимание, что пользователи без роли могут входить в Zabbix только с использованием аутентификации LDAP или SAML, если их данные LDAP/SAML соответствуют сопоставлениям групп пользователей, настроенным в Zabbix. |
| attempt_clock | timestamp | Время последней неудачной попытки входа.<br><br>Поведение свойства:<br>- *только для чтения* |
| attempt_failed | integer | Количество недавних неудачных попыток входа.<br><br>Поведение свойства:<br>- *только для чтения* |

| Property | Type | Description |
|---|---|---|
| attempt_ip | string | IP-адрес, с которого была выполнена последняя неудачная попытка входа.<br><br>Поведение свойства:<br>- *только для чтения* |
| autologin | integer | Включать ли автоматический вход.<br><br>Возможные значения:<br>0 - *(по умолчанию)* автоматический вход отключен;<br>1 - автоматический вход включен. |
| autologout | string | Время жизни сеанса пользователя. Принимает секунды и единицу времени с суффиксом. Если установлено значение 0s, сеанс никогда не истечет.<br><br>По умолчанию: 15m. |
| lang | string | Код языка пользователя, например `en_US`.<br><br>По умолчанию: `default` - системный язык по умолчанию. |
| name | string | Имя пользователя. |
| provisioned | integer | Был ли пользователь подготовлен.<br><br>Возможные значения:<br>0 - не подготовлен;<br>1 - подготовлен.<br><br>Поведение свойства:<br>- *только для чтения* |
| refresh | string | Период автоматического обновления. Принимает секунды или единицу времени с суффиксом (например, 30s, 90s, 1m, 1h).<br><br>По умолчанию: 30s. |
| rows_per_page | integer | Количество строк объекта, отображаемых на странице.<br><br>По умолчанию: 50. |
| surname | string | Фамилия пользователя. |
| theme | string | Тема пользователя.<br><br>Возможные значения:<br>`default` - *(по умолчанию)* системная тема по умолчанию;<br>`blue-theme` - Blue;<br>`dark-theme` - Dark;<br>`hc-light` - High-contrast light;<br>`hc-dark` - High-contrast dark. |
| ts_provisioned | timestamp | Время, когда была выполнена последняя операция подготовки.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *поддерживается* для пользователей типа *Super admin* |

| Property | Type | Description |
|---|---|---|
| url | string | URL страницы, на которую следует перенаправить пользователя после входа в систему. |
| userdirectoryid | ID | ID каталога пользователей, к которому привязан пользователь.<br><br>Используется для подготовки (создания или обновления), а также для входа пользователя, связанного с каталогом пользователей.<br><br>Для операций входа значение этого свойства будет иметь приоритет над свойством `userdirectoryid` групп пользователей, к которым принадлежит пользователь.<br><br>По умолчанию: 0.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *поддерживается* для пользователей типа *Super admin* |
| timezone | string | Часовой пояс пользователя, например `Europe/London`, `UTC`.<br><br>По умолчанию: `default` - системный часовой пояс по умолчанию.<br><br>Полный список поддерживаемых часовых поясов см. в документации PHP. |

Media

Объект media имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| mediaid | ID | ID media пользователя.<br><br>Поведение свойства:<br>- *только для чтения* |
| mediatypeid | ID | ID типа media, используемого media пользователя.<br><br>Поведение свойства:<br>- *обязательно* |
| sendto | string/array | Адрес, имя пользователя или другой идентификатор получателя.<br><br>Если `type` типа media установлен в ”Email”, значения представляются как массив. Для других типов media значение представляется как строка.<br><br>Поведение свойства:<br>- *обязательно* |
| active | integer | Включено ли media.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включено;<br>1 - отключено. |

| Property | Type | Description |
|---|---|---|
| severity | integer | Серьезности триггеров, о которых отправлять уведомления.<br><br>Возможные битовые значения:<br>1 - Не классифицировано;<br>2 - Информация;<br>4 - Предупреждение;<br>8 - Средняя;<br>16 - Высокая;<br>32 - Чрезвычайная.<br><br>Это поле является битовой маской; допустима любая сумма возможных битовых значений (например, 48 для Average, High и Disaster).<br><br>По умолчанию: 63. |
| period | string | Время, когда уведомления могут быть отправлены, в виде временного периода или пользовательских макросов, разделенных точкой с запятой.<br><br>По умолчанию: 1-7,00:00-24:00. |
| provisioned | integer | Был ли пользователь подготовлен.<br><br>Возможные значения:<br>0 - не подготовлен;<br>1 - подготовлен. |
| userdirectory_mediaid | ID | ID сопоставления media каталога пользователей для подготовленного media.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *поддерживается* для пользователей типа *Super admin* |

**user.checkAuthentication**

Описание

`object user.checkAuthentication`

Этот метод проверяет и продлевает сеанс пользователя.

> **Attention:**
> Вызов метода `user.checkAuthentication` с использованием параметра `sessionid` по умолчанию продлевает сеанс пользователя.

Параметры

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| extend | boolean | Следует ли продлевать сеанс пользователя.<br><br>Значение по умолчанию: ”true”.<br>Установка значения ”false” позволяет проверить сеанс пользователя без его продления. |
| sessionid | string | Поведение параметра:<br>- *supported* если `sessionid` задан<br>Токен аутентификации пользователя.<br><br>Поведение параметра:<br>- *required* если `token` не задан |

| Parameter | Type | Description |
|---|---|---|
| token | string | API-токен пользователя.<br><br>Поведение параметра:<br>- *required* если `sessionid` не задан |

Возвращаемые значения

`(object)` Возвращает объект, содержащий информацию о пользователе.

В дополнение к стандартным свойствам пользователя возвращается следующая информация.

| Property | Type | Description |
|---|---|---|
| auth_type | integer | Метод аутентификации по умолчанию для пользователя.<br><br>См. свойство `authentication_type` объекта Authentication object для списка возможных значений. |
| debug_mode | integer | Включен или отключен режим отладки для пользователя.<br><br>См. свойство `debug_mode` объекта User group object для списка возможных значений. |
| deprovisioned | boolean | Принадлежит ли пользователь группе deprovisioned users group. |
| gui_access | string | Метод аутентификации пользователя во веб-интерфейс.<br><br>См. свойство `gui_access` объекта User group object для списка возможных значений. |
| secret | string | Случайная строка из 32 символов. Генерируется при входе пользователя в систему.<br><br>Свойство `secret` не возвращается, если сеанс пользователя проверяется с использованием API-токена. |
| sessionid | string | Токен аутентификации, который необходимо использовать в следующих API-запросах.<br><br>Свойство `sessionid` не возвращается, если сеанс пользователя проверяется с использованием API-токена. |
| type | integer | Тип пользователя.<br><br>См. свойство `type` объекта Role object для списка возможных значений. |
| userip | string | IP-адрес пользователя. |

Примеры

Проверка аутентификации с использованием токена аутентификации

Проверьте и продлите сеанс пользователя с помощью токена аутентификации пользователя, а также верните дополнительную информацию о пользователе.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Проверка аутентификации с использованием API-токена

Проверяет сеанс пользователя с помощью API-токена пользователя и возвращает дополнительную информацию о пользователе.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "token": "00aff470e07c12d707e50d98cfe39edef9e6ec349c14728dbdfbc8ddc5ea3eae"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Источник

CUser::checkAuthentication() в *ui/include/classes/api/services/CUser.php*.

**user.create**

Описание

`object user.create(  /  users)`

Этот метод позволяет создавать новых пользователей.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Пользователи для создания.

В дополнение к стандартным свойствам пользователя, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| usrgrps | array | Группы пользователей, в которые нужно добавить пользователя.<br><br>Группы пользователей должны содержать только свойство `usrgrpid`. |
| medias | array | Медиа пользователя, которые нужно создать. |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID созданных пользователей, указанных в свойстве `userids`. Порядок возвращаемых ID совпадает с порядком переданных пользователей.

Примеры

Создание пользователя

Создайте нового пользователя, добавьте его в группу пользователей и создайте для него новый способ оповещения.

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "user.create",
    "params": {
        "username": "John",
        "passwd": "Doe123",
        "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

См. также

- Аутентификация
- Медиа
- Группа пользователей
- Роль

Источник

CUser::create() в *ui/include/classes/api/services/CUser.php*.

**user.delete**

Описание

`object user.delete(    users)`

Этот метод позволяет удалять пользователей.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(  ) ID удаляемых пользователей.

Возвращаемые значения

(  ) Возвращает объект, который содержит ID удаленных пользователей, указанных в свойстве `userids`.

Примеры

Удаление нескольких пользователей

Удалите двух пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.delete",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CUser::delete() в *ui/include/classes/api/services/CUser.php*.

**user.get**

Описание

`integer/array user.get(          )`

Метод позволяет получать пользователей по заданным параметрам.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

**Note:**
При запросе информации о пользовательских медиа, правах доступа или ролях пользователи типов *Admin* и *User* могут получать данные только о своем пользователе. Пример см. в разделе Получение пользователей как *Admin*.

| Parameter | Type | Description |
|---|---|---|
| mediaids | ID/array | Возвращать только пользователей, которые используют указанные медиа. |
| mediatypeids | ID/array | Возвращать только пользователей, которые используют указанные типы медиа. |
| userids | ID/array | Возвращать только пользователей с указанными ID. |
| usrgrpids | ID/array | Возвращать только пользователей, которые принадлежат к указанным группам пользователей. |

| Parameter | Type | Description |
|---|---|---|
| getAccess | flag | Добавляет дополнительную информацию о правах доступа пользователя.<br><br>Добавляет следующие свойства для каждого пользователя:<br>`gui_access` - `(integer)` метод аутентификации пользователя во веб-интерфейсе. Список возможных значений см. в свойстве `gui_access` объекта группы пользователей.<br>`debug_mode` - `(integer)` указывает, включен ли для пользователя режим отладки. Возможные значения: 0 - отладка отключена, 1 - отладка включена.<br>`users_status` - `(integer)` указывает, отключен ли пользователь. Возможные значения: 0 - пользователь включен, 1 - пользователь отключен. |
| selectMedias | query | Возвращать медиа, используемые пользователем, в свойстве `medias`. |
| selectMediatypes | query | Возвращать типы медиа, используемые пользователем, в свойстве `mediatypes`.<br><br>См. `mediatype.get` для ограничений в зависимости от типа пользователя. |
| selectUsrgrps | query | Возвращать группы пользователей, к которым принадлежит пользователь, в свойстве `usrgrps`.<br><br>См. `usergroup.get` для ограничений в зависимости от типа пользователя. |
| selectRole | query | Возвращать роль пользователя в свойстве `role`. |
| filter | object | Возвращать только те результаты, которые в точности соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Возможные свойства объекта User для пользователей типов *Admin* и *User* при запросе данных о пользователях в их группе пользователей: `userid`, `name`, `surname`, `username`. |
| output | query | Свойства объекта User, которые нужно вернуть.<br><br>Пользователи типов *Admin* и *User* могут получать только следующие свойства:<br>- Для своего пользователя: `userid`, `attempt_clock`, `attempt_failed`, `attempt_ip`, `autologin`, `autologout`, `lang`, `name`, `provisioned`, `refresh`, `roleid`, `rows_per_page`, `surname`, `theme`, `timezone`, `url`, `username`.<br>- Для пользователей в своей группе пользователей: `userid`, `name`, `surname`, `username`.<br><br>По умолчанию: `extend`. |
| search | object | Возвращать результаты, соответствующие заданному шаблону (без учета регистра).<br><br>Принимает объект, где ключи — это имена свойств, а значения — строки для поиска. Если дополнительные параметры не заданы, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживает только свойства с типами данных `string` и `text`.<br><br>Возможные свойства объекта User для пользователей типов *Admin* и *User* при запросе данных о пользователях в их группе пользователей: `name`, `surname`, `username`. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `userid`, `username`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| preservekeys | boolean | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение пользователей

Получить всех настроенных пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "userid": "1",
            "username": "Admin",
            "name": "Zabbix",
            "surname": "Administrator",
            "url": "",
            "autologin": "1",
            "autologout": "0",
            "lang": "en_US",
            "refresh": "0s",
            "theme": "default",
            "attempt_failed": "0",
            "attempt_ip": "",
            "attempt_clock": "0",
            "rows_per_page": "50",
            "timezone": "default",
            "roleid": "3",
            "userdirectoryid": "0",
            "ts_provisioned": "0"
        },
        {
            "userid": "2",
            "username": "guest",
```

```
            "name": "",
            "surname": "",
            "url": "",
            "autologin": "0",
            "autologout": "15m",
            "lang": "default",
            "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"
        },
        {
            "userid": "3",
            "username": "user",
            "name": "Zabbix",
            "surname": "User",
            "url": "",
            "autologin": "0",
            "autologout": "0",
            "lang": "ru_RU",
            "refresh": "15s",
            "theme": "dark-theme",
            "attempt_failed": "0",
            "attempt_ip": "",
            "attempt_clock": "0",
            "rows_per_page": "100",
            "timezone": "default",
            "roleid": "1",
            "userdirectoryid": "0",
            "ts_provisioned": "0"
        }
    ],
    "id": 1
}
```

Получение пользователей как *Admin*

Как пользователь типа *Admin*, получите подробные данные о собственном пользователе и ограниченные данные о пользователях в вашей группе пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend",
        "getAccess": true,
        "selectMedias": "extend",
        "selectMediatypes": "extend",
        "selectUsrgrps": "extend",
        "selectRole": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "userid": "1",
            "username": "Admin",
            "name": "Zabbix",
            "surname": "Administrator",
            "usrgrps": [
                {
                    "usrgrpid": "7",
                    "name": "Zabbix administrators",
                    "gui_access": "0",
                    "users_status": "0",
                    "debug_mode": "0",
                    "mfa_status": "0"
                }
            ]
        },
        {
            "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",
            "gui_access": "0",
            "debug_mode": "0",
            "users_status": "0",
            "usrgrps": [
                {
                    "usrgrpid": "7",
                    "name": "Zabbix administrators",
                    "gui_access": "0",
                    "users_status": "0",
                    "debug_mode": "0",
                    "mfa_status": "0"
                }
            ],
            "medias": [
                {
                    "mediaid": "2",
                    "mediatypeid": "1",
                    "sendto": [
                        "john.doe@example.com"
                    ],
                    "active": "0",
                    "severity": "63",
                    "period": "1-7,00:00-24:00",
                    "provisioned": 0
                }
```

```
            ],
            "mediatypes": [
                {
                    "mediatypeid": "1",
                    "type": "0",
                    "name": "Email",
                    "status": "0",
                    "description": "",
                    "maxattempts": "3"
                }
            ],
            "role": {
                "roleid": "2",
                "name": "Admin role",
                "type": "2",
                "readonly": "0"
            }
        }
    ],
    "id": 1
}
```

Получение данных пользователя

Получение данных пользователя с ID ”12”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": ["userid", "username"],
        "selectRole": "extend",
        "userids": "12"
    },
    "id": 1
}
```

Ответ:

```
{
  "jsonrpc": "2.0",
  "result": [
      {
          "userid": "12",
          "username": "John",
          "role": {
              "roleid": "5",
              "name": "Operator",
              "type": "1",
              "readonly": "0"
          }
      }
  ],
  "id": 1
}
```

См. также

- Медиа
- Тип медиа
- Группа пользователей
- Роль

Источник

CUser::get() в *ui/include/classes/api/services/CUser.php*.

**user.login**

Описание

`string/object user.login(object parameters)`

Этот метод позволяет выполнить вход в API и сгенерировать токен аутентификации.

**Warning:**
При использовании этого метода также необходимо выполнить user.logout, чтобы предотвратить создание большого количества открытых записей сеансов.

**Attention:**
Этот метод доступен только неаутентифицированным пользователям, которые не принадлежат ни к одной группе пользователей с включенной многофакторной аутентификацией.

Параметры

`(object)` Параметры, содержащие имя пользователя и пароль.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| password | string | Пароль пользователя.<br><br>Поведение параметра:<br>- *required* |
| username | string | Имя пользователя.<br><br>Поведение параметра:<br>- *required* |
| userData | flag | Возвращает информацию об аутентифицированном пользователе. |

Возвращаемые значения

`(string/object)` Если используется параметр `userData`, возвращает объект, содержащий информацию об аутентифицированном пользователе.

В дополнение к стандартным свойствам пользователя возвращается следующая информация:

| Property | Type | Description |
|---|---|---|
| auth_type | integer | Аутентификация по умолчанию для пользователя.<br><br>См. свойство `authentication_type` объекта Authentication object для списка возможных значений. |
| debug_mode | integer | Включен или отключен режим отладки для пользователя.<br><br>См. свойство `debug_mode` объекта User group object для списка возможных значений. |
| deprovisioned | boolean | Принадлежит ли пользователь группе deprovisioned users group. |
| gui_access | string | Метод аутентификации пользователя во веб-интерфейс.<br><br>См. свойство `gui_access` объекта User group object для списка возможных значений. |
| mfaid | integer | ID MFA method, который будет использоваться для пользователя при входе в систему.<br><br>Возвращает "0", если MFA глобально отключена или отключена для всех групп пользователей, к которым принадлежит пользователь. |

| Property | Type | Description |
|---|---|---|
| secret | string | Случайная строка из 32 символов. Генерируется при входе пользователя в систему. |
| sessionid | string | Токен аутентификации, который необходимо использовать в последующих API-запросах. |
| type | integer | Тип пользователя.<br><br>См. свойство type объекта Role object для списка возможных значений. |
| userip | string | IP-адрес пользователя. |

**Note:**
Если пользователь был успешно аутентифицирован после одной или нескольких неудачных попыток, метод вернет текущие значения свойств `attempt_clock`, `attempt_failed` и `attempt_ip`, а затем сбросит их.

Если параметр `userData` не используется, метод возвращает токен аутентификации, необходимый для аутентификации.

Примеры

Аутентификация пользователя

Аутентифицируйте пользователя.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

Запрос информации об аутентифицированном пользователе

Выполните аутентификацию и верните дополнительную информацию о пользователе.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix",
        "userData": true
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Смотрите также

- user.logout

Источник

CUser::login() в *ui/include/classes/api/services/CUser.php*.

**user.logout**

Описание

`string/object user.logout(    )`

Этот метод позволяет выйти из API и аннулирует текущий токен аутентификации.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) Метод принимает пустой массив.

Возвращаемые значения

(       ) Возвращает `true`, если пользователь успешно выполнил выход.

Примеры

Выход из системы

Выйти из API.

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "user.logout",
    "params": [],
```

```
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Смотрите также

- user.login

Источник

CUser::login() в *ui/include/classes/api/services/CUser.php*.

**user.provision**

Описание

`object user.provision(    /     users)`

Этот метод позволяет предоставлять пользователей LDAP.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы пользователей для подготовки.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы подготовленных пользователей в свойстве `userids`.

Примеры

Подготовка нескольких пользователей

Подготовьте двух пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.provision",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CUser::provision() в *ui/include/classes/api/services/CUser.php*.

**user.resettotp**

Описание

`object user.resettotp(  /  users)`

Этот метод позволяет сбросить TOTP-секреты пользователя.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы пользователей, для которых нужно сбросить секреты TOTP.

> **Note:**
> Сеансы пользователей для указанных пользователей также будут удалены (за исключением пользователя, отправившего запрос).

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы пользователей, для которых были сброшены секреты TOTP, в свойстве `userids`.

Примеры

Сброс секретов TOTP для нескольких пользователей

Сбросьте секреты TOTP для двух пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.resettotp",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

См. также

- Объект MFA

Источник

CUser::resettotp() в *ui/include/classes/api/services/CUser.php*.

### user.unblock

Описание

`object user.unblock( userids)`

Этот метод позволяет разблокировать пользователей.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID пользователей для разблокировки.

Возвращаемые значения

( ) Возвращает объект, который содержит ID разблокированных пользователей, указанных в свойстве `userids`.

Примеры

Разблокировка нескольких пользователей

Разблокируйте двух пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.unblock",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CUser::unblock() в *ui/include/classes/api/services/CUser.php*.

### user.update

Описание

`object user.update( / users)`

Этот метод позволяет обновлять существующих пользователей.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

**Note:**
Сложность пароля пользователя проверяется в соответствии с правилами политики паролей, определенными API аутентификации. Подробнее см. в API аутентификации.

Параметры

(object/array) Пользовательские свойства, которые необходимо обновить.

Свойство userid должно быть определено для каждого пользователя, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Кроме стандартных пользовательских свойств, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| current_passwd | string | Текущий пароль пользователя.<br><br>Значение этого параметра может быть пустой строкой, если пользователь связан с каталогом пользователей.<br><br>Поведение параметра:<br>- *только для записи*<br>- *обязательно*, если passwd объекта User object задан и пользователь изменяет собственный пароль |
| usrgrps | array | Группы пользователей, которые заменят существующие группы пользователей.<br><br>Группы пользователей должны иметь определенным только свойство usrgrpid. |
| medias | array | Пользовательские способы оповещения, которые заменят существующие, не предоставленные системой способы оповещения. Предоставленные системой способы оповещения можно не указывать при обновлении способов оповещения. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных пользователей, указанных в свойствеuserids.

Примеры

Переименование пользователя

Переименуйте пользователя в John Doe.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "1",
        "name": "John",
        "surname": "Doe"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1"
        ]
    },
```

```
    "id": 1
}
```

Изменение роли пользователя

Измените роль пользователя.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "12",
        "roleid": "6"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Смотрите также

- Аутентификация

Источник

CUser::update() в *ui/include/classes/api/services/CUser.php*.

**Пользовательский макрос**

Этот класс предназначен для работы с пользовательскими макросами уровня узла сети и глобальными пользовательскими макросами.

Ссылки на объекты:

- Глобальный макрос
- Макрос узла сети
    - Конфигурация макроса

Доступные методы:

- usermacro.create - создать новые макросы узла сети
- usermacro.createglobal - создать новые глобальные макросы
- usermacro.delete - удалить макросы узла сети
- usermacro.deleteglobal - удалить глобальные макросы
- usermacro.get - получить макросы узла сети и глобальные макросы
- usermacro.update - обновить макросы узла сети
- usermacro.updateglobal - обновить глобальные макросы

**Объект пользовательского макроса**

Следующие объекты напрямую связаны с API `usermacro`.

Глобальный макрос

Объект глобального макроса имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| globalmacroid | ID | ID глобального макроса.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| description | string | Описание макроса. |
| macro | string | Строка макроса.<br><br>Property behavior:<br>- *required* for create operations |
| type | integer | Тип макроса.<br><br>Possible values:<br>0 - *(default)* Текстовый макрос;<br>1 - Секретный макрос;<br>2 - Секрет хранилища. |
| value | string | Значение макроса.<br><br>Property behavior:<br>- *write-only* if `type` is set to "Secret macro"<br>- *required* for create operations |

Макрос узла сети

Объект макроса узла сети определяет макрос, доступный на узле сети, прототипе узла сети или шаблоне. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| hostmacroid | ID | ID макроса.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| automatic | integer | Определяет, управляется ли макрос правилом обнаружения.<br><br>Possible values:<br>0 - *(default)* Макрос управляется пользователем;<br>1 - Макрос управляется правилом обнаружения.<br><br>User is not allowed to create automatic macro.<br>To update automatic macro, it must be converted to manual. |
| config | object/array | Конфигурация макроса, отвечающая за то, как макрос отображается в Мастере узлов сети. |
| description | string | Описание макроса. |
| hostid | ID | ID узла сети, прототипа узла сети или шаблона, которому принадлежит макрос.<br><br>Property behavior:<br>- *constant*<br>- *required* for create operations |
| macro | string | Строка макроса.<br><br>Property behavior:<br>- *required* for create operations |
| type | integer | Тип макроса.<br><br>Possible values:<br>0 - *(default)* Текстовый макрос;<br>1 - Секретный макрос;<br>2 - Секрет из хранилища. |

| Property | Type | Description |
|---|---|---|
| value | string | Значение макроса.<br><br>Property behavior:<br>- *write-only* if `type` is set to ”Secret macro”<br>- *required* for create operations |

Конфигурация макроса

Объект конфигурации макроса определяет, как макрос отображается в Мастере узла сети.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип поля ввода макроса.<br><br>Возможные значения:<br>0 - Макрос не используется в Мастере узла сети;<br>1 - Текстовое поле;<br>2 - Список;<br>3 - Флажок.<br><br>Property behavior:<br>- *required* |
| label | string | Метка для поля ввода макроса.<br><br>Property behavior:<br>- *required* если `type` установлено в ”Textbox”, ”List” или ”Checkbox” |
| description | text | Справочный текст, отображаемый рядом с полем ввода макроса.<br>Поддерживает форматирование Markdown. |
| priority | integer | Позиция поля ввода макроса в списке макросов.<br><br>Если `priority` не задан, макрос добавляется в конец списка несгруппированных макросов. |
| required | integer | Помечает макрос как обязательный.<br><br>Возможные значения:<br>0 - Необязательный;<br>1 - Обязательный.<br><br>Property behavior:<br>- *supported* если `type` установлено в ”Textbox” или ”List” |
| regex | string | Регулярное выражение для проверки ввода пользователя в поле текстового ввода.<br><br>Property behavior:<br>- *supported* если `type` установлено в ”Textbox” |
| section_name | string | Метка сворачиваемого раздела, в котором сгруппирован макрос.<br><br>Если `section_name` не задан, макрос не сгруппирован.<br>Несгруппированные макросы отображаются первыми;<br>сгруппированные макросы отображаются ниже, упорядоченные по их `priority` внутри каждого сворачиваемого раздела. |

| Property | Type | Description |
|---|---|---|
| options | text | Строка JSON, определяющая элементы списка или значения флажка.<br><br>Для списков: массив объектов со свойствами `value` и `text`.<br>Пример: `[{"value": "http", "text": "HTTP"}, {"value": "https", "text": "HTTPS"}]`<br><br>Для флажков: объект со свойствами `checked` и `unchecked`.<br>Пример: `{"checked": true, "unchecked": false}`<br><br>Property behavior:<br>- *required* если `type` установлено в ”List” или ”Checkbox” |

**usermacro.create**

Описание

`object usermacro.create( / hostMacros)`

Этот метод позволяет создавать новые макросы хоста.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Макросы узла сети для создания.

Метод принимает макросы узла сети со стандартными свойствами макроса узла сети.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных макросов узла сети в свойстве `hostmacroids`. Порядок возвращаемых идентификаторов соответствует порядку переданных макросов узла сети.

Примеры

Создание макроса узла сети

Создайте макрос узла сети ”{$SNMP_COMMUNITY}” со значением ”public” на узле сети ”10198”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.create",
    "params": {
        "hostid": "10198",
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "11"
        ]
    },
```

```
    "id": 1
}
```

Источник

CUserMacro::create() в *ui/include/classes/api/services/CUserMacro.php*.

### usermacro.createglobal

Описание

`object usermacro.createglobal(    /     globalMacros)`

Этот метод позволяет создавать новые глобальные макросы.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Глобальные макросы для создания.

Метод принимает глобальные макросы со стандартными свойствами глобального макроса.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных глобальных макросов в свойстве `globalmacroids`. Порядок возвращаемых идентификаторов соответствует порядку переданных глобальных макросов.

Примеры

Создание глобального макроса

Создайте глобальный макрос "{$SNMP_COMMUNITY}" со значением "public".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.createglobal",
    "params":  {
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "6"
        ]
    },
    "id": 1
}
```

Источник

CUserMacro::createGlobal() в *ui/include/classes/api/services/CUserMacro.php*.

### usermacro.delete

Описание

`object usermacro.delete(    hostMacroIds)`

Этот метод позволяет удалять макросы хоста.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID удаляемых макросов узла сети.

Возвращаемые значения

( ) Возвращает объект, который содержит ID удаленных макросов узла сети, указанных в свойстве `hostmacroids`.

Примеры

Удаление нескольких макросов узла сети

Удаление двух макросов узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.delete",
    "params": [
        "32",
        "11"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "32",
            "11"
        ]
    },
    "id": 1
}
```

Источник

CUserMacro::delete() в *ui/include/classes/api/services/CUserMacro.php*.

**usermacro.deleteglobal**

Описание

`object usermacro.createglobal( / globalMacros)`

Этот метод позволяет создавать новые глобальные макросы.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы глобальных макросов для удаления.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных глобальных макросов в свойстве `globalmacroids`.

Примеры

Удаление нескольких глобальных макросов

Удаление двух глобальных макросов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.deleteglobal",
    "params": [
        "32",
        "11"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "32",
            "11"
        ]
    },
    "id": 1
}
```

Источник

CUserMacro::deleteGlobal() в *ui/include/classes/api/services/CUserMacro.php*.

**usermacro.get**

Описание

`integer/array usermacro.get(  )`

Метод позволяет получить хостовые и глобальные макросы в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| globalmacro | flag | Вернуть глобальные макросы вместо макросов узла сети. |
| globalmacroids | ID/array | Вернуть только глобальные макросы с указанными ID. |
| groupids | ID/array | Вернуть только макросы узла сети, которые принадлежат узлам сети или шаблонам из указанных групп узлов сети или групп шаблонов. |
| hostids | ID/array | Вернуть только макросы, которые принадлежат указанным узлам сети или шаблонам. |
| hostmacroids | ID/array | Вернуть только макросы узла сети с указанными ID. |
| inherited | boolean | Если установлено значение `true`, вернуть только пользовательские макросы прототипа узла сети, унаследованные от шаблона. |

| Parameter | Type | Description |
|---|---|---|
| selectHostGroups | query | Вернуть группы узлов сети, к которым принадлежит макрос узла сети, в свойстве `hostgroups`. |
| selectHosts | query | Используется только при получении макросов узла сети.<br>Вернуть узлы сети, к которым принадлежит макрос узла сети, в свойстве `hosts`. |
| selectTemplateGroups | query | Используется только при получении макросов узла сети.<br>Вернуть группы шаблонов, к которым принадлежит макрос шаблона, в свойстве `templategroups`. |
| selectTemplates | query | Используется только при получении макросов шаблона.<br>Вернуть шаблоны, к которым принадлежит макрос узла сети, в свойстве `templates`. |
| sortfield | string/array | Используется только при получении макросов узла сети.<br>Сортировать результат по указанным свойствам.<br><br>Возможные значения: `macro`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение макросов узла сети для узла сети

Получить все макросы узла сети, определенные для узла сети "10198".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10198"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Получение макросов узла сети для шаблона

Получите макросы узла сети, определенные для шаблона ”10265”, которые содержат ”STATUS” в имени макроса.

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10265",
        "search": {
            "macro": "STATUS"
        }
    },
    "id": 1
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostmacroid": "6709",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.HOST}",
            "value": "",
            "description": "         IP-                      Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "1",
                "section_name": "",
                "label": "                  Apache",
                "description": "         IP-                      Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "814",
            "hostid": "10265",
```

```
            "macro": "{$APACHE.STATUS.PATH}",
            "value": "server-status?auto",
            "description": "    URL             Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "3",
                "section_name": "",
                "label": "               Apache",
                "description": "    URL             Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "815",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.PORT}",
            "value": "80",
            "description": "               Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "2",
                "section_name": "",
                "label": "               Apache",
                "description": "          1    65535        .",
                "required": "1",
                "regex": "^-?([0-9]+|(([0-9]+)\\.([0-9]+)))$",
                "options": []
            }
        },
        {
            "hostmacroid": "816",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.SCHEME}",
            "value": "http",
            "description": "        ,             HTTP    HTTPS.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "2",
                "priority": "4",
                "section_name": "",
                "label": "        ",
                "description": "        ,             HTTP    HTTPS.",
                "required": "0",
                "regex": "",
                "options": [
                    {
                        "value": "http",
                        "text": "HTTP"
                    },
                    {
                        "value": "https",
                        "text": "HTTPS"
                    }
                ]
            }
```

```
        }
    ],
    "id": 1
}
```

Получение глобальных макросов

Получить все глобальные макросы.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "globalmacro": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "globalmacroid": "6",
            "macro": "{$SNMP_COMMUNITY}",
            "value": "public",
            "description": "",
            "type": "0"
        }
    ],
    "id": 1
}
```

Источник

CUserMacro::get() в *ui/include/classes/api/services/CUserMacro.php*.

**usermacro.update**

Описание

`object usermacro.update(    /    hostMacros)`

Этот метод позволяет обновлять существующие макросы хоста.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства макроса узла сети, которые необходимо обновить.

Свойство `hostmacroid` должно быть определено для каждого макроса узла сети, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных макросов узла сети в свойстве `hostmacroids`.

Примеры

Изменение значения макроса узла сети

Измените значение макроса узла сети на ”public”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "public"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Изменение значения макроса, созданного правилом обнаружения

Преобразуйте созданный правилом обнаружения ”автоматический” макрос в ”ручной” и измените его значение на ”new-value”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "new-value",
        "automatic": "0"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Источник

CUserMacro::update() в *ui/include/classes/api/services/CUserMacro.php*.

**usermacro.updateglobal**

Описание

`object usermacro.updateglobal(    /     globalMacros)`

Этот метод позволяет обновлять существующие глобальные макросы.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Свойства глобального макроса, которые необходимо обновить.

Свойство globalmacroid должно быть определено для каждого глобального макроса, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы обновленных глобальных макросов в свойстве globalmacroids.

Примеры

Изменение значения глобального макроса

Измените значение глобального макроса на "public".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.updateglobal",
    "params": {
        "globalmacroid": "1",
        "value": "public"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Источник

CUserMacro::updateGlobal() в *ui/include/classes/api/services/CUserMacro.php*.

**Правило LLD**

Этот класс предназначен для работы с правилами низкоуровневого обнаружения.

Ссылки на объекты:

- Правило LLD
  - HTTP-заголовок
  - Поле HTTP-запроса
- Фильтр правила LLD
  - Условие фильтра правила LLD
- Путь макроса LLD
- Предобработка правила LLD
- Переопределения правила LLD
  - Фильтр переопределения правила LLD
    - Условие фильтра переопределения правила LLD
  - Операция переопределения правила LLD
    - Статус операции переопределения правила LLD
    - Обнаружение операции переопределения правила LLD

    * Период операции переопределения правила LLD
    * История операции переопределения правила LLD
    * Тренды операции переопределения правила LLD
    * Серьезность операции переопределения правила LLD
    * Тег операции переопределения правила LLD
    * Шаблон операции переопределения правила LLD
    * Инвентаризация операции переопределения правила LLD

Доступные методы:

- discoveryrule.create - создать новые правила LLD
- discoveryrule.delete - удалить правила LLD
- discoveryrule.get - получить правила LLD
- discoveryrule.update - обновить правила LLD

**Объект правила LLD**

Следующие объекты напрямую связаны с API `discoveryrule`.

Правило LLD

Объект правила обнаружения низкого уровня имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| itemid | ID | ID правила LLD.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| delay | string | Интервал обновления правила LLD.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) и, при необходимости, один или несколько custom intervals, разделенных точкой с запятой. Пользовательские интервалы могут представлять собой сочетание гибких и расписанных интервалов.<br><br>Принимает пользовательские макросы. Если используется макрос, значение должно быть одним макросом. Несколько макросов или макросы, смешанные с текстом, не поддерживаются. Гибкие интервалы можно записывать как два макроса, разделенные косой чертой (например, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`).<br><br>Пример:<br>`1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr`<br><br>Property behavior:<br>- *required* if `type` is set to ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), or if `type` is set to ”Zabbix agent (active)” (7) and `key_` does not contain ”mqtt.get” |
| hostid | ID | ID узла сети, которому принадлежит правило LLD.<br><br>Property behavior:<br>- *constant*<br>- *required* for create operations |

| Property | Type | Description |
|---|---|---|
| flags | integer | Origin правила обнаружения.<br><br>Возможные значения:<br>1 - правило обнаружения низкого уровня;<br>5 - правило обнаружения низкого уровня, преобразованное из прототипа.<br><br>Property behavior:<br>- *read-only* |
| interfaceid | ID | ID интерфейса узла сети для правила LLD.<br><br>Property behavior:<br>- *required* if LLD rule belongs to host and `type` is set to ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, or ”SNMP agent”<br>- *supported* if LLD rule belongs to host and `type` is set to ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent”, or ”HTTP agent” |
| key_ | string | Ключ правила LLD.<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |
| name | string | Имя правила LLD.<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |
| type | integer | Тип правила LLD.<br><br>Возможные значения:<br>0 - Zabbix agent;<br>2 - Zabbix trapper;<br>3 - Simple check;<br>5 - Zabbix internal;<br>7 - Zabbix agent (active);<br>10 - External check;<br>11 - Database monitor;<br>12 - IPMI agent;<br>13 - SSH agent;<br>14 - TELNET agent;<br>16 - JMX agent;<br>18 - Dependent item;<br>19 - HTTP agent;<br>20 - SNMP agent;<br>21 - Script;<br>22 - Browser;<br>23 - Nested. Этот тип разрешен только если родитель правила является шаблоном (предполагается, что он связан с прототипом узла сети при обнаружении) или обнаруженным узлом сети (у которого правило LLD находится в корне).<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |
| url | string | Строка URL.<br>Поддерживает пользовательские макросы, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}.<br><br>Property behavior:<br>- *required* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| allow_traps | integer | Разрешить заполнение значения аналогично элементу данных trapper.<br><br>Возможные значения:<br>0 - *(default)* Не разрешать принимать входящие данные;<br>1 - Разрешить принимать входящие данные.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent” |
| authtype | integer | Метод аутентификации.<br><br>Возможные значения, если `type` установлен в ”SSH agent”:<br>0 - *(default)* пароль;<br>1 - публичный ключ.<br><br>Возможные значения, если `type` установлен в ”HTTP agent”:<br>0 - *(default)* нет;<br>1 - basic;<br>2 - NTLM;<br>3 - Kerberos;<br>4 - Digest.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SSH agent” or ”HTTP agent”<br>- *read-only* for inherited objects (if `type` is set to ”HTTP agent”) |
| description | string | Описание правила LLD. |
| error | string | Текст ошибки, если возникли проблемы при обновлении значения правила LLD.<br><br>Property behavior:<br>- *read-only* |
| follow_redirects | integer | Следовать перенаправлениям ответа при опросе данных.<br><br>Возможные значения:<br>0 - Не следовать перенаправлениям;<br>1 - *(default)* Следовать перенаправлениям.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| headers | array | Массив headers, которые будут отправляться при выполнении HTTP-запроса.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| http_proxy | string | Строка подключения к HTTP(S) proxy.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| ipmi_sensor | string | Датчик IPMI.<br><br>Property behavior:<br>- *required* if `type` is set to ”IPMI agent” and `key_` is not set to ”ipmi.get”<br>- *supported* if `type` is set to ”IPMI agent”<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| jmx_endpoint | string | Пользовательская строка подключения JMX agent.<br><br>По умолчанию:<br>service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi<br><br>Property behavior:<br>- *supported* if `type` is set to ”JMX agent” |
| lifetime | string | Период времени, по истечении которого элементы данных, которые больше не обнаруживаются, будут удалены. Принимает секунды, единицу времени с суффиксом или пользовательский макрос.<br><br>По умолчанию: `7d`. |
| lifetime_type | integer | Сценарий удаления потерянных ресурсов LLD.<br><br>Возможные значения:<br>0 - *(default)* Удалять после достижения порога lifetime;<br>1 - Не удалять;<br>2 - Удалять немедленно. |
| enabled_lifetime | string | Период времени, по истечении которого элементы данных, которые больше не обнаруживаются, будут отключены. Принимает секунды, единицу времени с суффиксом или пользовательский макрос.<br><br>По умолчанию: `0`. |
| enabled_lifetime_type | integer | Сценарий отключения потерянных ресурсов LLD.<br><br>Возможные значения:<br>0 - Отключать после достижения порога lifetime;<br>1 - Не отключать;<br>2 - *(default)* Отключать немедленно. |
| master_itemid | ID | ID master-элемента данных.<br>Правило обнаружения не может быть master-элементом данных для другого правила обнаружения.<br><br>Property behavior:<br>- *required* if `type` is set to ”Dependent item”<br>- *read-only* for inherited objects |
| output_format | integer | Следует ли преобразовывать ответ в JSON.<br><br>Возможные значения:<br>0 - *(default)* Сохранять как есть;<br>1 - Преобразовать в JSON.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| params | string | Дополнительные параметры в зависимости от типа правила LLD:<br>- выполняемый скрипт для правил LLD SSH и Telnet;<br>- SQL-запрос для правил LLD database monitor;<br>- формула для вычисляемых правил LLD;<br>- скрипт для правил LLD script и browser.<br><br>Property behavior:<br>- *required* if `type` is set to ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Script” or ”Browser”<br>- *read-only* for inherited objects (if `type` is set to ”Script” or ”Browser”) |

| Property | Type | Description |
|---|---|---|
| parameters | object/array | Дополнительные параметры, если `type` установлен в ”Script” или ”Browser”.<br>Массив объектов со свойствами `name` и `value`, где `name` должен быть уникальным.<br><br>Property behavior:<br>- *supported* if `type` is set to ”Script” or ”Browser”<br>- *read-only* for inherited objects |
| password | string | Пароль для аутентификации.<br><br>Property behavior:<br>- *required* if `type` is set to ”JMX agent” and `username` is set<br>- *supported* if `type` is set to ”Simple check”, ”Database monitor”, ”SSH agent”, ”TELNET agent”, or ”HTTP agent”<br>- *read-only* for inherited objects (if `type` is set to ”HTTP agent”) |
| post_type | integer | Тип тела POST-данных, хранящихся в свойстве `posts`.<br><br>Возможные значения:<br>0 - *(default)* Сырые данные;<br>2 - данные JSON;<br>3 - данные XML.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| posts | string | Данные тела HTTP(S)-запроса.<br><br>Property behavior:<br>- *required* if `type` is set to ”HTTP agent” and `post_type` is set to ”JSON data” or ”XML data”<br>- *supported* if `type` is set to ”HTTP agent” and `post_type` is set to ”Raw data”<br>- *read-only* for inherited objects |
| privatekey | string | Имя файла закрытого ключа.<br><br>Property behavior:<br>- *required* if `type` is set to ”SSH agent” and `authtype` is set to ”public key” |
| publickey | string | Имя файла открытого ключа.<br><br>Property behavior:<br>- *required* if `type` is set to ”SSH agent” and `authtype` is set to ”public key” |
| query_fields | array | Массив query fields, которые будут отправляться при выполнении HTTP-запроса.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| request_method | integer | Тип метода запроса.<br><br>Возможные значения:<br>0 - *(default)* GET;<br>1 - POST;<br>2 - PUT;<br>3 - HEAD.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| retrieve_mode | integer | Какая часть ответа должна быть сохранена.<br><br>Возможные значения, если `request_method` установлен в ”GET”, ”POST” или ”PUT”:<br>0 - *(default)* Тело;<br>1 - Заголовки;<br>2 - Будут сохранены и тело, и заголовки.<br><br>Возможные значения, если `request_method` установлен в ”HEAD”:<br>1 - Заголовки.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| snmp_oid | string | SNMP OID.<br><br>Property behavior:<br>- *required* if `type` is set to ”SNMP agent”<br>- *read-only* for inherited objects |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| state | integer | Состояние правила LLD.<br><br>Возможные значения:<br>0 - *(default)* normal;<br>1 - not supported.<br><br>Property behavior:<br>- *read-only* |
| status | integer | Статус правила LLD.<br><br>Возможные значения:<br>0 - *(default)* включено;<br>1 - отключено. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми. Также поддерживает пользовательские макросы как часть списка, разделенного запятыми.<br><br>Пример: 200,200-{$M},{$M},200-400<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| templateid | ID | ID родительского правила LLD шаблона.<br><br>Property behavior:<br>- *read-only* |

| Property | Type | Description |
|---|---|---|
| timeout | string | Тайм-аут запроса опроса данных элемента данных.<br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m). Также принимает пользовательские макросы.<br><br>Диапазон допустимых значений: 1-600s.<br><br>По умолчанию: "" - использовать настройки proxy/global.<br><br>Property behavior:<br>- *supported* if `type` is set to "Zabbix agent" (0), "Simple check" (3) and `key_` does not start with "`vmware.`" and "`icmpping`", "Zabbix agent (active)" (7), "External check" (10), "Database monitor" (11), "SSH agent" (13), "TELNET agent" (14), "HTTP agent" (19), "SNMP agent" (20) and `snmp_oid` starts with "`walk[`" or "`get[`", "Script" (21), "Browser" (22)<br>- *read-only* for inherited objects |
| trapper_hosts | string | Разрешенные узлы сети.<br><br>Property behavior:<br>- *supported* if `type` is set to "Zabbix trapper", or if `type` is set to "HTTP agent" and `allow_traps` is set to "Allow to accept incoming data" |
| username | string | Имя пользователя для аутентификации.<br><br>Property behavior:<br>- *required* if `type` is set to "SSH agent", "TELNET agent", or if `type` is set to "JMX agent" and `password` is set<br>- *supported* if `type` is set to "Simple check", "Database monitor", or "HTTP agent"<br>- *read-only* for inherited objects (if `type` is set to "HTTP agent") |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных правил LLD с уже существующими. Если не указан, генерируется автоматически.<br><br>Property behavior:<br>- *supported* if the LLD rule belongs to a template |
| verify_host | integer | Следует ли проверять, что имя узла сети для соединения совпадает с именем в сертификате узла сети.<br><br>Возможные значения:<br>0 - *(default)* Не проверять;<br>1 - Проверять.<br><br>Property behavior:<br>- *supported* if `type` is set to "HTTP agent"<br>- *read-only* for inherited objects |
| verify_peer | integer | Следует ли проверять, что сертификат узла сети является подлинным.<br><br>Возможные значения:<br>0 - *(default)* Не проверять;<br>1 - Проверять.<br><br>Property behavior:<br>- *supported* if `type` is set to "HTTP agent"<br>- *read-only* for inherited objects |

HTTP-заголовок

Объект заголовка имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя HTTP-заголовка.<br>Поведение свойства:<br>- *required* |
| value | string | Значение заголовка.<br>Поведение свойства:<br>- *required* |

Поле HTTP-запроса

Объект поля запроса определяет имя и значение, которые используются для указания параметра URL. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя параметра.<br>Property behavior:<br>- *required* |
| value | string | Значение параметра.<br>Property behavior:<br>- *required* |

Фильтр правила LLD

Объект фильтра правила LLD определяет набор условий, которые можно использовать для фильтрации обнаруженных объектов. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| conditions | object/array | Набор условий фильтра, используемых для фильтрации результатов. Условия будут отсортированы в порядке их размещения в формуле.<br><br>Поведение свойства:<br>- *required* |
| evaltype | integer | Метод оценки условий фильтра.<br><br>Возможные значения:<br>0 - And/Or;<br>1 - And;<br>2 - Or;<br>3 - Custom expression.<br><br>Поведение свойства:<br>- *required* |
| eval_formula | string | Сгенерированное выражение, которое будет использоваться для оценки условий фильтра. Выражение содержит ID, которые ссылаются на конкретные условия фильтра по их `formulaid`. Значение `eval_formula` равно значению `formula` для фильтров с пользовательским выражением.<br><br>Поведение свойства:<br>- *read-only* |

| Property | Type | Description |
|---|---|---|
| formula | string | Определенное пользователем выражение, которое будет использоваться для оценки условий фильтров с пользовательским выражением. Выражение должно содержать ID, которые ссылаются на конкретные условия фильтра по их `formulaid`. ID, используемые в выражении, должны точно совпадать с ID, определенными в условиях фильтра: ни одно условие не может остаться неиспользованным или быть пропущенным.<br><br>Поведение свойства:<br>- *required* if `evaltype` is set to ”custom expression” |

Условие фильтра правила LLD

Объект условия фильтра правила LLD определяет отдельную проверку, выполняемую над значением макроса LLD. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| macro | string | Макрос LLD, над которым выполняется проверка.<br><br>Property behavior:<br>- *required* |
| value | string | Значение для сравнения.<br><br>Property behavior:<br>- *required* if `operator` is set to ”matches regular expression” or ”does not match regular expression” |
| formulaid | string | Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но будет сгенерирован заново при последующем запросе.<br><br>Property behavior:<br>- *required* if `evaltype` of LLD rule filter object is set to ”custom expression” |
| operator | integer | Оператор условия.<br><br>Possible values:<br>8 - *(default)* matches regular expression;<br>9 - does not match regular expression;<br>12 - exists;<br>13 - does not exist. |

**Note:**
Чтобы лучше понять, как использовать фильтры с различными типами выражений, см. примеры на страницах методов discoveryrule.get и discoveryrule.create.

Путь макроса LLD

Путь макроса LLD имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| lld_macro | string | Макрос LLD.<br><br>Property behavior:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| path | string | Селектор для значения, которое будет назначено соответствующему макросу.<br><br>Property behavior:<br>- *required* |

Предварительная обработка правила LLD

Объект предварительной обработки правила LLD имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип опции предварительной обработки.<br><br>Возможные значения:<br>5 - Регулярное выражение;<br>11 - XML XPath;<br>12 - JSONPath;<br>14 - Совпадает с регулярным выражением;<br>15 - Не совпадает с регулярным выражением;<br>16 - Проверка на ошибку в JSON;<br>17 - Проверка на ошибку в XML;<br>20 - Отбрасывать неизменившееся с heartbeat;<br>21 - JavaScript;<br>23 - Prometheus to JSON;<br>24 - CSV to JSON;<br>25 - Замена;<br>27 - XML to JSON;<br>28 - Значение SNMP walk;<br>29 - SNMP walk to JSON;<br>30 - Значение SNMP get.<br><br>Property behavior:<br>- *required* |
| params | string | Дополнительные параметры, используемые опцией предварительной обработки. Несколько параметров разделяются символом новой строки (\n).<br><br>Property behavior:<br>- *required* если `type` установлено в ”Регулярное выражение” (5), ”XML XPath” (11), ”JSONPath” (12), ”Совпадает с регулярным выражением” (14), ”Не совпадает с регулярным выражением” (15), ”Проверка на ошибку в JSON” (16), ”Проверка на ошибку в XML” (17), ”Отбрасывать неизменившееся с heartbeat” (20), ”JavaScript” (21), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”Замена” (25), ”Значение SNMP walk” (28), ”SNMP walk to JSON” (29) или ”Значение SNMP get” (30) |

| Property | Type | Description |
|---|---|---|
| error_handler | integer | Тип действия, используемый в случае сбоя шага предварительной обработки.<br><br>Возможные значения:<br>0 - Сообщение об ошибке задается сервером Zabbix;<br>1 - Отбросить значение;<br>2 - Установить пользовательское значение;<br>3 - Установить пользовательское сообщение об ошибке.<br><br>Property behavior:<br>- *required* если `type` установлено в ”Регулярное выражение” (5), ”XML XPath” (11), ”JSONPath” (12), ”Совпадает с регулярным выражением” (14), ”Не совпадает с регулярным выражением” (15), ”Проверка на ошибку в JSON” (16), ”Проверка на ошибку в XML” (17), ”Prometheus to JSON” (23), ”CSV to JSON” (24), ”XML to JSON” (27), ”Значение SNMP walk” (28), ”SNMP walk to JSON” (29) или ”Значение SNMP get” (30) |
| error_handler_params | string | Параметры обработчика ошибок.<br><br>Property behavior:<br>- *required* если `error_handler` установлено в ”Установить пользовательское значение” или ”Установить пользовательское сообщение об ошибке” |

Для каждого типа предварительной обработки поддерживаются следующие параметры и обработчики ошибок.

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 5 | Регулярное выражение | 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 | Совпадает с регулярным выражением | pattern[1] | | | 0, 1, 2, 3 |
| 15 | Не совпадает с регулярным выражением | pattern[1] | | | 0, 1, 2, 3 |
| 16 | Проверка на ошибку в JSON | path[3] | | | 0, 1, 2, 3 |
| 17 | Проверка на ошибку в XML | path[3] | | | 0, 1, 2, 3 |
| 20 | Отбрасывать неизменившееся с heart-beat | seconds^4, 5^ | | | |
| 21 | JavaScript | script[2] | | | |
| 23 | Prometheus to JSON | pattern^5, 6^ | | | 0, 1, 2, 3 |

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 24 | CSV to JSON | character[2] | character[2] | 0,1 | 0, 1, 2, 3 |
| 25 | Замена | search string[2] | replacement[2] | | |
| 27 | XML to JSON | | | | 0, 1, 2, 3 |
| 28 | Значение SNMP walk | OID[2] | Format:<br>0 - Без изменений<br>1 - UTF-8 из Hex-STRING<br>2 - MAC из Hex-STRING<br>3 - Integer из BITS | | 0, 1, 2, 3 |
| 29 | SNMP walk to JSON[7] | Field name[2] | OID prefix[2] | Format:<br>0 - Без изменений<br>1 - UTF-8 из Hex-STRING<br>2 - MAC из Hex-STRING<br>3 - Integer из BITS | 0, 1, 2, 3 |
| 30 | Значение SNMP get | Format:<br>1 - UTF-8 из Hex-STRING<br>2 - MAC из Hex-STRING<br>3 - Integer из BITS | | | 0, 1, 2, 3 |

[1] регулярное выражение<br> [2] строка<br> [3] JSONPath или XML XPath<br> [4] положительное целое число (с поддержкой суффиксов времени, например 30s, 1m, 2h, 1d)<br> [5] пользовательский макрос<br> [6] шаблон Prometheus со следующим синтаксисом: `<metric name>{<label name>="<label value>", ...} == <value>`. Каждый компонент шаблона Prometheus (имя метрики, имя метки, значение метки и значение метрики) может быть пользовательским макросом.<br> [7] Поддерживает несколько записей "Field name,OID prefix,Format records", разделенных символом новой строки.

Переопределения правил LLD

Объект переопределений правил LLD определяет набор правил (фильтры, условия и операции), которые используются для переопределения свойств различных объектов-прототипов. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Уникальное имя переопределения.<br><br>Property behavior:<br>- *required* |
| step | integer | Уникальный порядковый номер переопределения.<br><br>Property behavior:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| stop | integer | Остановить обработку следующих переопределений, если есть совпадение.<br><br>Possible values:<br>0 - *(default)* не останавливать обработку переопределений;<br>1 - остановить обработку переопределений, если фильтр совпадает. |
| filter | object | Фильтр переопределения. |
| operations | object/array | Операции переопределения. |

Фильтр переопределения правила LLD

Объект фильтра переопределения правила LLD определяет набор условий, которые, если они совпадают с обнаруженным объектом, приводят к применению переопределения. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| conditions | object/array | Набор условий фильтра переопределения, используемых для сопоставления обнаруженных объектов. Условия будут отсортированы в порядке их размещения в формуле.<br><br>Поведение свойства:<br>- *required* |
| evaltype | integer | Метод оценки условий фильтра переопределения.<br><br>Возможные значения:<br>0 - And/Or;<br>1 - And;<br>2 - Or;<br>3 - Custom expression.<br><br>Поведение свойства:<br>- *required* |
| eval_formula | string | Сгенерированное выражение, которое будет использоваться для оценки условий фильтра переопределения. Выражение содержит идентификаторы, которые ссылаются на конкретные условия фильтра переопределения по их `formulaid`. Значение `eval_formula` равно значению `formula` для фильтров с пользовательским выражением.<br><br>Поведение свойства:<br>- *read-only* |
| formula | string | Определяемое пользователем выражение, используемое для оценки условий фильтров переопределения с пользовательским выражением. Выражение должно содержать идентификаторы, которые ссылаются на конкретные условия фильтра переопределения по их `formulaid`. Идентификаторы, используемые в выражении, должны в точности совпадать с идентификаторами, определенными в условиях фильтра переопределения: ни одно условие не может остаться неиспользованным или опущенным.<br><br>Поведение свойства:<br>- *required* if `evaltype` is set to ”custom expression” |

Условие фильтра переопределения правила LLD

Объект условия фильтра переопределения правила LLD определяет отдельную проверку, выполняемую для значения макроса LLD. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| macro | string | Макрос LLD, для которого выполняется проверка.<br><br>Property behavior:<br>- *required* |
| value | string | Значение для сравнения.<br><br>Property behavior:<br>- *required* if `operator` is set to ”matches regular expression” or ”does not match regular expression” |
| formulaid | string | Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но будет сгенерирован заново при последующем запросе.<br><br>Property behavior:<br>- *required* if `evaltype` of LLD rule override filter object is set to ”custom expression” |
| operator | integer | Оператор условия.<br><br>Possible values:<br>8 - *(default)* matches regular expression;<br>9 - does not match regular expression;<br>12 - exists;<br>13 - does not exist. |

Операция переопределения правила LLD

Операция переопределения правила LLD представляет собой комбинацию условий и действий, выполняемых над объектом-прототипом. Она имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| operationobject | integer | Тип обнаруженного объекта, для которого выполняется действие.<br><br>Возможные значения:<br>0 - прототип элемента данных;<br>1 - прототип триггера;<br>2 - прототип графика;<br>3 - прототип узла сети;<br>4 - прототип обнаружения.<br><br>Поведение свойства:<br>- *required* |
| operator | integer | Оператор условия переопределения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* равно;<br>1 - не равно;<br>2 - содержит;<br>3 - не содержит;<br>8 - соответствует;<br>9 - не соответствует. |

| Property | Type | Description |
|---|---|---|
| value | string | Шаблон для сопоставления с именем элемента данных, триггера, графика или прототипа узла сети в зависимости от выбранного объекта. |
| opstatus | object | Объект переопределения статуса операции для объектов типа элемент данных, триггер и прототип узла сети. |
| opdiscover | object | Объект переопределения статуса обнаружения операции (для всех типов объектов). |
| opperiod | object | Объект переопределения периода операции (интервала обновления) для объекта типа прототип элемента данных. |
| ophistory | object | Объект переопределения истории операции для объекта типа прототип элемента данных. |
| optrends | object | Объект переопределения трендов операции для объекта типа прототип элемента данных. |
| opseverity | object | Объект переопределения серьезности операции для объекта типа прототип триггера. |
| optag | object/array | Объект переопределения тега операции для объектов типа прототип триггера и прототип узла сети. |
| optemplate | object/array | Объект переопределения шаблона операции для объекта типа прототип узла сети. |
| opinventory | object | Объект переопределения инвентаря операции для объекта типа прототип узла сети. |

Статус операции переопределения правила LLD

Статус операции переопределения правила LLD, который задается для обнаруженного объекта. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| status | integer | Переопределить статус для выбранного объекта.<br><br>Возможные значения:<br>0 - Создать включенным;<br>1 - Создать отключенным.<br><br>Поведение свойства:<br>*- required* |

Операция переопределения правила LLD: обнаружение

Операция переопределения правила LLD: статус обнаружения, который задается для обнаруженного объекта. Она имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| discover | integer | Переопределяет статус обнаружения для выбранного объекта.<br><br>Возможные значения:<br>0 - Да, продолжать обнаружение объектов;<br>1 - Нет, новые объекты не будут обнаруживаться, а существующие будут помечены как потерянные.<br><br>Поведение свойства:<br>*- required* |

Период операции переопределения правила LLD

Период операции переопределения правила LLD — это значение интервала обновления, которое задается для обнаруженного элемента данных. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| delay | string | Переопределяет интервал обновления прототипа элемента данных.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) и, при необходимости, один или несколько пользовательских интервалов, разделенных точкой с запятой. Пользовательские интервалы могут представлять собой сочетание гибких интервалов и интервалов по расписанию.<br><br>Принимает пользовательские макросы или макросы LLD (поддержка макросов LLD восстановлена в Zabbix 7.4.11). Если используется, значение должно быть одним макросом. Несколько макросов или макросы, смешанные с текстом, не поддерживаются. Гибкие интервалы можно записывать как два макроса, разделенные косой чертой (например, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`).<br><br>Пример:<br>`1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr`<br><br>Поведение свойства:<br>- *required* |

История операции переопределения правила LLD

Значение истории операции переопределения правила LLD, которое задается для обнаруженного элемента данных. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| history | string | Переопределяет историю прототипа элемента данных, задавая единицу времени, в течение которой должны храниться данные истории. Также принимает пользовательский макрос и LLD-макрос (поддержка LLD-макросов восстановлена в Zabbix 7.4.11).<br><br>Property behavior:<br>- *required* |

Переопределение тенденций правила LLD

Переопределение тенденций правила LLD — значение, которое задается для обнаруженного элемента данных. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| trends | string | Переопределяет тенденции прототипа элемента данных, то есть задает единицу времени, в течение которой должны храниться данные тенденций. Также принимает пользовательский макрос и LLD-макрос (поддержка LLD-макросов восстановлена в Zabbix 7.4.11).<br><br>Property behavior:<br>- *required* |

Уровень серьезности операции переопределения правила LLD

Значение уровня серьезности операции переопределения правила LLD, которое задается для обнаруженного триггера. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| severity | integer | Переопределяет уровень серьезности прототипа триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа.<br><br>Поведение свойства:<br>- *required* |

Тег операции переопределения правила LLD

Объект тега операции переопределения правила LLD содержит имя тега и значение, которые задаются для обнаруженного объекта. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| tag | string | Новое имя тега. Также принимает пользовательский макрос и LLD-макрос.<br><br>Property behavior:<br>- *required* |
| value | string | Новое значение тега. Также принимает пользовательский макрос и LLD-макрос. |

Шаблон операции переопределения правила LLD

Объект шаблона операции переопределения правила LLD, который связан с обнаруженным узлом сети. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| templateid | ID | Переопределить шаблон связанных шаблонов прототипа узла сети.<br><br>Property behavior:<br>- *required* |

Инвентарь операции переопределения правила LLD

Значение режима инвентаря операции переопределения правила LLD, которое задается для обнаруженного узла сети. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| inventory_mode | integer | Переопределяет режим инвентаря прототипа узла сети.<br><br>Возможные значения:<br>-1 - отключено;<br>0 - *(по умолчанию)* вручную;<br>1 - автоматически.<br><br>Поведение свойства:<br>- *required* |

**discoveryrule.create**

Описание

`object discoveryrule.create( / lldRules)`

Этот метод позволяет создавать новые правила LLD.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Правила LLD для создания.

В дополнение к стандартным свойствам правила LLD, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| filter | object | Фильтр правила LLD для правила LLD. |
| preprocessing | object/array | Параметры предобработки правила LLD. |
| lld_macro_paths | object/array | Параметры lld_macro_path правила LLD. |
| overrides | object/array | Параметры переопределений правила LLD. |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID созданных правил LLD, указанных в свойстве `itemids`. Порядок возвращаемых ID совпадает с порядком переданных правил LLD.

Примеры

Создание правила LLD

Создайте правило LLD агента Zabbix для обнаружения смонтированных файловых систем. Обнаруженные элементы данных будут обновляться каждые 30 секунд.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "                          ",
        "key_": "vfs.fs.discovery",
        "hostid": "10197",
        "type": 0,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Использование фильтра

Создайте правило LLD с набором условий для фильтрации результатов. Условия будут объединены с помощью логического оператора "and".

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Создание правила LLD с путями макросов

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "     LLD            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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Использование пользовательского фильтра выражения

Создайте правило LLD с фильтром, который будет использовать пользовательское выражение для оценки условий. Правило LLD должно обнаруживать только те объекты, у которых значение макроса "{#MACRO1}" соответствует одновременно регулярным выражениям "regex1" и "regex2", а значение "{#MACRO2}" соответствует либо "regex3", либо "regex4". Идентификаторы формулы "A", "B", "C" и "D" выбраны произвольно.

Request:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Использование пользовательских полей запроса и заголовков

Создайте правило LLD с пользовательскими полями запроса и заголовками.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
    "id": 1
}
```

Создание правила LLD с предварительной обработкой

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Discovery rule with preprocessing",
        "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Создание правила LLD с переопределениями

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "30980"
        ]
    },
    "id": 1
}
```

Создание правила LLD для скрипта

Создайте простой сбор данных с помощью правила LLD для скрипта.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
```

```
        ]
    },
    "id": 1
}
```

Создание правила LLD с заданным периодом времени для отключения и без удаления

Создайте правило LLD с пользовательским периодом времени для отключения сущности после того, как она больше не обнаруживается, с настройкой, при которой она никогда не будет удалена.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

См. также

- Фильтр правила LLD
- Пути макросов LLD
- Предобработка правила LLD

Источник

CDiscoveryRule::create() в *ui/include/classes/api/services/CDiscoveryRule.php*.

**discoveryrule.delete**

Описание

`object discoveryrule.delete(array lldRuleIds)`

Этот метод позволяет удалять правила LLD.

> **Note:**
> Этот метод доступен только для типов пользователей *Администратор* и *Супер-администратор*. Разрешения на вызов метода могут быть отозваны в настройках роли пользователей. Смотрите Роли пользователей для получения дополнительной информации.

Параметры

( ) ID удаляемых правил LLD.

Возвращаемые значения

( ) Возвращает объект, который содержит идентификаторы удаленных правил LLD, указанных в свойстве `ruleids`.

Примеры

Удаление нескольких правил LLD

Удалить два правила LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.delete",
    "params": [
    "27665",
    "27668"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
     "ruleids": [
     "27665",
     "27668"
    ]
    },
    "id": 1
}
```

Источник

CDiscoveryRule::delete() в *ui/include/classes/api/services/CDiscoveryRule.php*.

**discoveryrule.get**

Описание

`integer/array discoveryrule.get(  )`

Этот метод позволяет получать правила LLD в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| itemids | ID/array | Возвращать только правила LLD с указанными ID. |
| groupids | ID/array | Возвращать только правила LLD, которые принадлежат узлам сети из указанных групп. |
| hostids | ID/array | Возвращать только правила LLD, которые принадлежат указанным узлам сети. |
| inherited | boolean | Если установлено значение `true`, возвращать только правила LLD, унаследованные от шаблона. |
| interfaceids | ID/array | Возвращать только правила LLD, использующие указанные интерфейсы узлов сети. |
| monitored | boolean | Если установлено значение `true`, возвращать только включенные правила LLD, которые принадлежат контролируемым узлам сети. |
| templated | boolean | Если установлено значение `true`, возвращать только правила LLD, которые принадлежат шаблонам. |

| Parameter | Type | Description |
|---|---|---|
| templateids | ID/array | Возвращать только правила LLD, которые принадлежат указанным шаблонам. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения правила LLD. Объект обнаружения правила LLD связывает обнаруженное правило LLD с прототипом правила LLD, из которого оно было обнаружено. |
| selectDiscoveryRulePrototypes | query | Возвращать свойство `discoveryRulePrototypes` с прототипами правил LLD, которые принадлежат правилу LLD.<br><br>Поддерживает `count`. |
| selectFilter | query | Возвращать свойство `filter` с данными фильтра, используемого правилом LLD. |
| selectGraphs | query | Возвращает свойство `graphs` с прототипами графиков, которые принадлежат правилу LLD.<br><br>Поддерживает `count`. |
| selectHostPrototypes | query | Возвращать свойство `hostPrototypes` с прототипами узлов сети, которые принадлежат правилу LLD.<br><br>Поддерживает `count`. |
| selectHosts | query | Возвращать свойство `hosts` с массивом узлов сети, к которым принадлежит правило LLD. |
| selectItems | query | Возвращать свойство `items` с прототипами элементов данных, которые принадлежат правилу LLD.<br><br>Поддерживает `count`. |
| selectTriggers | query | Возвращать свойство `triggers` с прототипами триггеров, которые принадлежат правилу LLD.<br><br>Поддерживает `count`. |
| selectLLDMacroPaths | query | Возвращать свойство `lld_macro_paths` со списком макросов LLD и путей к значениям, назначенным каждому соответствующему макросу. |
| selectPreprocessing | query | Возвращать свойство `preprocessing` с параметрами предварительной обработки правила LLD. |
| selectOverrides | query | Возвращать свойство `lld_rule_overrides` со списком фильтров переопределения, условий и операций, выполняемых над объектами прототипов. |
| filter | object | Возвращать только те результаты, которые точно соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений для сопоставления.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит правило LLD. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подвыборками.<br><br>Применяется к следующим подвыборкам: `selectItems`, `selectGraphs`, `selectTriggers`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `itemid`, `name`, `key_`, `delay`, `type`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |

| Parameter | Type | Description |
|---|---|---|
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение правил обнаружения с узла сети

Получить все правила обнаружения для указанного ID узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": "extend",
        "hostids": "10202"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "27425",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10202",
            "name": "                    ",
            "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": "                    ,                                \"Network interfaces for disc
            "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",
        "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": "                          ",
        "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": "                          ,                                \"File systems for disco
        "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",
        "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
}
```

Получение условий фильтра

Получите имя правила LLD "24681" и его условия фильтра. Фильтр использует тип оценки "and", поэтому свойство `formula` пустое, а `eval_formula` генерируется автоматически.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "selectFilter": "extend",
        "itemids": ["24681"]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Получение правила LLD по URL

Получение правила LLD для узла сети по значению поля URL правила. Поддерживается только точное совпадение строки URL, определенной для правила LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "hostids": "10257",
        "filter": {
            "type": 19,
            "url": "http://127.0.0.1/discoverer.php"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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",
            "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
}
```


Получение правила LLD с переопределениями

Получение одного правила LLD с различными настройками переопределений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "itemids": "30980",
        "selectOverrides": ["name", "step", "stop", "filter", "operations"]
    },
    "id": 1
}
```

Ответ:


```
{
    "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
}
```


См. также

- Прототип графика
- Узел сети
- Прототип элемента данных
- Фильтр правила LLD
- Прототип триггера

Источник

CDiscoveryRule::get() в *ui/include/classes/api/services/CDiscoveryRule.php*.

**discoveryrule.update**

Описание

`object discoveryrule.update(  /  lldRules)`

Этот метод позволяет обновлять существующие правила LLD.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства правила LLD, которые необходимо обновить.

Свойство `itemid` должно быть определено для каждого правила LLD, все остальные свойства являются необязательными. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам правила LLD, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| filter | object | Фильтр правила LLD для замены существующего фильтра. |
| preprocessing | object/array | Параметры предобработки правила LLD для замены существующих параметров предобработки.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |
| lld_macro_paths | object/array | Параметры lld_macro_path правила LLD для замены существующих параметров lld_macro_path.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |
| overrides | object/array | Параметры переопределений правила LLD для замены существующих параметров переопределений.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных правил LLD, указанных в свойстве `itemids`.

Примеры

Добавление фильтра к правилу LLD

Добавьте фильтр, чтобы содержимое макроса *{#FSTYPE}* соответствовало регулярному выражению *@File systems for discovery*.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "filter": {
            "evaltype": 1,
            "conditions": [
                {
                    "macro": "{#FSTYPE}",
                    "value": "@File systems for discovery"
                }
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Добавление путей макросов LLD

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "lld_macro_paths": [
            {
                "lld_macro": "{#MACRO1}",
                "path": "$.json.path"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Отключение trap-обработки

Отключить trap-обработку LLD для правила обнаружения.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "28336",
        "allow_traps": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
    "id": 1
}
```

Обновление параметров предварительной обработки правила LLD

Обновите правило LLD с правилом предварительной обработки "JSONPath".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Обновление скрипта правила LLD

Обновите скрипт правила LLD, указав другой скрипт, и удалите ненужные параметры, которые использовались предыдущим скриптом.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Обновление срока жизни правила LLD

Обновите правило LLD, чтобы отключать больше не обнаруживаемый объект через 12 часов и удалять его через 7 дней.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "46864",
        "lifetime_type": 0,
        "lifetime": "7d",
        "enabled_lifetime_type": 0,
        "enabled_lifetime": "12h"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

Источник

CDiscoveryRule::update() в *ui/include/classes/api/services/CDiscoveryRule.php*.

### Правило обнаружения

Этот класс предназначен для работы с правилами обнаружения сети.

**Note:**
Этот API предназначен для работы с правилами обнаружения сети. Для правил низкоуровневого обнаружения см. API правила LLD.

Ссылки на объекты:

- Правило обнаружения

Доступные методы:

- drule.create - создать новые правила обнаружения сети
- drule.delete - удалить правила обнаружения сети
- drule.get - получить правила обнаружения сети
- drule.update - обновить правила обнаружения сети

### Объект правила обнаружения

Следующие объекты напрямую связаны с API `drule`.

Правило обнаружения

Объект правила обнаружения определяет правило сетевого обнаружения. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| druleid | ID | ID правила обнаружения.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| iprange | string | Один или несколько диапазонов IP-адресов для проверки, разделенных запятыми.<br><br>Дополнительные сведения о поддерживаемых форматах диапазонов IP-адресов см. в разделе настройка сетевого обнаружения.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Имя правила обнаружения.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| delay | string | Интервал выполнения правила обнаружения.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) либо пользовательский макрос.<br><br>По умолчанию: 1h. |
| proxyid | ID | ID прокси, используемого для обнаружения. |
| status | integer | Включено ли правило обнаружения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включено;<br>1 - отключено. |
| concurrency_max | integer | Максимальное число одновременных проверок на одно правило обнаружения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* неограниченное число проверок;<br>1 - одна проверка;<br>2-999 - настраиваемое число проверок. |
| error | string | Текст ошибки, если при выполнении правила обнаружения возникли какие-либо проблемы.<br><br>Поведение свойства:<br>- *только для чтения* |

**drule.create**

Описание

`object drule.create( / discoveryRules)`

Этот метод позволяет создавать новые правила обнаружения.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Правила обнаружения для создания.

Помимо стандартных свойств правила обнаружения, метод принимает следующие параметры.

| Параметр | Тип | Описание |
|---|---|---|
| dchecks | array | Проверки обнаружения для создания для правила обнаружения.<br><br>Поведение параметра:<br>- *обязательный* |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID созданных правил обнаружения, указанных в свойстве `druleids`. Порядок возвращаемых ID совпадает с порядком переданных правил обнаружения.

Примеры

Создание правила обнаружения

Создайте правило обнаружения, чтобы найти машины, на которых запущен агент Zabbix, в локальной сети. Правило должно использовать одну проверку агента Zabbix на порту 10050.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

См. также

- Проверка обнаружения

Источник

CDRule::create() в *ui/include/classes/api/services/CDRule.php*.

**drule.delete**

Описание

`object drule.delete(    discoveryRuleIds)`

Этот метод позволяет удалять правила обнаружения.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых правил обнаружения.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных правил обнаружения, указанных в свойстве `druleids`.

Примеры

Удаление нескольких правил обнаружения

Удаление двух правил обнаружения.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "drule.delete",
    "params": [
        "4",
        "6"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "4",
            "6"
        ]
    },
    "id": 1
}
```

Источник

CDRule::delete() в *ui/include/classes/api/services/CDRule.php*.

**drule.get**

Описание

`integer/array drule.get(           )`

Этот метод позволяет получать правила обнаружения в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dhostids | ID/array | Возвращать только правила обнаружения, которые создали указанные обнаруженные узлы сети. |
| druleids | ID/array | Возвращать только правила обнаружения с указанными ID. |
| dserviceids | ID/array | Возвращать только правила обнаружения, которые создали указанные обнаруженные сервисы. |
| selectDChecks | query | Возвращать свойство `dchecks` с проверками обнаружения, используемыми правилом обнаружения.<br><br>Поддерживает `count`. |
| selectDHosts | query | Возвращать свойство `dhosts` с обнаруженными узлами сети, созданными правилом обнаружения.<br><br>Поддерживает `count`. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам:<br>`selectDChecks` - результаты будут отсортированы по `dcheckid`;<br>`selectDHosts` - результаты будут отсортированы по `dhostsid`. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `druleid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получить все правила обнаружения

Получить все настроенные правила обнаружения и проверки обнаружения, которые они используют.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "drule.get",
    "params": {
        "output": "extend",
        "selectDChecks": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Обнаруженный узел сети
- Проверка обнаружения

Источник

CDRule::get() в *ui/include/classes/api/services/CDRule.php*.

**drule.update**

Описание

```
drule.update(object/array discoveryRules)
```

Этот метод позволяет обновлять существующие правила обнаружения.

**Note:**
Этот метод доступен только пользователям с ролью *Admin* или *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. См. Роли пользователей для получения дополнительной информации.

Параметры

`(object/array)` Свойства правила обнаружения, которые необходимо обновить.

Для каждого правила обнаружения должно быть определено свойство `druleid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам правила обнаружения, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dchecks | array | Проверки обнаружения для замены существующих проверок. |

Возвращаемые значения

`(object)` Возвращает объект, который содержит ID обновленных правил обнаружения под свойством `druleids`.

Примеры

Изменение диапазона IP адресов в правиле обнаружения

Изменение в правила обнаружения диапазона IP адресов на ”192.168.2.1-255”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "drule.update",
    "params": {
        "druleid": "6",
        "iprange": "192.168.2.1-255"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
```

```
    },
    "id": 1
}
```

См. также

- Проверка обнаружения

Источник

CDRule::update() в *ui/include/classes/api/services/CDRule.php*.

#### Проблема

Этот класс предназначен для работы с проблемами.

Ссылки на объекты:

- Проблема
  - URL типа медиа
- Тег проблемы

Доступные методы:

- problem.get - получить проблемы

#### Объект problem

Следующие объекты напрямую связаны с API `problem`.

Проблема

> **Note:**
> Проблемы создаются сервером Zabbix и не могут быть изменены через API.

Объект проблемы имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| eventid | ID | ID события проблемы. |
| source | integer | Тип события проблемы.<br><br>Возможные значения:<br>0 - событие, созданное триггером;<br>3 - внутреннее событие;<br>4 - событие, созданное при обновлении статуса сервиса. |
| object | integer | Тип объекта, связанного с событием проблемы.<br><br>Возможные значения, если `source` установлено в ”event created by a trigger”:<br>0 - триггер.<br><br>Возможные значения, если `source` установлено в ”internal event”:<br>0 - триггер;<br>4 - элемент данных;<br>5 - правило LLD.<br><br>Возможные значения, если `source` установлено в ”event created on service status update”:<br>6 - сервис. |
| objectid | ID | ID связанного объекта. |
| clock | timestamp | Время создания события проблемы. |
| ns | integer | Наносекунды в момент создания события проблемы. |
| r_eventid | ID | ID события восстановления. |
| r_clock | timestamp | Время создания события восстановления. |
| r_ns | integer | Наносекунды в момент создания события восстановления. |

| Property | Type | Description |
|---|---|---|
| cause_eventid | ID | ID события причины. |
| correlationid | ID | ID правила корреляции, если это событие было восстановлено глобальным правилом корреляции. |
| userid | ID | ID пользователя, который закрыл проблему (если проблема была закрыта вручную). |
| name | string | Имя решенной проблемы. |
| acknowledged | integer | Состояние подтверждения проблемы.<br><br>Возможные значения:<br>0 - не подтверждена;<br>1 - подтверждена. |
| severity | integer | Текущая серьезность проблемы.<br><br>Возможные значения:<br>0 - не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа. |
| suppressed | integer | Подавлена ли проблема.<br><br>Возможные значения:<br>0 - проблема находится в нормальном состоянии;<br>1 - проблема подавлена. |
| opdata | string | Оперативные данные с раскрытыми макросами. |
| urls | array | Активные URL-адреса типа носителя. |

URL типа оповещения

Объект URL типа оповещения имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя URL, определенное для типа оповещения. |
| url | string | Значение URL, определенное для типа оповещения. |

Результаты будут содержать записи только для активных типов оповещения с включенным пунктом меню событий. Макрос, используемый в свойствах, будет раскрыт, но если одно из свойств содержит нераскрытый макрос, оба свойства будут исключены из результатов. Поддерживаемые макросы см. в разделе *Поддерживаемые макросы*.

Тег проблемы

Объект тега проблемы имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега проблемы. |
| value | string | Значение тега проблемы. |

**problem.get**

Описание

`integer/array problem.get(object parameters)`

Метод позволяет получать проблемы в соответствии с заданными параметрами.

Этот метод предназначен для получения нерешенных проблем. При необходимости также можно дополнительно получать недавно решенные проблемы. Период, определяющий, насколько "недавно" — это недавно, задается в разделе *Администрирование* → *Общие настройки*. Проблемы, которые были решены раньше этого периода, не хранятся в таблице проблем. Чтобы получить проблемы, решенные более давно, используйте метод `event.get`.

**Attention:**
Этот метод может возвращать проблемы удаленной сущности, если эти проблемы еще не были удалены housekeeper.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках роли пользователя. См. Роли пользователей для получения дополнительной информации.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| eventids | ID/array | Возвращать только проблемы с указанными ID. |
| groupids | ID/array | Возвращать только проблемы, созданные объектами, которые принадлежат указанным группам узлов сети. |
| hostids | ID/array | Возвращать только проблемы, созданные объектами, которые принадлежат указанным узлам сети. |
| objectids | ID/array | Возвращать только проблемы, созданные указанными объектами. |
| source | integer | Возвращать только проблемы с указанным типом.<br><br>Список поддерживаемых типов событий см. на странице объекта события проблемы.<br><br>По умолчанию: 0 - проблема, созданная триггером. |
| object | integer | Возвращать только проблемы, созданные объектами указанного типа.<br><br>Список поддерживаемых типов объектов см. на странице объекта события проблемы.<br><br>По умолчанию: 0 - триггер. |
| acknowledged | boolean | Если задано `true`, возвращать только подтвержденные проблемы. |
| action | integer | Возвращать только проблемы, для которых были выполнены указанные действия обновления события. Для нескольких действий используйте битовую маску как сумму любых допустимых значений bitmap (например, 34 для подтверждения и подавления события). |
| action_userids | ID/array | Возвращать только проблемы с указанными ID пользователей, выполнивших действия обновления события проблемы. |
| suppressed | boolean | Если задано `true`, возвращать только подавленные проблемы. |
| symptom | boolean | Если задано `true`, возвращать только события проблем-симптомов. |
| severities | integer/array | Возвращать только проблемы с указанными уровнями важности события. Применяется только если object - триггер. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И;<br>2 - ИЛИ. |

| Parameter | Type | Description |
|---|---|---|
| tags | array | Возвращать только проблемы с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все проблемы.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| recent | boolean | Если задано `true`, возвращать недавние проблемы, включая те, что были решены в пределах `ok_period`, а также любые нерешенные проблемы.<br>Если задано `false`, возвращать только нерешенные проблемы. |
| eventid_from | string | Возвращать только проблемы с ID больше или равным указанному ID. |
| eventid_till | string | Возвращать только проблемы с ID меньше или равным указанному ID. |
| time_from | timestamp | Возвращать только проблемы, созданные после указанного времени или в указанное время. |
| time_till | timestamp | Возвращать только проблемы, созданные до указанного времени или в указанное время. |
| selectAcknowledges | query | Возвращать свойство `acknowledges` с обновлениями проблемы.<br>Обновления проблемы сортируются в обратном хронологическом порядке.<br><br>Объект обновления проблемы имеет следующие свойства:<br>`acknowledgeid` - `(ID)` ID обновления;<br>`userid` - `(ID)` ID пользователя, который обновил событие;<br>`eventid` - `(ID)` ID обновленного события;<br>`clock` - `(timestamp)` время обновления события;<br>`message` - `(string)` текст сообщения;<br>`action` - `(integer)`тип действия обновления (см. `event.acknowledge`);<br>`old_severity` - `(integer)` уровень важности события до этого действия обновления;<br>`new_severity` - `(integer)` уровень важности события после этого действия обновления;<br>`suppress_until` - `(timestamp)` время, до которого событие будет подавлено;<br>`taskid` - `(ID)` ID задачи, если текущее событие находится в процессе изменения ранга.<br><br>Поддерживает `count`. |
| selectTags | query | Возвращать свойство `tags` с тегами проблемы. Формат вывода: `[{"tag": "<tag>", "value": "<value>"}, ...]`. |
| selectSuppressionData | query | Возвращать свойство `suppression_data` со списком активных обслуживаний и ручных подавлений:<br>`maintenanceid` - `(ID)` ID обслуживания;<br>`userid` - `(ID)` ID пользователя, который подавил проблему;<br>`suppress_until` - `(integer)` время, до которого проблема подавлена. |
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи - это имена свойств, а значения - либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Не поддерживает свойства с типом данных `text`. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `eventid`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение событий проблем триггера

Получить недавние события из триггера ”15112.”

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "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",
            "opdata": "",
            "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"
                }
            ],
            "suppression_data": [
                {
                    "maintenanceid": "15",
                    "suppress_until": "1472511600",
                    "userid": "0"
                }
            ],
            "suppressed": "1",
            "tags": [
                {
                    "tag": "test-tag",
                    "value": "test-value"
                }
            ]
        }
    ],
    "id": 1
}
```

Получение проблем, подтвержденных указанным пользователем

Получение проблем, подтвержденных пользователем с ID=10

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "problem.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Ответ:

```
{
    "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",
            "opdata": "",
            "acknowledges": [
                {
                    "userid": "10",
                    "action": "2"
                }
            ],
            "suppressed": "0"
        }
    ],
    "id": 1
}
```


См. также

- Оповещение
- Элемент данных
- Узел сети
- Правило LLD
- Триггер

Источник

CEvent::get() в *ui/include/classes/api/services/CProblem.php*.

#### Проверка обнаружения

Этот класс предназначен для работы с проверками обнаружения.

Ссылки на объекты:

- Проверка обнаружения

Доступные методы:

- dcheck.get - получить проверки обнаружения

#### Объект проверки обнаружения

Следующие объекты напрямую связаны с API `dcheck`.

Проверка обнаружения

Объект проверки обнаружения определяет конкретную проверку, выполняемую правилом сетевого обнаружения. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dcheckid | ID | ID проверки обнаружения. |
| druleid | ID | ID правила обнаружения, которому принадлежит проверка. |

| Property | Type | Description |
|---|---|---|
| key_ | string | Ключ элемента данных (если `type` установлен в ”Zabbix agent”) или SNMP OID (если `type` установлен в ”SNMPv1 agent”, ”SNMPv2 agent” или ”SNMPv3 agent”).<br><br>Property behavior:<br>- *required* if `type` is set to ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent”, or ”SNMPv3 agent” |
| ports | string | Один или несколько диапазонов портов для проверки, разделенных запятыми.<br><br>Default: 0.<br><br>Property behavior:<br>- *supported* if `type` is set to ”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), or ”Telnet” (15) |
| snmp_community | string | SNMP community.<br><br>Property behavior:<br>- *required* if `type` is set to ”SNMPv1 agent” or ”SNMPv2 agent” |
| snmpv3_authpassphrase | string | Фраза-пароль аутентификации.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” and `snmpv3_securitylevel` is set to ”authNoPriv” or ”authPriv” |
| snmpv3_authprotocol | integer | Протокол аутентификации.<br><br>Possible values:<br>0 - *(default)* MD5;<br>1 - SHA1;<br>2 - SHA224;<br>3 - SHA256;<br>4 - SHA384;<br>5 - SHA512.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” and `snmpv3_securitylevel` is set to ”authNoPriv” or ”authPriv” |
| snmpv3_contextname | string | Имя контекста SNMPv3.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” |
| snmpv3_privpassphrase | string | Фраза-пароль конфиденциальности.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” and `snmpv3_securitylevel` is set to ”authPriv” |
| snmpv3_privprotocol | integer | Протокол конфиденциальности.<br><br>Possible values:<br>0 - *(default)* DES;<br>1 - AES128;<br>2 - AES192;<br>3 - AES256;<br>4 - AES192C;<br>5 - AES256C.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” and `snmpv3_securitylevel` is set to ”authPriv” |

| Property | Type | Description |
|---|---|---|
| snmpv3_securitylevel | string | Уровень безопасности.<br><br>Possible values:<br>0 - noAuthNoPriv;<br>1 - authNoPriv;<br>2 - authPriv.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” |
| snmpv3_securityname | string | Имя безопасности.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SNMPv3 agent” |
| type | integer | Тип проверки.<br><br>Possible values:<br>0 - SSH;<br>1 - LDAP;<br>2 - SMTP;<br>3 - FTP;<br>4 - HTTP;<br>5 - POP;<br>6 - NNTP;<br>7 - IMAP;<br>8 - TCP;<br>9 - Zabbix agent;<br>10 - SNMPv1 agent;<br>11 - SNMPv2 agent;<br>12 - ICMP ping;<br>13 - SNMPv3 agent;<br>14 - HTTPS;<br>15 - Telnet.<br><br>Property behavior:<br>- *required* |
| uniq | integer | Использовать ли эту проверку как критерий уникальности устройства. Для правила обнаружения может быть настроена только одна уникальная проверка.<br><br>Possible values:<br>0 - *(default)* не использовать эту проверку как критерий уникальности;<br>1 - использовать эту проверку как критерий уникальности.<br><br>Property behavior:<br>- *supported* if `type` is set to ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent”, or ”SNMPv3 agent” |
| host_source | integer | Источник имени узла сети.<br><br>Possible values:<br>1 - *(default)* DNS;<br>2 - IP;<br>3 - значение обнаружения этой проверки. |
| name_source | integer | Источник видимого имени.<br><br>Possible values:<br>0 - *(default)* не указано;<br>1 - DNS;<br>2 - IP;<br>3 - значение обнаружения этой проверки. |

| Property | Type | Description |
|---|---|---|
| allow_redirect | integer | Разрешить ситуацию, когда цель ICMP ping отвечает с другого IP-адреса.<br><br>Possible values:<br>0 - *(default)* считать перенаправленные ответы так, как если бы целевой узел сети был недоступен (fail);<br>1 - считать перенаправленные ответы так, как если бы целевой узел сети был доступен (success).<br><br>Property behavior:<br>- *supported* if `type` is set to "ICMP ping" |

**dcheck.get**

Описание

`integer/array dcheck.get(parameters)`

Метод позволяет получить проверки обнаружения по заданным параметрам.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dcheckids | ID/array | Возвращать только проверки обнаружения с указанными ID. |
| druleids | ID/array | Возвращать только проверки обнаружения, которые принадлежат указанным правилам обнаружения. |
| dserviceids | ID/array | Возвращать только проверки обнаружения, которые обнаружили указанные обнаруженные сервисы. |
| selectDRules | query | Возвращать правила обнаружения, связанные с проверками обнаружения. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `dcheckid`, `druleid`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение проверок обнаружения для правила обнаружения

Получить все проверки обнаружения, используемые правилом обнаружения ”6”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "dcheck.get",
    "params": {
        "output": "extend",
        "dcheckids": "6"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Источник

CDCheck::get() в *ui/include/classes/api/services/CDCheck.php*.

### Прокси

Этот класс предназначен для работы с прокси.

Ссылки на объекты:

- Proxy

Доступные методы:

- proxy.create - создать новые прокси
- proxy.delete - удалить прокси
- proxy.get - получить прокси
- proxy.update - обновить прокси

### Объект proxy

Следующие объекты напрямую связаны с API `proxy`.

Прокси

Объект прокси имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| proxyid | ID | ID прокси.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |
| name | string | Имя прокси.<br><br>Property behavior:<br>- *required* for create operations |
| proxy_groupid | ID | ID группы прокси.<br><br>0 - прокси не назначен ни одной группе прокси. |
| local_address | string | Адрес для активных агентов. IP-адрес или DNS-имя для подключения.<br><br>Property behavior:<br>- *required* if `proxy_groupid` is not 0 |
| local_port | string | Номер локального порта прокси для подключения.<br><br>Поддерживаются пользовательские макросы.<br><br>По умолчанию: 10051.<br><br>Property behavior:<br>- *supported* if `proxy_groupid` is not 0 |
| operating_mode | integer | Тип прокси.<br><br>Возможные значения:<br>0 - активный прокси;<br>1 - пассивный прокси.<br><br>Property behavior:<br>- *required* for create operations |
| description | text | Описание прокси. |
| lastaccess | timestamp | Время, когда прокси в последний раз подключался к серверу.<br><br>Property behavior:<br>- *read-only* |
| address | string | IP-адрес или DNS-имя для подключения.<br><br>Поддерживаются пользовательские макросы.<br><br>Property behavior:<br>- *required* if the Zabbix proxy operating mode is passive |
| port | string | Номер порта для подключения.<br><br>Поддерживаются пользовательские макросы.<br><br>По умолчанию: 10051.<br><br>Property behavior:<br>- *supported* if the Zabbix proxy operating mode is passive |
| allowed_addresses | string | IP-адреса или DNS-имена активного прокси Zabbix, разделенные запятыми. |

| Property | Type | Description |
|---|---|---|
| tls_connect | integer | Подключения к прокси.<br><br>Возможные значения:<br>1 - *(default)* Без шифрования;<br>2 - PSK;<br>4 - сертификат. |
| tls_accept | integer | Подключения от прокси.<br><br>Возможные значения битовой маски:<br>1 - *(default)* Без шифрования;<br>2 - PSK;<br>4 - сертификат.<br><br>Это поле является битовой маской; допустима любая сумма возможных значений битовой маски (например, 6 для PSK и сертификата). |
| tls_issuer | string | Издатель сертификата. |
| tls_subject | string | Тема сертификата. |
| tls_psk_identity | string | Идентификатор PSK; должен быть связан только с одним PSK (среди авторегистрации, узлов сети и прокси).<br><br>Не включайте конфиденциальную информацию в идентификатор PSK, так как он передается по сети без шифрования, чтобы получатель знал, какой PSK использовать.<br><br>Property behavior:<br>- *write-only*<br>- *required* if `tls_connect` is set to ”PSK”, or `tls_accept` contains the ”PSK” bit |
| tls_psk | string | Предварительно общий ключ (PSK); должен содержать не менее 32 шестнадцатеричных цифр.<br><br>Property behavior:<br>- *write-only*<br>- *required* if `tls_connect` is set to ”PSK”, or `tls_accept` contains the ”PSK” bit |
| custom_timeouts | integer | Следует ли переопределять глобальные тайм-ауты элементов данных на уровне прокси.<br><br>Возможные значения:<br>0 - *(default)* использовать глобальные настройки;<br>1 - переопределить тайм-ауты. |
| timeout_zabbix_agent | string | Тратить на обработку не более `timeout_*` секунд.<br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m). Также принимает пользовательские макросы.<br><br>Диапазон возможных значений: 1-600s.<br><br>По умолчанию: ””.<br><br>Property behavior:<br>- *required* if `custom_timeouts` is set to 1. |
| timeout_simple_check | | |
| timeout_snmp_agent | | |
| timeout_external_check | | |
| timeout_db_monitor | | |
| timeout_http_agent | | |
| timeout_ssh_agent | | |
| timeout_telnet_agent | | |
| timeout_script | | |

| Property | Type | Description |
|---|---|---|
| timeout_browser | | |
| version | integer | Версия прокси.<br><br>Трехкомпонентный номер версии Zabbix, где для каждой части используются две десятичные цифры, например 60401 для версии 6.4.1, 70002 для версии 7.0.2 и т. д.<br>0 - неизвестная версия прокси.<br><br>Property behavior:<br>- *read-only* |
| compatibility | integer | Версия прокси по сравнению с версией сервера Zabbix.<br><br>Возможные значения:<br>0 - не определено;<br>1 - текущая версия (у прокси и сервера совпадает основная версия);<br>2 - устаревшая версия (версия прокси старше версии сервера, но частично поддерживается);<br>3 - неподдерживаемая версия (версия прокси старше версии предыдущего LTS-релиза сервера или основная версия сервера старше основной версии прокси).<br><br>Property behavior:<br>- *read-only* |
| state | integer | Состояние прокси.<br><br>Возможные значения:<br>0 - неизвестно;<br>1 - не в сети;<br>2 - в сети.<br><br>Property behavior:<br>- *read-only* |

**proxy.create**

Описание

`object proxy.create(  /  proxies)`

Этот метод позволяет создавать новые прокси.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Прокси для создания.

Помимо стандартных свойств прокси, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hosts | array | Узлы сети, которые должны отслеживаться прокси.<br>Если узел сети уже отслеживается другим прокси, он будет переназначен текущему прокси.<br><br>У узлов сети должно быть определено только свойство `hostid`. |

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы созданных прокси в свойстве proxyids. Порядок возвращаемых идентификаторов соответствует порядку переданных прокси.

Примеры

Создание активного прокси

Создайте активный прокси "Active proxy" и назначьте ему узел сети для мониторинга.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "hosts": [
            {
                "hostid": "10279"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10280"
        ]
    },
    "id": 1
}
```

Создание пассивного прокси

Создайте пассивный прокси "Passive proxy" и назначьте ему два узла сети для мониторинга.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "proxyids": [
            "10284"
        ]
    },
    "id": 1
}
```

Создание прокси и добавление его в группу прокси

Создайте активный прокси ”Active proxy” и добавьте его в группу прокси с ID ”1”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "proxy_groupid": "1",
        "operating_mode": "0"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Создание прокси с шифрованием на основе сертификата

Создайте активный прокси с шифрованием на основе сертификата. Прокси настроен на мониторинг узла сети ”10785”.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "2"
        ]
```

```
    },
    "id": 1
}
```

См. также

- Узел сети
- Группа прокси

Источник

CProxy::create() в *ui/include/classes/api/services/CProxy.php*.

**proxy.delete**

Описание

`object proxy.delete(    proxies)`

Этот метод позволяет удалять прокси.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(    )` ID удаляемых прокси.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных прокси в свойстве `proxyids`.

Примеры

Удаление нескольких прокси

Удаление двух прокси.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.delete",
    "params": [
        "10286",
        "10285"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10286",
            "10285"
        ]
    },
    "id": 1
}
```

Источник

CProxy::delete() в *ui/include/classes/api/services/CProxy.php*.

**proxy.get**

Описание

`integer/array proxy.get( )`

Этот метод позволяет получать прокси в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие желаемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| proxyids | ID/array | Возвращать только прокси с указанными ID. |
| proxy_groupids | ID/array | Возвращать только прокси, которые принадлежат указанным группам прокси. |
| selectAssignedHosts | query | Возвращать свойство `assignedHosts` с узлами сети, назначенными прокси.<br><br>Поддерживает `count`. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, контролируемыми прокси.<br><br>Поддерживает `count`. |
| selectProxyGroup | query | Возвращать свойство `proxyGroup` с объектом группы прокси. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `proxyid`, `name`, `operating_mode`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получить все прокси

Получить все настроенные прокси и их интерфейсы.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.get",
    "params": {
```

```
        "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Узел сети
- Группа прокси

Источник

CProxy::get() в *ui/include/classes/api/services/CProxy.php*.

**proxy.update**

Описание

`object proxy.update(  /  proxies)`

Этот метод позволяет обновлять существующие прокси.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства прокси, которые необходимо обновить.

Для каждого прокси должно быть определено свойство `proxyid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам прокси метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hosts | array | Узлы сети, которые должны отслеживаться прокси.<br>Если узел сети уже отслеживается другим прокси, он будет переназначен текущему прокси.<br><br>Для узлов сети должно быть определено только свойство `hostid`. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных прокси в свойстве `proxyids`.

Примеры

Изменение узлов сети, мониторинг которых выполняет прокси

Обновите прокси, чтобы он выполнял мониторинг двух указанных узлов сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "hosts": [
            {
```

```
                "hostid": "10294"
            },
            {
                "hostid": "10295"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
    },
    "id": 1
}
```

Изменение статуса прокси

Измените прокси на активный прокси и переименуйте его в "Active proxy".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "name": "Active proxy",
        "operating_mode": "0"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
    },
    "id": 1
}
```

Добавление прокси в группу прокси

Обновите прокси с ID "5" и добавьте его в группу прокси с ID "1".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "proxyid": "5",
        "proxy_groupid": "1",
        "local_address": "127.0.0.1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

См. также

- Узел сети
- Группа прокси

Источник

CProxy::update() в *ui/include/classes/api/services/CProxy.php*.

#### Прототип графика

Этот класс предназначен для работы с прототипами графиков.

Ссылки на объекты:

- Прототип графика

Доступные методы:

- graphprototype.create - создать новые прототипы графиков
- graphprototype.delete - удалить прототипы графиков
- graphprototype.get - получить прототипы графиков
- graphprototype.update - обновить прототипы графиков

#### Объект прототипа графика

Следующие объекты напрямую связаны с API `graphprototype`.

Прототип графика

Объект прототипа графика имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| graphid | ID | ID прототипа графика.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| height | integer | Высота прототипа графика в пикселях.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Имя прототипа графика.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| width | integer | Ширина прототипа графика в пикселях.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| flags | integer | Происхождение прототипа графика.<br><br>Возможные значения:<br>2 - прототип графика;<br>6 - обнаруженный прототип графика<br><br>Поведение свойства:<br>- *только для чтения* |
| graphtype | integer | Тип макета прототипа графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* обычный;<br>1 - составной;<br>2 - круговой;<br>3 - развернутый. |
| percent_left | float | Левый процентиль.<br><br>По умолчанию: 0. |
| percent_right | float | Правый процентиль.<br><br>По умолчанию: 0. |
| show_3d | integer | Показывать ли обнаруженные круговые и развернутые графики в 3D.<br><br>Возможные значения:<br>0 - *(по умолчанию)* показывать в 2D;<br>1 - показывать в 3D. |
| show_legend | integer | Показывать ли легенду на обнаруженном графике.<br><br>Возможные значения:<br>0 - скрыть;<br>1 - *(по умолчанию)* показать. |
| show_work_period | integer | Показывать ли рабочее время на обнаруженном графике.<br><br>Возможные значения:<br>0 - скрыть;<br>1 - *(по умолчанию)* показать. |
| templateid | ID | ID родительского прототипа графика шаблона.<br><br>Поведение свойства:<br>- *только для чтения* |
| yaxismax | float | Фиксированное максимальное значение для оси Y. |
| yaxismin | float | Фиксированное минимальное значение для оси Y. |
| ymax_itemid | ID | ID элемента данных, который используется как максимальное значение для оси Y.<br><br>Если у пользователя нет доступа к указанному элементу данных, график отображается так, как если бы `ymax_type` был установлен в ”calculated”. |
| ymax_type | integer | Метод расчета максимального значения для оси Y.<br><br>Возможные значения:<br>0 - *(по умолчанию)* calculated;<br>1 - fixed;<br>2 - item. |
| ymin_itemid | ID | ID элемента данных, который используется как минимальное значение для оси Y.<br><br>Если у пользователя нет доступа к указанному элементу данных, график отображается так, как если бы `ymin_type` был установлен в ”calculated”. |

| Property | Type | Description |
|---|---|---|
| ymin_type | integer | Метод расчета минимального значения для оси Y.<br><br>Возможные значения:<br>0 - *(по умолчанию)* calculated;<br>1 - fixed;<br>2 - item. |
| discover | integer | Статус обнаружения прототипа графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* новые графики будут обнаруживаться;<br>1 - новые графики не будут обнаруживаться, а существующие графики будут помечены как потерянные. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных прототипов графиков с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается*, если прототип графика принадлежит шаблону |

**graphprototype.create**

Описание

`object graphprototype.create( / graphPrototypes)`

Этот метод позволяет создавать новые прототипы графиков.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Прототипы графиков для создания.

Помимо стандартных свойств прототипа графика, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| gitems | array | Элементы графика, которые будут созданы для прототипов графиков. Элементы графика могут ссылаться как на элементы данных, так и на прототипы элементов данных, но должен присутствовать как минимум один прототип элемента данных.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID созданных прототипов графиков, указанных в свойстве `graphids`. Порядок возвращаемых ID совпадает с порядком переданных прототипов графиков.

Примеры

Создание прототипа графика

Создайте прототип графика с двумя элементами данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.create",
```

```
    "params": {
        "name": "                    {#FSNAME}",
        "width": 900,
        "height": 200,
        "gitems": [
            {
                "itemid": "22828",
                "color": "00AA00"
            },
            {
                "itemid": "22829",
                "color": "3333FF"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

См. также

- Элемент данных графика

Источник

CGraphPrototype::create() в *ui/include/classes/api/services/CGraphPrototype.php*.

**graphprototype.delete**

Описание

`object graphprototype.delete(    graphPrototypeIds)`

Этот метод позволяет удалять прототипы графиков.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых прототипов графиков.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных прототипов графиков, указанных в свойстве `graphids`.

Примеры

Удаление нескольких прототипов графиков

Удаление двух прототипов графиков.

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "graphprototype.delete",
    "params": [
        "652",
        "653"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Источник

CGraphPrototype::delete() в *ui/include/classes/api/services/CGraphPrototype.php*.

**graphprototype.get**

Описание

`integer/array graphprototype.get(  )`

Этот метод позволяет получать прототипы графиков в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| discoveryids | ID/array | Возвращать только прототипы графиков, которые принадлежат указанным правилам обнаружения. |
| graphids | ID/array | Возвращать только прототипы графиков с указанными ID. |
| groupids | ID/array | Возвращать только прототипы графиков, которые принадлежат узлам сети или шаблонам в указанных группах узлов сети или группах шаблонов. |
| hostids | ID/array | Возвращать только прототипы графиков, которые принадлежат указанным узлам сети. |
| inherited | boolean | Если установлено значение `true`, возвращать только прототипы графиков, унаследованные от шаблона. |
| itemids | ID/array | Возвращать только прототипы графиков, которые содержат указанные прототипы элементов данных. |
| templated | boolean | Если установлено значение `true`, возвращать только прототипы графиков, которые принадлежат шаблонам. |
| templateids | ID/array | Возвращать только прототипы графиков, которые принадлежат указанным шаблонам. |

| Parameter | Type | Description |
|---|---|---|
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения прототипа графика. Объект обнаружения прототипа графика связывает обнаруженный прототип графика с прототипом графика, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`parent_graphid` - `(ID)` ID прототипа графика, на основе которого был создан график;<br>`status` - `(int)` статус обнаружения графика:<br>0 - *(по умолчанию)* прототип графика обнаружен,<br>1 - прототип графика больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда будет удален прототип графика, который больше не обнаружен. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с правилом LLD, к которому принадлежит прототип графика. |
| selectDiscoveryRulePrototype | query | Возвращать свойство `discoveryRulePrototype` с родительским прототипом правила LLD, к которому принадлежит прототип графика. |
| selectGraphItems | query | Возвращать свойство `gitems` с элементами графика, используемыми в прототипе графика. |
| selectHostGroups | query | Возвращать свойство `hostgroups` с группами узлов сети, к которым принадлежит прототип графика. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, к которым принадлежит прототип графика. |
| selectItems | query | Возвращать свойство `items` с элементами данных и прототипами элементов данных, используемыми в прототипе графика. |
| selectTemplateGroups | query | Возвращать свойство `templategroups` с группами шаблонов, к которым принадлежит прототип графика. |
| selectTemplates | query | Возвращать свойство `templates` с шаблонами, к которым принадлежит прототип графика. |
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений для сопоставления.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит прототип графика;<br>`hostid` - ID узла сети, к которому принадлежит прототип графика. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `graphid`, `name`, `graphtype`, `discovered`. |
| countOutput | boolean | Эти параметры описаны в комментарии к справочнику. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;

- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение прототипов графиков из правила LLD

Получить все прототипы графиков из правила LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "graphid": "1017",
            "name": "Disk space usage {#FSNAME}",
            "width": "600",
            "height": "340",
            "yaxismin": "0.0000",
            "yaxismax": "0.0000",
            "templateid": "442",
            "show_work_period": "0",
            "show_triggers": "0",
            "graphtype": "2",
            "show_legend": "1",
            "show_3d": "1",
            "percent_left": "0.0000",
            "percent_right": "0.0000",
            "ymin_type": "0",
            "ymax_type": "0",
            "ymin_itemid": "0",
            "ymax_itemid": "0",
            "flags": "2",
            "discover": "0"
        }
    ],
    "id": 1
}
```

См. также

- Правило обнаружения
- Элемент графика
- Элемент данных
- Узел сети
- Группа узлов сети
- Шаблон
- Группа шаблонов

Источник

CGraphPrototype::get() в *ui/include/classes/api/services/CGraphPrototype.php*.

**graphprototype.update**

Описание

`object graphprototype.update( / graphPrototypes)`

Этот метод позволяет обновлять существующие прототипы графиков.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства прототипа графика, которые необходимо обновить.

Для каждого прототипа графика должно быть определено свойство `graphid`; все остальные свойства являются необязательными. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств прототипа графика, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| gitems | array | Элементы графика для замены существующих элементов графика. Если для элемента графика определено свойство `gitemid`, он будет обновлен, в противном случае будет создан новый элемент графика. |

Возвращаемые значения

`(object)` Возвращает объект, который содержит ID обновленных прототипов графиков, указанных в свойстве `graphids`.

Примеры

Изменение размера прототипа графика

Измените размер прототипа графика на 1100 x 400 пикселей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.update",
    "params": {
        "graphid": "439",
        "width": 1100,
        "height": 400
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Источник

CGraphPrototype::update() в *ui/include/classes/api/services/CGraphPrototype.php*.

**Прототип правила LLD**

Этот класс предназначен для работы с прототипами правил низкоуровневого обнаружения.

Ссылки на объекты:

- Прототип правила LLD
  - HTTP header
  - HTTP query field
- Фильтр прототипа правила LLD
  - Условие фильтра прототипа правила LLD
- Путь макроса LLD-прототипа
- Предобработка прототипа правила LLD
- Переопределения прототипа правила LLD
  - Фильтр переопределения прототипа правила LLD
    - Условие фильтра переопределения прототипа правила LLD
  - Операция переопределения прототипа правила LLD
    - Статус операции переопределения прототипа правила LLD
    - Обнаружение операции переопределения прототипа правила LLD
    - Период операции переопределения прототипа правила LLD
    - История операции переопределения прототипа правила LLD
    - Тренды операции переопределения прототипа правила LLD
    - Серьезность операции переопределения прототипа правила LLD
    - Тег операции переопределения прототипа правила LLD
    - Шаблон операции переопределения прототипа правила LLD
    - Инвентаризация операции переопределения прототипа правила LLD

Доступные методы:

- discoveryruleprototype.create - создать новые прототипы правил LLD
- discoveryruleprototype.delete - удалить прототипы правил LLD
- discoveryruleprototype.get - получить прототипы правил LLD
- discoveryruleprototype.update - обновить прототипы правил LLD

**Объект прототипа правила LLD**

Следующие объекты напрямую связаны с API `discoveryruleprototype`.

Прототип правила LLD

Объект прототипа правила обнаружения низкого уровня имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| itemid | ID | ID прототипа правила LLD.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| ruleid | ID | ID родительского правила LLD/прототипа правила LLD.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| delay | string | Интервал обновления прототипа правила LLD.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) и, при необходимости, один или несколько пользовательских интервалов, разделенных точкой с запятой. Пользовательские интервалы могут быть сочетанием гибких и расписанных интервалов.<br><br>Принимает пользовательские макросы. Если используется макрос, значение должно быть одним макросом. Несколько макросов или макросы, смешанные с текстом, не поддерживаются. Гибкие интервалы можно записывать как два макроса, разделенные косой чертой (например, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`).<br><br>Пример:<br>`1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr`<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”Zabbix agent” (0), ”Simple check” (3), ”Zabbix internal” (5), ”External check” (10), ”Database monitor” (11), ”IPMI agent” (12), ”SSH agent” (13), ”TELNET agent” (14), ”JMX agent” (16), ”HTTP agent” (19), ”SNMP agent” (20), ”Script” (21), ”Browser” (22), или если `type` задан как ”Zabbix agent (active)” (7) и `key_` не содержит ”mqtt.get” |
| hostid | ID | ID узла сети, которому принадлежит прототип правила LLD.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |
| flags | integer | Происхождение прототипа правила LLD.<br><br>Возможные значения:<br>3 - прототип правила обнаружения низкого уровня;<br>7 - обнаруженный прототип правила обнаружения низкого уровня.<br><br>Поведение свойства:<br>- *только для чтения* |
| interfaceid | ID | ID интерфейса узла сети прототипа правила LLD.<br><br>Поведение свойства:<br>- *обязательно* если прототип правила LLD принадлежит узлу сети и `type` задан как ”Zabbix agent”, ”IPMI agent”, ”JMX agent” или ”SNMP agent”<br>- *поддерживается* если прототип правила LLD принадлежит узлу сети и `type` задан как ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” или ”HTTP agent” |
| key_ | string | Ключ прототипа правила LLD. Требуется как минимум один макрос LLD.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов |
| name | string | Имя прототипа правила LLD.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов |

| Property | Type | Description |
|---|---|---|
| type | integer | Тип прототипа правила LLD.<br><br>Возможные значения:<br>0 - Zabbix agent;<br>2 - Zabbix trapper;<br>3 - Simple check;<br>5 - Zabbix internal;<br>7 - Zabbix agent (active);<br>10 - External check;<br>11 - Database monitor;<br>12 - IPMI agent;<br>13 - SSH agent;<br>14 - TELNET agent;<br>16 - JMX agent;<br>18 - Dependent item;<br>19 - HTTP agent;<br>20 - SNMP agent;<br>21 - Script;<br>22 - Browser;<br>23 - Nested.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов |
| url | string | Строка URL.<br>Поддерживает пользовательские макросы, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |
| allow_traps | integer | Разрешить заполнять значение аналогично элементу данных типа trapper.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не разрешать принимать входящие данные;<br>1 - разрешать принимать входящие данные.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent” |
| authtype | integer | Метод аутентификации.<br><br>Возможные значения, если `type` задан как ”SSH agent”:<br>0 - *(по умолчанию)* пароль;<br>1 - публичный ключ.<br><br>Возможные значения, если `type` задан как ”HTTP agent”:<br>0 - *(по умолчанию)* нет;<br>1 - basic;<br>2 - NTLM.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”SSH agent” или ”HTTP agent”<br>- *только для чтения* для унаследованных объектов (если `type` задан как ”HTTP agent”) |
| description | string | Описание прототипа правила LLD. |

| Property | Type | Description |
|---|---|---|
| error | string | Текст ошибки, если при обновлении значения прототипа правила LLD возникли проблемы.<br><br>Поведение свойства:<br>- *только для чтения* |
| follow_redirects | integer | Следовать перенаправлениям ответа при опросе данных.<br><br>Возможные значения:<br>0 - не следовать перенаправлениям;<br>1 - *(по умолчанию)* следовать перенаправлениям.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "HTTP agent"<br>- *только для чтения* для унаследованных объектов |
| headers | array | Массив заголовков, которые будут отправлены при выполнении HTTP-запроса.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "HTTP agent"<br>- *только для чтения* для унаследованных объектов |
| http_proxy | string | Строка подключения к HTTP(S)-прокси.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "HTTP agent"<br>- *только для чтения* для унаследованных объектов |
| ipmi_sensor | string | Датчик IPMI.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как "IPMI agent" и `key_` не задан как "ipmi.get"<br>- *поддерживается* если `type` задан как "IPMI agent"<br>- *только для чтения* для унаследованных объектов |
| jmx_endpoint | string | Пользовательская строка подключения агента JMX.<br><br>По умолчанию:<br>service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "JMX agent" |
| lifetime | string | Период времени, по истечении которого элементы данных, которые больше не обнаруживаются, будут удалены. Принимает секунды, единицу времени с суффиксом или пользовательский макрос.<br><br>По умолчанию: `7d`. |
| lifetime_type | integer | Сценарий удаления потерянных ресурсов LLD.<br><br>Возможные значения:<br>0 - *(по умолчанию)* удалить после достижения порога lifetime;<br>1 - не удалять;<br>2 - удалить немедленно. |
| enabled_lifetime | string | Период времени, по истечении которого элементы данных, которые больше не обнаруживаются, будут отключены. Принимает секунды, единицу времени с суффиксом или пользовательский макрос.<br><br>По умолчанию: 0. |

| Property | Type | Description |
|---|---|---|
| enabled_lifetime_type | integer | Сценарий отключения потерянных ресурсов LLD.<br><br>Возможные значения:<br>0 - отключить после достижения порога lifetime;<br>1 - не отключать;<br>2 - *(по умолчанию)* отключить немедленно. |
| master_itemid | ID | ID мастер-элемента данных.<br>Правило обнаружения не может быть мастер-элементом данных для другого правила обнаружения.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”Dependent item”<br>- *только для чтения* для унаследованных объектов |
| output_format | integer | Следует ли преобразовывать ответ в JSON.<br><br>Возможные значения:<br>0 - *(по умолчанию)* хранить как есть;<br>1 - преобразовать в JSON.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |
| params | string | Дополнительные параметры в зависимости от типа прототипа правила LLD:<br>- выполняемый скрипт для правил LLD SSH и Telnet;<br>- SQL-запрос для правил LLD типа database monitor;<br>- формула для вычисляемых правил LLD;<br>- скрипт для правил LLD типа script и browser.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Script” или ”Browser”<br>- *только для чтения* для унаследованных объектов (если `type` задан как ”Script” или ”Browser”) |
| parameters | object/array | Дополнительные параметры, если `type` задан как ”Script” или ”Browser”.<br>Массив объектов со свойствами `name` и `value`, где `name` должен быть уникальным.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”Script” или ”Browser”<br>- *только для чтения* для унаследованных объектов |
| password | string | Пароль для аутентификации.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”JMX agent” и задан `username`<br>- *поддерживается* если `type` задан как ”Simple check”, ”Database monitor”, ”SSH agent”, ”TELNET agent” или ”HTTP agent”<br>- *только для чтения* для унаследованных объектов (если `type` задан как ”HTTP agent”) |
| post_type | integer | Тип тела POST-данных, хранящихся в свойстве `posts`.<br><br>Возможные значения:<br>0 - *(по умолчанию)* необработанные данные;<br>2 - данные JSON;<br>3 - данные XML.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |

| Property | Type | Description |
|---|---|---|
| posts | string | Данные тела HTTP(S)-запроса.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”HTTP agent” и `post_type` задан как ”JSON data” или ”XML data”<br>- *поддерживается* если `type` задан как ”HTTP agent” и `post_type` задан как ”Raw data”<br>- *только для чтения* для унаследованных объектов |
| privatekey | string | Имя файла закрытого ключа.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”SSH agent” и `authtype` задан как ”public key” |
| publickey | string | Имя файла открытого ключа.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”SSH agent” и `authtype` задан как ”public key” |
| query_fields | array | Массив полей запроса, которые будут отправлены при выполнении HTTP-запроса.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |
| request_method | integer | Тип метода запроса.<br><br>Возможные значения:<br>0 - *(по умолчанию)* GET;<br>1 - POST;<br>2 - PUT;<br>3 - HEAD.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |
| retrieve_mode | integer | Какая часть ответа должна быть сохранена.<br><br>Возможные значения, если `request_method` задан как ”GET”, ”POST” или ”PUT”:<br>0 - *(по умолчанию)* тело;<br>1 - заголовки;<br>2 - будут сохранены и тело, и заголовки.<br><br>Возможные значения, если `request_method` задан как ”HEAD”:<br>1 - заголовки.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |
| snmp_oid | string | OID SNMP.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”SNMP agent”<br>- *только для чтения* для унаследованных объектов |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |

| Property | Type | Description |
|---|---|---|
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "HTTP agent"<br>- *только для чтения* для унаследованных объектов |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "HTTP agent"<br>- *только для чтения* для унаследованных объектов |
| state | integer | Состояние прототипа правила LLD.<br><br>Возможные значения:<br>0 - *(по умолчанию)* нормальное;<br>1 - не поддерживается.<br><br>Поведение свойства:<br>- *только для чтения* |
| status | integer | Статус прототипа правила LLD.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включенный прототип правила LLD;<br>1 - отключенный прототип правила LLD. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми. Также поддерживает пользовательские макросы как часть списка, разделенного запятыми.<br><br>Пример: 200,200-{$M},{$M},200-400<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "HTTP agent"<br>- *только для чтения* для унаследованных объектов |
| templateid | ID | ID родительского шаблона прототипа правила LLD.<br><br>Поведение свойства:<br>- *только для чтения* |
| timeout | string | Тайм-аут запроса опроса данных элемента данных. Принимает секунды или единицу времени с суффиксом (например, 30s, 1m). Также принимает пользовательские макросы.<br><br>Диапазон возможных значений: 1-600s.<br><br>По умолчанию: "" - использовать настройки прокси/глобальные настройки.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "Zabbix agent" (0), "Simple check" (3) и `key_` не начинается с "`vmware.`" и "`icmpping`", "Zabbix agent (active)" (7), "External check" (10), "Database monitor" (11), "SSH agent" (13), "TELNET agent" (14), "HTTP agent" (19), "SNMP agent" (20) и `snmp_oid` начинается с "`walk[`" или "`get[`", "Script" (21), "Browser" (22)<br>- *только для чтения* для унаследованных объектов |
| trapper_hosts | string | Разрешенные узлы сети.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как "Zabbix trapper", или если `type` задан как "HTTP agent" и `allow_traps` задан как "Allow to accept incoming data" |

| Property | Type | Description |
|---|---|---|
| username | string | Имя пользователя для аутентификации.<br><br>Поведение свойства:<br>- *обязательно* если `type` задан как ”SSH agent”, ”TELNET agent”, или если `type` задан как ”JMX agent” и задан `password`<br>- *поддерживается* если `type` задан как ”Simple check”, ”Database monitor” или ”HTTP agent”<br>- *только для чтения* для унаследованных объектов (если `type` задан как ”HTTP agent”) |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных прототипов правил LLD с уже существующими. Автоматически генерируется, если не задан.<br><br>Поведение свойства:<br>- *поддерживается* если прототип правила LLD принадлежит шаблону |
| verify_host | integer | Следует ли проверять, что имя узла для подключения совпадает с именем в сертификате узла.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не проверять;<br>1 - проверять.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |
| verify_peer | integer | Следует ли проверять, что сертификат узла является подлинным.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не проверять;<br>1 - проверять.<br><br>Поведение свойства:<br>- *поддерживается* если `type` задан как ”HTTP agent”<br>- *только для чтения* для унаследованных объектов |

HTTP-заголовок

Объект заголовка имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя HTTP-заголовка.<br>Поведение свойства:<br>- *required* |
| value | string | Значение заголовка.<br>Поведение свойства:<br>- *required* |

Поле HTTP-запроса

Объект поля запроса определяет имя и значение, которые используются для указания параметра URL. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя параметра.<br>Property behavior:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| value | string | Значение параметра.<br>Property behavior:<br>- *required* |

Фильтр прототипа правила LLD

Объект фильтра прототипа правила LLD определяет набор условий, которые можно использовать для фильтрации обнаруженных объектов. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| conditions | object/array | Набор условий фильтра, используемых для фильтрации результатов. Условия будут отсортированы в порядке их размещения в формуле.<br><br>Поведение свойства:<br>- *required* |
| evaltype | integer | Метод оценки условий фильтра.<br><br>Возможные значения:<br>0 - And/Or;<br>1 - And;<br>2 - Or;<br>3 - Custom expression.<br><br>Поведение свойства:<br>- *required* |
| eval_formula | string | Сгенерированное выражение, которое будет использоваться для оценки условий фильтра. Выражение содержит идентификаторы, которые ссылаются на конкретные условия фильтра по их `formulaid`. Значение `eval_formula` равно значению `formula` для фильтров с пользовательским выражением.<br><br>Поведение свойства:<br>- *read-only* |
| formula | string | Определенное пользователем выражение, которое будет использоваться для оценки условий фильтров с пользовательским выражением. Выражение должно содержать идентификаторы, которые ссылаются на конкретные условия фильтра по их `formulaid`. Идентификаторы, используемые в выражении, должны точно совпадать с теми, которые определены в условиях фильтра: ни одно условие не может остаться неиспользованным или опущенным.<br><br>Поведение свойства:<br>- *required* if `evaltype` is set to ”custom expression” |

Условие фильтра прототипа правила LLD

Объект условия фильтра прототипа правила LLD определяет отдельную проверку, выполняемую над значением макроса LLD. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| macro | string | Макрос LLD, над которым выполняется проверка.<br><br>Property behavior:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| value | string | Значение для сравнения.<br><br>Property behavior:<br>- *required* if `operator` is set to ”matches regular expression” or ”does not match regular expression” |
| formulaid | string | Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтра, но при последующем запросе будет сгенерирован заново.<br><br>Property behavior:<br>- *required* if `evaltype` of LLD rule prototype filter object is set to ”custom expression” |
| operator | integer | Оператор условия.<br><br>Possible values:<br>8 - *(default)* matches regular expression;<br>9 - does not match regular expression;<br>12 - exists;<br>13 - does not exist. |

**Note:**
Чтобы лучше понять, как использовать фильтры с различными типами выражений, см. примеры на страницах методов discoveryruleprototype.get и discoveryruleprototype.create.

Путь макроса LLD

Путь макроса LLD имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| lld_macro | string | Макрос LLD.<br><br>Поведение свойства:<br>- *required* |
| path | string | Селектор для значения, которое будет назначено соответствующему макросу.<br><br>Поведение свойства:<br>- *required* |

Прототип предварительной обработки правила LLD

Объект предварительной обработки прототипа правила LLD имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип параметра предварительной обработки.<br><br>Возможные значения:<br>5 - Регулярное выражение;<br>11 - XML XPath;<br>12 - JSONPath;<br>14 - Совпадает с регулярным выражением;<br>15 - Не совпадает с регулярным выражением;<br>16 - Проверка на ошибку в JSON;<br>17 - Проверка на ошибку в XML;<br>20 - Отбрасывать неизменившееся с heartbeat;<br>21 - JavaScript;<br>23 - Prometheus в JSON;<br>24 - CSV в JSON;<br>25 - Замена;<br>27 - XML в JSON;<br>28 - Значение SNMP walk;<br>29 - SNMP walk в JSON;<br>30 - Значение SNMP get.<br><br>Property behavior:<br>- *required* |
| params | string | Дополнительные параметры, используемые параметром предварительной обработки. Несколько параметров разделяются символом новой строки (\n).<br><br>Property behavior:<br>- *required* если `type` установлено в ”Регулярное выражение” (5), ”XML XPath” (11), ”JSONPath” (12), ”Совпадает с регулярным выражением” (14), ”Не совпадает с регулярным выражением” (15), ”Проверка на ошибку в JSON” (16), ”Проверка на ошибку в XML” (17), ”Отбрасывать неизменившееся с heartbeat” (20), ”JavaScript” (21), ”Prometheus в JSON” (23), ”CSV в JSON” (24), ”Замена” (25), ”Значение SNMP walk” (28), ”SNMP walk в JSON” (29) или ”Значение SNMP get” (30) |
| error_handler | integer | Тип действия, используемый в случае сбоя шага предварительной обработки.<br><br>Возможные значения:<br>0 - Сообщение об ошибке задается сервером Zabbix;<br>1 - Отбросить значение;<br>2 - Задать пользовательское значение;<br>3 - Задать пользовательское сообщение об ошибке.<br><br>Property behavior:<br>- *required* если `type` установлено в ”Регулярное выражение” (5), ”XML XPath” (11), ”JSONPath” (12), ”Совпадает с регулярным выражением” (14), ”Не совпадает с регулярным выражением” (15), ”Проверка на ошибку в JSON” (16), ”Проверка на ошибку в XML” (17), ”Prometheus в JSON” (23), ”CSV в JSON” (24), ”XML в JSON” (27), ”Значение SNMP walk” (28), ”SNMP walk в JSON” (29) или ”Значение SNMP get” (30) |
| error_handler_params | string | Параметры обработчика ошибок.<br><br>Property behavior:<br>- *required* если `error_handler` установлено в ”Задать пользовательское значение” или ”Задать пользовательское сообщение об ошибке” |

Для каждого типа предварительной обработки поддерживаются следующие параметры и обработчики ошибок.

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 5 | Регулярное выражение | 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 | Совпадает с регулярным выражением | pattern[1] | | | 0, 1, 2, 3 |
| 15 | Не совпадает с регулярным выражением | pattern[1] | | | 0, 1, 2, 3 |
| 16 | Проверка на ошибку в JSON | path[3] | | | 0, 1, 2, 3 |
| 17 | Проверка на ошибку в XML | path[3] | | | 0, 1, 2, 3 |
| 20 | Отбрасывать неизменившееся с heart-beat | seconds^4, 5^ | | | |
| 21 | JavaScript | script[2] | | | |
| 23 | Prometheus в JSON | pattern^5, 6^ | | | 0, 1, 2, 3 |
| 24 | CSV в JSON | character[2] | character[2] | 0,1 | 0, 1, 2, 3 |
| 25 | Замена | search string[2] | replacement[2] | | |
| 27 | XML в JSON | | | | 0, 1, 2, 3 |
| 28 | Значение SNMP walk | OID[2] | Format:<br>0 - Без изменений<br>1 - UTF-8 из Hex-STRING<br>2 - MAC из Hex-STRING<br>3 - Integer из BITS | | 0, 1, 2, 3 |
| 29 | SNMP walk в JSON[7] | Field name[2] | OID prefix[2] | Format:<br>0 - Без изменений<br>1 - UTF-8 из Hex-STRING<br>2 - MAC из Hex-STRING<br>3 - Integer из BITS | 0, 1, 2, 3 |

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 30 | Значение SNMP get | Format:<br>1 - UTF-8 из Hex-STRING<br>2 - MAC из Hex-STRING<br>3 - Integer из BITS | | | 0, 1, 2, 3 |

[1] регулярное выражение<br> [2] строка<br> [3] JSONPath или XML XPath<br> [4] положительное целое число (с поддержкой суффиксов времени, например 30s, 1m, 2h, 1d)<br> [5] пользовательский макрос<br> [6] шаблон Prometheus в синтаксисе: `<metric name>{<label name>="<label value>", ...} == <value>`. Каждый компонент шаблона Prometheus (метрика, имя метки, значение метки и значение метрики) может быть пользовательским макросом.<br> [7] Поддерживает несколько записей "Field name,OID prefix,Format", разделенных символом новой строки.

Переопределения прототипов правил LLD

Объект переопределений прототипов правил LLD определяет набор правил (фильтры, условия и операции), которые используются для переопределения свойств различных объектов-прототипов. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Уникальное имя переопределения.<br><br>Property behavior:<br>- *required* |
| step | integer | Уникальный порядковый номер переопределения.<br><br>Property behavior:<br>- *required* |
| stop | integer | Останавливать обработку следующих переопределений, если есть совпадение.<br><br>Possible values:<br>0 - *(default)* don't stop processing overrides;<br>1 - stop processing overrides if filter matches. |
| filter | object | Фильтр переопределения. |
| operations | object/array | Операции переопределения. |

Фильтр переопределения прототипа правила LLD

Объект фильтра переопределения прототипа правила LLD определяет набор условий, при совпадении которых с обнаруженным объектом применяется переопределение. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| conditions | object/array | Набор условий фильтра переопределения, используемых для сопоставления обнаруженных объектов. Условия будут отсортированы в порядке их размещения в формуле.<br><br>Поведение свойства:<br>- *required* |
| evaltype | integer | Метод оценки условий фильтра переопределения.<br><br>Возможные значения:<br>0 - And/Or;<br>1 - And;<br>2 - Or;<br>3 - Custom expression.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| eval_formula | string | Сгенерированное выражение, которое будет использоваться для оценки условий фильтра переопределения. Выражение содержит идентификаторы, которые ссылаются на конкретные условия фильтра переопределения по их `formulaid`. Значение `eval_formula` равно значению `formula` для фильтров с пользовательским выражением.<br><br>Поведение свойства:<br>- *read-only* |
| formula | string | Определяемое пользователем выражение, используемое для оценки условий фильтров переопределения с пользовательским выражением. Выражение должно содержать идентификаторы, которые ссылаются на конкретные условия фильтра переопределения по их `formulaid`. Идентификаторы, используемые в выражении, должны точно соответствовать тем, которые определены в условиях фильтра переопределения: ни одно условие не может остаться неиспользованным или быть пропущенным.<br><br>Поведение свойства:<br>- *required* if `evaltype` is set to ”custom expression” |

Условие фильтра переопределения прототипа правила LLD

Объект условия фильтра переопределения прототипа правила LLD определяет отдельную проверку, выполняемую для значения макроса LLD. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| macro | string | Макрос LLD, для которого выполняется проверка.<br><br>Property behavior:<br>- *required* |
| value | string | Значение для сравнения.<br><br>Property behavior:<br>- *required* if `operator` is set to ”matches regular expression” or ”does not match regular expression” |
| formulaid | string | Произвольный уникальный ID, который используется для ссылки на условие из пользовательского выражения. Может содержать только заглавные буквы. ID должен быть задан пользователем при изменении условий фильтрации, но будет сгенерирован заново при последующем запросе.<br><br>Property behavior:<br>- *required* if `evaltype` of LLD rule prototype override filter object is set to ”custom expression” |

| Property | Type | Description |
|---|---|---|
| operator | integer | Оператор условия.<br><br>Possible values:<br>8 - *(default)* matches regular expression;<br>9 - does not match regular expression;<br>12 - exists;<br>13 - does not exist. |

Переопределение операции прототипа правила LLD

Переопределение операции прототипа правила LLD представляет собой комбинацию условий и действий, выполняемых над объектом прототипа. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| operationobject | integer | Тип обнаруженного объекта, для которого выполняется действие.<br><br>Возможные значения:<br>0 - прототип элемента данных;<br>1 - прототип триггера;<br>2 - прототип графика;<br>3 - прототип узла сети.<br><br>Поведение свойства:<br>- *required* |
| operator | integer | Оператор условия переопределения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* равно;<br>1 - не равно;<br>2 - содержит;<br>3 - не содержит;<br>8 - соответствует;<br>9 - не соответствует. |
| value | string | Шаблон для сопоставления с именем элемента данных, триггера, графика или прототипа узла сети в зависимости от выбранного объекта. |
| opstatus | object | Объект переопределения статуса операции для объектов прототипа элемента данных, триггера и узла сети. |
| opdiscover | object | Объект переопределения статуса обнаружения операции (для всех типов объектов). |
| opperiod | object | Объект переопределения периода операции (интервала обновления) для объекта прототипа элемента данных. |
| ophistory | object | Объект переопределения истории операции для объекта прототипа элемента данных. |
| optrends | object | Объект переопределения трендов операции для объекта прототипа элемента данных. |

| Property | Type | Description |
|---|---|---|
| opseverity | object | Объект переопределения серьезности операции для объекта прототипа триггера. |
| optag | object/array | Объект переопределения тега операции для объектов прототипа триггера и узла сети. |
| optemplate | object/array | Объект переопределения шаблона операции для объекта прототипа узла сети. |
| opinventory | object | Объект переопределения инвентаря операции для объекта прототипа узла сети. |

Статус операции переопределения прототипа правила LLD

Статус операции переопределения прототипа правила LLD, который задается для обнаруженного объекта. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| status | integer | Переопределяет статус для выбранного объекта.<br><br>Возможные значения:<br>0 - Создать включенным;<br>1 - Создать отключенным.<br><br>Поведение свойства:<br>- *required* |

Переопределение операции обнаружения прототипа правила LLD

Переопределение операции обнаружения прототипа правила LLD для статуса обнаружения, который установлен для обнаруженного объекта. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| discover | integer | Переопределить статус обнаружения для выбранного объекта.<br><br>Возможные значения:<br>0 - Да, продолжить обнаружение объектов;<br>1 - Нет, новые объекты не будут обнаруживаться, а существующие будут помечены как потерянные.<br><br>Property behavior:<br>- *required* |

Переопределение периода операции прототипа правила LLD

Переопределение периода операции прототипа правила LLD — это значение интервала обновления, которое задается для обнаруженного элемента данных. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| delay | string | Переопределяет интервал обновления прототипа элемента данных.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) и, при необходимости, один или несколько пользовательских интервалов, разделенных точкой с запятой. Пользовательские интервалы могут представлять собой сочетание гибких интервалов и интервалов по расписанию.<br><br>Принимает пользовательские макросы или макросы LLD. Если используется, значение должно быть одним макросом. Несколько макросов или макросы, смешанные с текстом, не поддерживаются. Гибкие интервалы можно записывать как два макроса, разделенные косой чертой (например, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`).<br><br>Пример:<br>`1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr`<br><br>Поведение свойства:<br>*- required* |

История операций переопределения прототипа правила LLD

Значение истории операций переопределения прототипа правила LLD, которое задается для обнаруженного элемента данных. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| history | string | Переопределяет историю прототипа элемента данных, то есть задает единицу времени, в течение которой должны храниться данные истории. Также принимает пользовательский макрос и LLD-макрос.<br><br>Property behavior:<br>*- required* |

Переопределение тенденций операции прототипа правила LLD

Значение тенденций переопределения операции прототипа правила LLD, которое задается для обнаруженного элемента данных. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| trends | string | Переопределяет тенденции прототипа элемента данных, то есть задает единицу времени, в течение которой должны храниться данные тенденций. Также принимает пользовательский макрос и LLD-макрос.<br><br>Property behavior:<br>*- required* |

Переопределение серьезности операции прототипа правила LLD

Значение серьезности переопределения операции прототипа правила LLD, которое задается для обнаруженного триггера. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| severity | integer | Переопределяет серьезность прототипа триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа.<br><br>Поведение свойства:<br>- *required* |

Тег операции переопределения прототипа правила LLD

Объект тега операции переопределения прототипа правила LLD содержит имя тега и значение, которые задаются для обнаруженного объекта. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| tag | string | Новое имя тега.<br>Поведение свойства:<br>- *required* |
| value | string | Новое значение тега. |

Шаблон операции переопределения прототипа правила LLD

Объект шаблона операции переопределения прототипа правила LLD, который связан с обнаруженным узлом сети. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| templateid | ID | Переопределяет шаблон связанных шаблонов прототипа узла сети.<br><br>Property behavior:<br>- *required* |

Инвентаризация операции переопределения прототипа правила LLD

Значение режима инвентаризации операции переопределения прототипа правила LLD, которое задается для обнаруженного узла сети. Оно имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| inventory_mode | integer | Переопределяет режим инвентаризации прототипа узла сети.<br><br>Возможные значения:<br>-1 - отключено;<br>0 - *(по умолчанию)* ручной;<br>1 - автоматический.<br><br>Поведение свойства:<br>- *required* |

**discoveryruleprototype.create**

Описание

```
object discoveryruleprototype.create(object/array lldRules)
```

Этот метод позволяет создавать новые прототипы правил LLD.

> **Note:**
> Этот метод доступен только для типов пользователей *Admin* и *Super admin*. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе User roles.

Параметры

`(object/array)` Прототипы правил LLD для создания.

В дополнение к стандартным свойствам прототипа правила LLD, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| filter | object | Фильтр прототипа правила LLD для правила LLD. |
| preprocessing | object/array | Параметры предобработки прототипа правила LLD. |
| lld_macro_paths | object/array | Параметры lld_macro_path прототипа правила LLD. |
| overrides | object/array | Параметры переопределений прототипа правила LLD. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных прототипов правил LLD в свойстве `itemids`. Порядок возвращаемых идентификаторов соответствует порядку переданных прототипов правил LLD.

Примеры

Создание прототипа правила LLD

Создайте прототип правила LLD (Тип: Nested), чтобы обнаруживать табличные пространства в экземпляре базы данных.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47252"
        ]
    },
    "id": 1
}
```

См. также

- Фильтр прототипа правила LLD
- Пути макросов LLD
- Предварительная обработка прототипа правила LLD

Исходный код

CDiscoveryRulePrototype::create() в *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

### discoveryruleprototype.delete

Описание

`object discoveryruleprototype.delete(array lldRuleIds)`

Этот метод позволяет удалять прототипы правил LLD.

> **Note:**
> Этот метод доступен только пользователям типов *Admin* и *Super admin*. Разрешения на вызов метода могут быть отозваны в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы прототипов правил LLD, которые нужно удалить.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных прототипов правил LLD в свойстве `ruleids`.

Примеры

Удаление нескольких прототипов правил LLD

Удаление двух прототипов правил LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.delete",
    "params": [
        "47252",
        "47253"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "ruleids": [
            "47252",
            "47253"
        ]
    },
    "id": 1
}
```

Источник

CDiscoveryRulePrototype::delete() в *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

### discoveryruleprototype.get

Описание

`integer/array discoveryruleprototype.get(object parameters)`

Метод позволяет получать прототипы правил LLD в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода могут быть отозваны в настройках роли пользователя. См. Роли пользователей для получения дополнительной информации.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| itemids | ID/array | Возвращать только прототипы правил LLD с указанными ID. |
| groupids | ID/array | Возвращать только прототипы правил LLD, которые принадлежат узлам сети из указанных групп. |
| hostids | ID/array | Возвращать только прототипы правил LLD, которые принадлежат указанным узлам сети. |
| inherited | boolean | Если установлено значение `true`, возвращать только прототипы правил LLD, унаследованные от шаблона. |
| interfaceids | ID/array | Возвращать только прототипы правил LLD, использующие указанные интерфейсы узлов сети. |
| monitored | boolean | Если установлено значение `true`, возвращать только включенные прототипы правил LLD, которые принадлежат контролируемым узлам сети. |
| templated | boolean | Если установлено значение `true`, возвращать только прототипы правил LLD, которые принадлежат шаблонам. |
| templateids | ID/array | Возвращать только прототипы правил LLD, которые принадлежат указанным шаблонам. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с родительским правилом LLD для прототипа правила LLD. |
| selectDiscoveryRulePrototype | query | Возвращать свойство `discoveryRulePrototype` с родительским прототипом правила LLD для прототипа правила LLD. |
| selectDiscoveryRulePrototypes | query | Возвращать свойство `discoveryRulePrototypes` с дочерними прототипами правил LLD, которые принадлежат прототипу правила LLD.<br><br>Поддерживает `count`. |
| selectFilter | query | Возвращать свойство `filter` с данными фильтра, используемого прототипом правила LLD. |
| selectGraphs | query | Возвращает свойство `graphs` с прототипами графиков, которые принадлежат прототипу правила LLD.<br><br>Поддерживает `count`. |
| selectHostPrototypes | query | Возвращать свойство `hostPrototypes` с прототипами узлов сети, которые принадлежат прототипу правила LLD.<br><br>Поддерживает `count`. |
| selectHosts | query | Возвращать свойство `hosts` с массивом узлов сети, к которым принадлежит прототип правила LLD. |
| selectItems | query | Возвращать свойство `items` с прототипами элементов данных, которые принадлежат прототипу правила LLD.<br><br>Поддерживает `count`. |
| selectTriggers | query | Возвращать свойство `triggers` с прототипами триггеров, которые принадлежат прототипу правила LLD.<br><br>Поддерживает `count`. |
| selectLLDMacroPaths | query | Возвращать свойство `lld_macro_paths` со списком LLD-макросов и путей к значениям, назначенным каждому соответствующему макросу. |
| selectPreprocessing | query | Возвращать свойство `preprocessing` с параметрами предварительной обработки прототипа правила LLD. |
| selectOverrides | query | Возвращать свойство `lld_rule_overrides` со списком фильтров, условий и операций переопределения, выполняемых над объектами прототипов. |

| Parameter | Type | Description |
|---|---|---|
| filter | object | Возвращать только те результаты, которые точно соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений для сопоставления.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит прототип правила LLD. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подзапросами.<br><br>Применяется к следующим подзапросам: `selectItems`, `selectGraphs`, `selectTriggers`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `itemid`, `name`, `key_`, `delay`, `type`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение прототипов правил обнаружения для узла сети

Получить все прототипы правил обнаружения для указанного ID узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.get",
    "params": {
        "hostids": "10084"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "47253",
            "type": "23",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "       tablespaces    {#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",
            "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
}
```


См. также

- Прототип графика
- Узел сети
- Прототип элемента данных
- Фильтр прототипа правила LLD
- Прототип триггера

Источник

CDiscoveryRulePrototype::get() в *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

**discoveryruleprototype.update**

Описание

`object discoveryruleprototype.update(object/array lldRules)`

Этот метод позволяет обновлять существующие прототипы правил LLD.

Обратите внимание, что обновление уже обнаруженных прототипов ограничено.

> **Note:**
> Этот метод доступен только для типов пользователей *Admin* и *Super admin*. Разрешения на вызов метода могут быть отозваны в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства прототипа правила LLD, которые нужно обновить.

Свойство `itemid` должно быть определено для каждого прототипа правила LLD, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам прототипа правила LLD, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| filter | object | Фильтр прототипа правила LLD для замены существующего фильтра. |
| preprocessing | object/array | Параметры предобработки прототипа правила LLD для замены существующих параметров предобработки.<br><br>Поведение параметра:<br>- *только чтение* для наследуемых объектов |
| lld_macro_paths | object/array | Параметры lld_macro_path прототипа правила LLD для замены существующих параметров lld_macro_path.<br><br>Поведение параметра:<br>- *только чтение* для наследуемых объектов |
| overrides | object/array | Параметры переопределений прототипа правила LLD для замены существующих параметров переопределений.<br><br>Поведение параметра:<br>- *только чтение* для наследуемых объектов |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных прототипов правил LLD в свойстве `itemids`.

Примеры

Обновление параметров предварительной обработки прототипа правила LLD

Обновите прототип правила LLD с помощью правила предварительной обработки JSONPath. Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.update",
    "params": {
        "itemid": "47253",
        "preprocessing": [
            {
                "type": 12,
                "params": "$.tablespaces",
                "error_handler": 1
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47253"
        ]
    },
    "id": 1
}
```

Исходный код

CDiscoveryRulePrototype::update() в *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

### Прототип триггера

Этот класс предназначен для работы с прототипами триггеров.

Ссылки на объекты:

- Прототип триггера
- Тег прототипа триггера

Доступные методы:

- triggerprototype.create - создать новые прототипы триггеров
- triggerprototype.delete - удалить прототипы триггеров
- triggerprototype.get - получить прототипы триггеров
- triggerprototype.update - обновить прототипы триггеров

### Объект прототипа триггера

Следующие объекты напрямую связаны с API `triggerprototype`.

Прототип триггера

Объект прототипа триггера имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| triggerid | ID | ID прототипа триггера.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| description | string | Имя прототипа триггера.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| expression | string | Выражение триггера.<br><br>Должно содержать как минимум один прототип элемента данных. Прототипы элементов данных с `value_type`, установленным в 5 (binary), нельзя использовать в выражениях триггера.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| event_name | string | Имя события, генерируемого триггером. |
| opdata | string | Оперативные данные. |
| comments | string | Дополнительные комментарии к прототипу триггера. |

| Property | Type | Description |
|---|---|---|
| priority | integer | Серьезность прототипа триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа. |
| status | integer | Включен или отключен прототип триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включен;<br>1 - отключен. |
| flags | integer | Происхождение прототипа триггера.<br><br>Возможные значения:<br>2 - прототип триггера;<br>6 - обнаруженный прототип триггера.<br><br>Поведение свойства:<br>- *только для чтения* |
| templateid | ID | ID родительского шаблона прототипа триггера.<br><br>Поведение свойства:<br>- *только для чтения* |
| type | integer | Может ли прототип триггера генерировать несколько событий проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не генерировать несколько событий;<br>1 - генерировать несколько событий. |
| url | string | URL, связанный с прототипом триггера. |
| url_name | string | Метка для URL, связанного с прототипом триггера. |
| recovery_mode | integer | Режим генерации события OK.<br><br>Возможные значения:<br>0 - *(по умолчанию)* выражение;<br>1 - выражение восстановления;<br>2 - нет. |
| recovery_expression | string | Выражение восстановления триггера.<br><br>Должно содержать как минимум один прототип элемента данных. Прототипы элементов данных с `value_type`, установленным в 5 (binary), нельзя использовать в выражениях восстановления триггера. |
| correlation_mode | integer | Закрытие события OK.<br><br>Возможные значения:<br>0 - *(по умолчанию)* все проблемы;<br>1 - все проблемы, если значения тегов совпадают. |
| correlation_tag | string | Тег для сопоставления. |
| manual_close | integer | Разрешить ручное закрытие.<br><br>Возможные значения:<br>0 - *(по умолчанию)* нет;<br>1 - да. |

| Property | Type | Description |
|---|---|---|
| discover | integer | Статус обнаружения прототипа триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* новые триггеры будут обнаруживаться;<br>1 - новые триггеры не будут обнаруживаться, а существующие триггеры будут помечены как потерянные. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных прототипов триггеров с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается*, если прототип триггера принадлежит шаблону |

Тег прототипа триггера

Объект тега прототипа триггера имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега прототипа триггера.<br><br>Property behavior:<br>- *required* |
| value | string | Значение тега прототипа триггера. |

**triggerprototype.create**

Описание

`object triggerprototype.create( / triggerPrototypes)`

Этот метод позволяет создавать новые прототипы триггеров.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Прототипы триггеров для создания.

Помимо стандартных свойств прототипа триггера, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dependencies | array | Триггеры и прототипы триггеров, от которых зависит прототип триггера.<br><br>У триггеров должно быть определено только свойство `triggerid`. |
| tags | array | Теги прототипа триггера. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных прототипов триггеров в свойстве `triggerids`. Порядок возвращаемых идентификаторов соответствует порядку переданных прототипов триггеров.

Примеры

Создание прототипа триггера

Создайте прототип триггера для обнаружения ситуации, когда в файловой системе остается менее 20% свободного места на диске.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.create",
    "params": {
        "description": "                    20%      {#FSNAME}",
        "expression": "last(/Zabbix server/vfs.fs.size[{#FSNAME},pfree])<20",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17372"
        ]
    },
    "id": 1
}
```

Источник

CTriggerPrototype::create() в *ui/include/classes/api/services/CTriggerPrototype.php*.

**triggerprototype.delete**

Описание

`object triggerprototype.delete(    triggerPrototypeIds)`

Этот метод позволяет удалять прототипы триггеров.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` Идентификаторы прототипов триггеров для удаления.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных прототипов триггеров в свойстве `triggerids`.

Примеры

Удаление нескольких прототипов триггеров

Удаление двух прототипов триггеров.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.delete",
```

```
    "params": [
        "12002",
        "12003"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "12002",
            "12003"
        ]
    },
    "id": 1
}
```

Источник

CTriggerPrototype::delete() в *ui/include/classes/api/services/CTriggerPrototype.php*.

### triggerprototype.get

Описание

`integer/array triggerprototype.get(  )`

Метод позволяет получать прототипы триггеров по заданным параметрам.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| active | flag | Возвращать только включенные прототипы триггеров, которые принадлежат мониторимым узлам сети. |
| discoveryids | ID/array | Возвращать только прототипы триггеров, которые принадлежат указанным правилам LLD. |
| functions | string/array | Возвращать только триггеры, которые используют указанные функции.<br><br>Список поддерживаемых функций см. на странице Поддерживаемые функции. |
| group | string | Возвращать только прототипы триггеров, которые принадлежат узлам сети или шаблонам из групп узлов сети или групп шаблонов с указанным именем. |
| groupids | ID/array | Возвращать только прототипы триггеров, которые принадлежат узлам сети или шаблонам из указанных групп узлов сети или групп шаблонов. |
| host | string | Возвращать только прототипы триггеров, которые принадлежат узлам сети с указанным именем. |
| hostids | ID/array | Возвращать только прототипы триггеров, которые принадлежат указанным узлам сети. |
| inherited | boolean | Если установлено значение `true`, возвращать только прототипы триггеров, унаследованные от шаблона. |

| Parameter | Type | Description |
|---|---|---|
| maintenance | boolean | Если установлено значение `true`, возвращать только включенные прототипы триггеров, которые принадлежат узлам сети в режиме обслуживания. |
| min_severity | integer | Возвращать только прототипы триггеров с уровнем серьезности не ниже указанного. |
| monitored | flag | Возвращать только включенные прототипы триггеров, которые принадлежат мониторимым узлам сети и содержат только включенные элементы данных. |
| templated | boolean | Если установлено значение `true`, возвращать только прототипы триггеров, которые принадлежат шаблонам. |
| templateids | ID/array | Возвращать только прототипы триггеров, которые принадлежат указанным шаблонам. |
| triggerids | ID/array | Возвращать только прототипы триггеров с указанными ID. |
| expandExpression | flag | Расширять функции и макросы в выражении триггера. |
| selectDependencies | query | Возвращать прототипы триггеров и триггеры, от которых зависит прототип триггера, в свойстве `dependencies`. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения прототипа триггера. Объект обнаружения прототипа триггера связывает обнаруженный прототип триггера с прототипом триггера, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`parent_triggerid` - `(ID)` ID прототипа триггера, из которого был создан прототип триггера;<br>`status` - `(int)` статус обнаружения прототипа триггера:<br>0 - *(по умолчанию)* прототип триггера обнаружен,<br>1 - прототип триггера больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда будет удален прототип триггера, который больше не обнаружен;<br>`ts_disable` - `(timestamp)` время, когда будет отключен прототип триггера, который больше не обнаружен;<br>`disable_source` - `(int)` признак того, был ли прототип триггера отключен правилом LLD или вручную:<br>0 - *(по умолчанию)* отключен автоматически,<br>1 - отключен правилом LLD. |
| selectDiscoveryRule | query | Возвращать правило LLD, к которому принадлежит прототип триггера, в свойстве `discoveryRule`. |
| selectDiscoveryRulePrototype | query | Возвращать свойство `discoveryRulePrototype` с родительским прототипом правила LLD, к которому принадлежит прототип триггера. |
| selectFunctions | query | Возвращать функции, используемые в прототипе триггера, в свойстве `functions`.<br><br>Объекты функций представляют функции, используемые в выражении триггера, и имеют следующие свойства:<br>`functionid` - `(ID)` ID функции;<br>`itemid` - `(ID)` ID элемента данных, используемого в функции;<br>`function` - `(string)` имя функции;<br>`parameter` - `(string)` параметр, переданный функции.<br>Параметр запроса в возвращаемой строке заменяется символом $. |
| selectHostGroups | query | Возвращать группы узлов сети, к которым принадлежит прототип триггера, в свойстве `hostgroups`. |
| selectHosts | query | Возвращать узлы сети, к которым принадлежит прототип триггера, в свойстве `hosts`. |
| selectItems | query | Возвращать элементы данных и прототипы элементов данных, используемые прототипом триггера, в свойстве `items`. |
| selectTags | query | Возвращать теги прототипа триггера в свойстве `tags`. |
| selectTemplateGroups | query | Возвращать группы шаблонов, к которым принадлежит прототип триггера, в свойстве `templategroups`. |

| Parameter | Type | Description |
|---|---|---|
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи - это имена свойств, а значения - либо одно значение, либо массив значений для сопоставления.<br><br>Не поддерживает свойства с типом данных `text` data type.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит прототип триггера;<br>`hostid` - ID узла сети, к которому принадлежит прототип триггера. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подвыборками.<br><br>Применяется к следующим подвыборкам:<br>`selectHosts` - результаты будут отсортированы по `host`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `triggerid`, `description`, `status`, `priority`, `discovered`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использовался параметр `countOutput`.

Примеры

Получение прототипов триггеров из правила LLD

Получение всех прототипов триггеров и их функций из правила LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": "extend",
        "selectFunctions": "extend",
        "discoveryids": "22450"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
```

```
            "triggerid": "13272",
            "expression": "{12598}<20",
            "description": "Free inodes is less than 20% on volume {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "6ce467d05e8745409a177799bed34bb3",
            "url_name": "",
            "functions": [
                {
                    "functionid": "12598",
                    "itemid": "22454",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        },
        {
            "triggerid": "13266",
            "expression": "{13500}<20",
            "description": "Free disk space is less than 20% on volume {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "74a1fc62bfe24b7eabe4e244c70dc384",
            "url_name": "",
            "functions": [
                {
                    "functionid": "13500",
                    "itemid": "22686",
```

```
                    "parameter": "$",
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```

Получение конкретного прототипа триггера с тегами

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "17373",
            "description": "Free disk space is less than 20% on volume {#FSNAME}",
            "tags": [
                {
                    "tag": "volume",
                    "value": "{#FSNAME}"
                },
                {
                    "tag": "type",
                    "value": "{#FSTYPE}"
                }
            ]
        }
    ],
    "id": 1
}
```

См. также

- Правило обнаружения
- Элемент данных
- Узел сети
- Группа узлов сети
- Группа шаблонов

Источник

CTriggerPrototype::get() в *ui/include/classes/api/services/CTriggerPrototype.php*.

**triggerprototype.update**

Описание

`object triggerprototype.update( / triggerPrototypes)`

Этот метод позволяет обновлять существующие прототипы триггеров.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства прототипа триггера, которые необходимо обновить.

Для каждого прототипа триггера должно быть определено свойство `triggerid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств прототипа триггера метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dependencies | array | Триггеры и прототипы триггеров, от которых зависит прототип триггера.<br><br>У триггеров должно быть определено только свойство `triggerid`. |
| tags | array | Теги прототипа триггера. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных прототипов триггеров в свойстве `triggerids`.

Примеры

Включение прототипа триггера

Чтобы включить прототип триггера, установите для его статуса значение ”0”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Замена тегов прототипа триггера

Заменяет теги для одного прототипа триггера.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
```

```
    "params": {
        "triggerid": "17373",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Источник

CTriggerPrototype::update() в *ui/include/classes/api/services/CTriggerPrototype.php*.

#### Прототип узла сети

Этот класс предназначен для работы с прототипами узлов сети.

Ссылки на объекты:

- Прототип узла сети
- Связь с группой
- Прототип группы
- Тег прототипа узла сети
- Пользовательский интерфейс
  - Сведения о пользовательском интерфейсе

Доступные методы:

- hostprototype.create - создать новые прототипы узлов сети
- hostprototype.delete - удалить прототипы узлов сети
- hostprototype.get - получить прототипы узлов сети
- hostprototype.update - обновить прототипы узлов сети

#### Объект прототипа узла сети

Следующие объекты напрямую связаны с API `hostprototype`.

Прототип узла сети

Объект прототипа узла сети имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| hostid | ID | ID прототипа узла сети.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| host | string | Техническое имя прототипа узла сети.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов |
| name | string | Отображаемое имя прототипа узла сети.<br><br>По умолчанию: значение свойства `host`.<br><br>Поведение свойства:<br>- *только для чтения* для унаследованных объектов |
| status | integer | Статус прототипа узла сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* контролируемый узел сети;<br>1 - неконтролируемый узел сети. |
| flags | integer | Происхождение прототипа узла сети.<br><br>Возможные значения:<br>2 - прототип узла сети;<br>6 - обнаруженный прототип узла сети<br><br>Поведение свойства:<br>- *только для чтения* |
| inventory_mode | integer | Режим заполнения инвентарных данных узла сети.<br><br>Возможные значения:<br>-1 - *(по умолчанию)* отключено;<br>0 - вручную;<br>1 - автоматически. |
| templateid | ID | ID родительского шаблона прототипа узла сети.<br><br>Поведение свойства:<br>- *только для чтения* |
| discover | integer | Статус обнаружения прототипа узла сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* новые узлы сети будут обнаруживаться;<br>1 - новые узлы сети не будут обнаруживаться, а существующие узлы сети будут помечены как потерянные. |
| custom_interfaces | integer | Источник пользовательских интерфейсов для узлов сети, созданных по прототипу узла сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* наследовать интерфейсы от родительского узла сети;<br>1 - использовать пользовательские интерфейсы прототипа узла сети.<br><br>Поведение свойства:<br>- *только для чтения* для унаследованных объектов |

| Property | Type | Description |
|---|---|---|
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных прототипов узлов сети с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается* если прототип узла сети принадлежит шаблону |

Связь с группой

Объект связи с группой связывает прототип узла сети с группой узлов сети. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| groupid | ID | ID группы узлов сети.<br>Поведение свойства:<br>- *required* |

Прототип группы

Объект прототипа группы определяет группу, которая будет создана для обнаруженного узла сети. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| group_prototypeid | ID | ID прототипа группы.<br><br>Property behavior:<br>- *read-only* |
| name | string | Имя прототипа группы.<br><br>Property behavior:<br>- *required* for create operations |

Тег прототипа узла сети

Объект тега прототипа узла сети имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега прототипа узла сети.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение тега прототипа узла сети. |

Пользовательский интерфейс

Пользовательские интерфейсы поддерживаются, если `custom_interfaces` объекта прототип узла сети установлено в значение ”use host prototypes custom interfaces”. Объект пользовательского интерфейса имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип интерфейса.<br><br>Возможные значения:<br>1 - Агент;<br>2 - SNMP;<br>3 - IPMI;<br>4 - JMX.<br><br>Поведение свойства:<br>- *required* |
| useip | integer | Следует ли выполнять подключение через IP.<br><br>Возможные значения:<br>0 - подключаться, используя DNS-имя узла сети;<br>1 - подключаться, используя IP-адрес узла сети.<br><br>Поведение свойства:<br>- *required* |
| ip | string | IP-адрес, используемый интерфейсом.<br>Может содержать макросы.<br><br>Поведение свойства:<br>- *required* if `useip` is set to ”connect using host IP address” |
| dns | string | DNS-имя, используемое интерфейсом.<br>Может содержать макросы.<br><br>Поведение свойства:<br>- *required* if `useip` is set to ”connect using host DNS name” |
| port | string | Номер порта, используемый интерфейсом.<br>Может содержать пользовательские макросы и макросы LLD.<br><br>Поведение свойства:<br>- *required* |
| main | integer | Используется ли интерфейс как интерфейс по умолчанию на узле сети.<br>На узле сети только один интерфейс некоторого типа может быть установлен как интерфейс по умолчанию.<br><br>Возможные значения:<br>0 - не по умолчанию;<br>1 - по умолчанию.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| details | object | Дополнительный объект для сведений о пользовательском интерфейсе.<br><br>Поведение свойства:<br>- *required* if `type` is set to ”SNMP” |

Сведения о пользовательском интерфейсе

Объект details имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| version | integer | Версия интерфейса SNMP.<br><br>Возможные значения:<br>1 - SNMPv1;<br>2 - SNMPv2c;<br>3 - SNMPv3.<br><br>Поведение свойства:<br>- *required* |
| bulk | integer | Использовать ли массовые запросы SNMP.<br><br>Возможные значения:<br>0 - не использовать массовые запросы;<br>1 - *(по умолчанию)* - использовать массовые запросы. |
| community | string | Сообщество SNMP.<br><br>Поведение свойства:<br>- *required* если `version` установлено в ”SNMPv1” или ”SNMPv2c” |
| max_repetitions | integer | Максимальное значение повторений для нативных массовых запросов SNMP (GetBulkRequest-PDU).<br>Используется только для элементов данных `discovery[]` и `walk[]` в SNMPv2 и v3.<br><br>По умолчанию: 10. |
| securityname | string | Имя безопасности SNMPv3.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” |
| securitylevel | integer | Уровень безопасности SNMPv3.<br><br>Возможные значения:<br>0 - *(по умолчанию)* - noAuthNoPriv;<br>1 - authNoPriv;<br>2 - authPriv.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” |
| authpassphrase | string | Фраза-пароль аутентификации SNMPv3.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” и `securitylevel` установлено в ”authNoPriv” или ”authPriv” |
| privpassphrase | string | Фраза-пароль конфиденциальности SNMPv3.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” и `securitylevel` установлено в ”authPriv” |

| Property | Type | Description |
|---|---|---|
| authprotocol | integer | Протокол аутентификации SNMPv3.<br><br>Возможные значения:<br>0 - *(по умолчанию)* - MD5;<br>1 - SHA1;<br>2 - SHA224;<br>3 - SHA256;<br>4 - SHA384;<br>5 - SHA512.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” и `securitylevel` установлено в ”authNoPriv” или ”authPriv” |
| privprotocol | integer | Протокол конфиденциальности SNMPv3. Используется только интерфейсами SNMPv3.<br><br>Возможные значения:<br>0 - *(по умолчанию)* - DES;<br>1 - AES128;<br>2 - AES192;<br>3 - AES256;<br>4 - AES192C;<br>5 - AES256C.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” и `securitylevel` установлено в ”authPriv” |
| contextname | string | Имя контекста SNMPv3.<br><br>Поведение свойства:<br>- *supported* если `version` установлено в ”SNMPv3” |

**hostprototype.create**

Описание

`object hostprototype.create( / hostPrototypes)`

Этот метод позволяет создавать новые прототипы узлов сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Узлы сети-прототипы для создания.

В дополнение к стандартным свойствам узла сети-прототипа метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groupLinks | array | Связи групп, которые будут созданы для узла сети-прототипа.<br><br>Поведение параметра:<br>- *required* |

| Parameter | Type | Description |
|---|---|---|
| ruleid | ID | ID правила LLD, к которому принадлежит узел сети-прототип.<br><br>Поведение параметра:<br>- *required* |
| groupPrototypes | array | Прототипы групп, которые будут созданы для узла сети-прототипа. |
| macros | object/array | Пользовательские макросы, которые будут созданы для узла сети-прототипа. |
| tags | object/array | Теги узла сети-прототипа. |
| interfaces | object/array | Пользовательские интерфейсы узла сети-прототипа. |
| templates | object/array | Шаблоны, которые будут привязаны к узлу сети-прототипу.<br><br>У шаблонов должно быть определено только свойство `templateid`. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных прототипов узлов сети, указанных в свойстве `hostids`. Порядок возвращаемых ID совпадает с порядком переданных прототипов узлов сети.

Примеры

Создание прототипа узла сети

Создайте прототип узла сети "{#VM.NAME}" на правиле LLD "23542" с прототипом группы "{#HV.NAME}", парой тегов "datacenter": "{#DATACENTER.NAME}" и пользовательским интерфейсом SNMPv2 127.0.0.1:161 с community {$SNMP_COMMUNITY}. Свяжите его с группой узлов сети "2".

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10103"
        ]
    },
    "id": 1
}
```

См. также

- Ссылка на группу
- Прототип группы
- Тег прототипа узла сети
- Пользовательский интерфейс
- Пользовательский макрос

Источник

CHostPrototype::create() в *ui/include/classes/api/services/CHostPrototype.php*.

**hostprototype.delete**

Описание

`object hostprototype.delete(    hostPrototypeIds)`

Этот метод позволяет удалять прототипы узлов сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых прототипов узлов сети

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных прототипов узлов сети, указанных в свойстве `hostids`.

Примеры

Удаление нескольких прототипов узлов сети

Удалите два прототипа узлов сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.delete",
```

```
    "params": [
        "10103",
        "10105"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10103",
            "10105"
        ]
    },
    "id": 1
}
```

Источник

CHostPrototype::delete() в *ui/include/classes/api/services/CHostPrototype.php*.

**hostprototype.get**

Описание

`integer/array hostprototype.get(          )`

Этот метод позволяет получать прототипы узлов сети в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hostids | ID/array | Возвращать только прототипы узлов сети с указанными ID. |
| discoveryids | ID/array | Возвращать только прототипы узлов сети, которые принадлежат указанным правилам LLD. |
| inherited | boolean | Если установлено значение `true`, возвращать только элементы данных, унаследованные от шаблона. |

| Parameter | Type | Description |
|---|---|---|
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения прототипа узла сети. Объект обнаружения прототипа узла сети связывает обнаруженный прототип узла сети с прототипом узла сети, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`host` - (`string`) ID узла сети;<br>`parent_hostid` - (`string`) ID прототипа узла сети, из которого был создан прототип узла сети;<br>`status` - (`int`) статус обнаружения прототипа узла сети:<br>0 - *(default)* прототип узла сети обнаружен,<br>1 - прототип узла сети больше не обнаружен;<br>`ts_delete` - (`timestamp`) время, когда прототип узла сети, который больше не обнаружен, будет удален;<br>`ts_disable` - (`timestamp`) время, когда прототип узла сети, который больше не обнаружен, будет отключен;<br>`disable_source` - (`int`) признак того, был ли прототип узла сети отключен правилом LLD или вручную:<br>0 - *(default)* отключен автоматически,<br>1 - отключен правилом LLD. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с правилом LLD, к которому принадлежит прототип узла сети. |
| selectDiscoveryRulePrototype | query | Возвращать свойство `discoveryRulePrototype` с родительским прототипом правила LLD, к которому принадлежит прототип узла сети. |
| selectInterfaces | query | Возвращать свойство `interfaces` с пользовательскими интерфейсами прототипа узла сети. |
| selectGroupLinks | query | Возвращать свойство `groupLinks` со связями групп прототипа узла сети. |
| selectGroupPrototypes | query | Возвращать свойство `groupPrototypes` с прототипами групп прототипа узла сети. |
| selectMacros | query | Возвращать свойство `macros` с макросами прототипа узла сети. |
| selectParentHost | query | Возвращать свойство `parentHost` с узлом сети, к которому принадлежит прототип узла сети. |
| selectTags | query | Возвращать свойство `tags` с тегами прототипа узла сети. |
| selectTemplates | query | Возвращать свойство `templates` с шаблонами, связанными с прототипом узла сети.<br><br>Поддерживает `count`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `hostid`, `host`, `name`, `status`, `discovered`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение прототипов узлов сети из правила LLD

Получите все прототипы узлов сети, их связи с группами, прототипы групп и теги из правила LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.get",
    "params": {
        "output": "extend",
        "selectInterfaces": "extend",
        "selectGroupLinks": "extend",
        "selectGroupPrototypes": "extend",
        "selectTags": "extend",
        "discoveryids": "23554"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10092",
            "host": "{#HV.UUID}",
            "name": "{#HV.UUID}",
            "status": "0",
            "templateid": "0",
            "flags": "2",
            "discover": "0",
            "custom_interfaces": "1",
            "inventory_mode": "-1",
            "groupLinks": [
                {
                    "group_prototypeid": "4",
                    "hostid": "10092",
                    "groupid": "7",
                    "templateid": "0"
                }
            ],
            "groupPrototypes": [
                {
                    "group_prototypeid": "7",
                    "hostid": "10092",
                    "name": "{#CLUSTER.NAME}",
                    "templateid": "0"
                }
            ],
            "tags": [
                {
                    "tag": "datacenter",
                    "value": "{#DATACENTER.NAME}"
                },
                {
                    "tag": "instance-type",
                    "value": "{#INSTANCE_TYPE}"
                }
            ],
            "interfaces": [
                {
                    "main": "1",
```

```
                    "type": "2",
                    "useip": "1",
                    "ip": "127.0.0.1",
                    "dns": "",
                    "port": "161",
                    "details": {
                        "version": "2",
                        "bulk": "1",
                        "community": "{$SNMP_COMMUNITY}",
                        "max_repetitions": "10"
                    }
                }
            ]
        }
    ],
    "id": 1
}
```


См. также

- Ссылка на группу
- Прототип группы
- Пользовательский макрос

Источник

CHostPrototype::get() в *ui/include/classes/api/services/CHostPrototype.php*.

**hostprototype.update**

Описание

`object hostprototype.update( / hostPrototypes)`

Этот метод позволяет обновлять существующие прототипы узлов сети.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства прототипа узла сети, которые необходимо обновить.

Свойство `hostid` должно быть определено для каждого прототипа узла сети, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств прототипа узла сети, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groupLinks | array | Связи групп, которые заменят текущие связи групп на прототипе узла сети.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |
| groupPrototypes | array | Прототипы групп, которые заменят существующие прототипы групп на прототипе узла сети.<br><br>Все прототипы групп, не перечисленные в запросе, будут удалены.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |

| Parameter | Type | Description |
|---|---|---|
| macros | object/array | Пользовательские макросы, которые заменят текущие пользовательские макросы.<br>Все макросы, не перечисленные в запросе, будут удалены. |
| tags | object/array | Теги прототипа узла сети, которые заменят текущие теги.<br>Все теги, не перечисленные в запросе, будут удалены.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |
| interfaces | object/array | Пользовательские интерфейсы прототипа узла сети, которые заменят текущие интерфейсы.<br>Объект пользовательского интерфейса должен содержать все свои параметры.<br>Все интерфейсы, не перечисленные в запросе, будут удалены.<br><br>Поведение параметра:<br>- *поддерживается*, если `custom_interfaces` объекта прототипа узла сети установлено в "use host prototypes custom interfaces"<br>- *только для чтения* для наследуемых объектов |
| templates | object/array | Шаблоны, которые заменят текущие связанные шаблоны.<br><br>У шаблонов должно быть определено только свойство `templateid`. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных прототипов узлов сети, указанных в свойстве `hostids`.

Примеры

Отключение прототипа узла сети

Отключите прототип узла сети, то есть установите для его статуса значение "1".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "status": 1
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Обновление тегов прототипа узла сети

Замените теги прототипа узла сети новыми.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
```

```
    "params": {
        "hostid": "10092",
        "tags": [
            {
                "tag": "datacenter",
                "value": "{#DATACENTER.NAME}"
            },
            {
                "tag": "instance-type",
                "value": "{#INSTANCE_TYPE}"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Обновление пользовательских интерфейсов узла сети-прототипа

Замените унаследованные интерфейсы пользовательскими интерфейсами узла сети-прототипа.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
```

```
        ]
    },
    "id": 1
}
```

См. также

- Ссылка на группу
- Прототип группы
- Тег прототипа узла сети
- Пользовательский интерфейс
- Пользовательский макрос

Источник

CHostPrototype::update() в *ui/include/classes/api/services/CHostPrototype.php*.

#### Прототип элемента данных

Этот класс предназначен для работы с прототипами элементов данных.

Ссылки на объекты:

- Прототип элемента данных
  - HTTP header
  - HTTP query field
- Тег прототипа элемента данных
- Предобработка прототипа элемента данных

Доступные методы:

- itemprototype.create - создать новые прототипы элементов данных
- itemprototype.delete - удалить прототипы элементов данных
- itemprototype.get - получить прототипы элементов данных
- itemprototype.update - обновить прототипы элементов данных

#### Объект прототипа элемента данных

Следующие объекты напрямую связаны с API `itemprototype`.

Прототип элемента данных

Объект прототипа элемента данных имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| itemid | ID | ID прототипа элемента данных.<br><br>Property behavior:<br>- *read-only*<br>- *required* for update operations |

| Property | Type | Description |
|---|---|---|
| delay | string | Интервал обновления прототипа элемента данных.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) и, дополнительно, один или несколько custom intervals, все разделенные точкой с запятой. Пользовательские интервалы могут представлять собой смесь гибких и интервальных расписаний.<br><br>Принимает пользовательские макросы и макросы LLD. Если используется, значение должно быть одним макросом. Несколько макросов или макросы, смешанные с текстом, не поддерживаются. Гибкие интервалы можно записывать как два макроса, разделенные косой чертой (например, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`).<br><br>Пример:<br>`1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr`<br><br>Property behavior:<br>- *required* if `type` is set to ”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), or if `type` is set to ”Zabbix agent (active)” (7) and `key_` does not contain ”mqtt.get” |
| hostid | ID | ID узла сети, которому принадлежит прототип элемента данных.<br><br>Property behavior:<br>- *constant*<br>- *required* for create operations |
| interfaceid | ID | ID интерфейса узла сети прототипа элемента данных.<br><br>Property behavior:<br>- *required* if item prototype belongs to host and `type` is set to ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, ”SNMP trap”, or ”SNMP agent”<br>- *supported* if item prototype belongs to host and `type` is set to ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent”, or ”HTTP agent” |
| key_ | string | Ключ прототипа элемента данных.<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |
| name | string | Имя прототипа элемента данных.<br>Поддерживает пользовательские макросы.<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| type | integer | Тип прототипа элемента данных.<br><br>Possible values:<br>0 - Zabbix agent;<br>2 - Zabbix trapper;<br>3 - Simple check;<br>5 - Zabbix internal;<br>7 - Zabbix agent (active);<br>10 - External check;<br>11 - Database monitor;<br>12 - IPMI agent;<br>13 - SSH agent;<br>14 - TELNET agent;<br>15 - Calculated;<br>16 - JMX agent;<br>17 - SNMP trap;<br>18 - Dependent item;<br>19 - HTTP agent;<br>20 - SNMP agent;<br>21 - Script;<br>22 - Browser.<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |
| url | string | Строка URL.<br>Поддерживает макросы LLD, пользовательские макросы, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}.<br><br>Property behavior:<br>- *required* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| value_type | integer | Тип информации прототипа элемента данных.<br><br>Possible values:<br>0 - numeric float;<br>1 - character;<br>2 - log;<br>3 - numeric unsigned;<br>4 - text;<br>5 - binary.<br><br>Property behavior:<br>- *required* for create operations<br>- *read-only* for inherited objects |
| allow_traps | integer | Разрешает заполнять значение аналогично элементу данных типа trapper.<br><br>0 - *(default)* Не разрешать принимать входящие данные;<br>1 - Разрешить принимать входящие данные.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent” |

| Property | Type | Description |
|---|---|---|
| authtype | integer | Метод аутентификации.<br><br>Possible values if `type` is set to ”SSH agent”:<br>0 - *(default)* password;<br>1 - public key.<br><br>Possible values if `type` is set to ”HTTP agent”:<br>0 - *(default)* none;<br>1 - basic;<br>2 - NTLM;<br>3 - Kerberos;<br>4 - Digest.<br><br>Property behavior:<br>- *supported* if `type` is set to ”SSH agent” or ”HTTP agent”<br>- *read-only* for inherited objects (if `type` is set to ”HTTP agent”) |
| description | string | Описание прототипа элемента данных. |
| flags | integer | Origin прототипа элемента данных.<br><br>Possible values:<br>2 - an item prototype;<br>6 - a discovered item prototype<br><br>Property behavior:<br>- *read-only* |
| follow_redirects | integer | Следовать перенаправлениям ответа при опросе данных.<br><br>Possible values:<br>0 - Do not follow redirects;<br>1 - *(default)* Follow redirects.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| headers | array | Массив headers, которые будут отправлены при выполнении HTTP-запроса.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| history | string | Единица времени, определяющая, как долго должны храниться данные истории.<br>Также принимает пользовательский макрос и макрос LLD.<br><br>По умолчанию: 31d. |
| http_proxy | string | Строка подключения к HTTP(S) proxy.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| ipmi_sensor | string | Датчик IPMI.<br><br>Property behavior:<br>- *required* if `type` is set to ”IPMI agent” and `key_` is not set to ”ipmi.get”<br>- *supported* if `type` is set to ”IPMI agent”<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| jmx_endpoint | string | Пользовательская строка подключения JMX gateway.<br><br>По умолчанию:<br>service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi<br><br>Property behavior:<br>- *supported* if `type` is set to ”JMX agent” |
| logtimefmt | string | Формат времени в записях журнала.<br><br>Property behavior:<br>- *supported* if `value_type` is set to ”log”<br>- *read-only* for inherited objects |
| master_itemid | ID | ID основного элемента данных.<br><br>Property behavior:<br>- *required* if `type` is set to ”Dependent item”<br>- *read-only* for inherited objects |
| output_format | integer | Следует ли преобразовывать ответ в JSON.<br><br>Possible values:<br>0 - *(default)* Store raw;<br>1 - Convert to JSON.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| params | string | Дополнительные параметры в зависимости от типа прототипа элемента данных:<br>- выполняемый скрипт для прототипов элементов данных SSH agent и TELNET agent;<br>- SQL-запрос для прототипов элементов данных Database monitor;<br>- формула для прототипов вычисляемых элементов данных;<br>- скрипт для прототипов элементов данных Script и Browser.<br><br>Property behavior:<br>- *required* if `type` is set to ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Calculated”, ”Script”, or ”Browser”<br>- *read-only* for inherited objects (if `type` is set to ”Script” or ”Browser”) |
| parameters | object/array | Дополнительные параметры, если `type` установлен в ”Script” или ”Browser”. Массив объектов со свойствами `name` и `value`, где `name` должен быть уникальным.<br><br>Property behavior:<br>- *supported* if `type` is set to ”Script” or ”Browser”<br>- *read-only* for inherited objects |
| password | string | Пароль для аутентификации.<br><br>Property behavior:<br>- *required* if `type` is set to ”JMX agent” and `username` is set<br>- *supported* if `type` is set to ”Simple check”, ”SSH agent”, ”TELNET agent”, ”Database monitor”, or ”HTTP agent”<br>- *read-only* for inherited objects (if `type` is set to ”HTTP agent”) |
| post_type | integer | Тип тела POST-данных, хранящихся в свойстве `posts`.<br><br>Possible values:<br>0 - *(default)* Raw data.<br>2 - JSON data.<br>3 - XML data.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| posts | string | Данные тела HTTP(S)-запроса.<br><br>Property behavior:<br>- *required* if `type` is set to ”HTTP agent” and `post_type` is set to ”JSON data” or ”XML data”<br>- *supported* if `type` is set to ”HTTP agent” and `post_type` is set to ”Raw data”<br>- *read-only* for inherited objects |
| privatekey | string | Имя файла закрытого ключа.<br><br>Property behavior:<br>- *required* if `type` is set to ”SSH agent” and `authtype` is set to ”public key” |
| publickey | string | Имя файла открытого ключа.<br><br>Property behavior:<br>- *required* if `type` is set to ”SSH agent” and `authtype` is set to ”public key” |
| query_fields | array | Массив query fields, которые будут отправлены при выполнении HTTP-запроса.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| request_method | integer | Тип метода запроса.<br><br>Possible values:<br>0 - *(default)* GET;<br>1 - POST;<br>2 - PUT;<br>3 - HEAD.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| retrieve_mode | integer | Какая часть ответа должна быть сохранена.<br><br>Possible values if `request_method` is set to ”GET”, ”POST”, or ”PUT”:<br>0 - *(default)* Body;<br>1 - Headers;<br>2 - Both body and headers will be stored.<br><br>Possible values if `request_method` is set to ”HEAD”:<br>1 - Headers.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| snmp_oid | string | SNMP OID.<br><br>Property behavior:<br>- *required* if `type` is set to ”SNMP agent”<br>- *read-only* for inherited objects |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |

| Property | Type | Description |
|---|---|---|
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| status | integer | Статус прототипа элемента данных.<br><br>Possible values:<br>0 - *(default)* enabled item prototype;<br>1 - disabled item prototype;<br>3 - unsupported item prototype. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми.<br>Также поддерживает пользовательские макросы или макросы LLD как часть списка, разделенного запятыми.<br><br>Пример: 200,200-{$M},{$M},200-400<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| templateid | ID | ID родительского шаблона прототипа элемента данных.<br><br>Property behavior:<br>- *read-only* |
| timeout | string | Тайм-аут запроса опроса данных элемента данных.<br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m). Также принимает пользовательские макросы и макросы LLD.<br><br>Диапазон допустимых значений: 1-600s.<br><br>По умолчанию: ”” - использовать настройки proxy/глобальные настройки.<br><br>Property behavior:<br>- *supported* if `type` is set to ”Zabbix agent” (0), ”Simple check” (3) and `key_` does not start with ”`vmware.`” and ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) and `snmp_oid` starts with ”`walk[`” or ”`get[`”, ”Script” (21), ”Browser” (22)<br>- *read-only* for inherited objects |
| trapper_hosts | string | Разрешенные узлы сети.<br><br>Property behavior:<br>- *supported* if `type` is set to ”Zabbix trapper”, or if `type` is set to ”HTTP agent” and `allow_traps` is set to ”Allow to accept incoming data” |
| trends | string | Единица времени, определяющая, как долго должны храниться данные трендов.<br>Также принимает пользовательский макрос и макрос LLD.<br><br>По умолчанию: 365d.<br><br>Property behavior:<br>- *supported* if `value_type` is set to ”numeric float” or ”numeric unsigned” |

| Property | Type | Description |
|---|---|---|
| units | string | Единицы значения.<br><br>Property behavior:<br>- *supported* if `value_type` is set to ”numeric float” or ”numeric unsigned”<br>- *read-only* for inherited objects |
| username | string | Имя пользователя для аутентификации.<br><br>Property behavior:<br>- *required* if `type` is set to ”SSH agent” or ”TELNET agent”, or if `type` is set to ”JMX agent” and `password` is set<br>- *supported* if `type` is set to ”Simple check”, ”Database monitor”, or ”HTTP agent”<br>- *read-only* for inherited objects (if `type` is set to ”HTTP agent”) |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных прототипов элементов данных с уже существующими. Генерируется автоматически, если не указан.<br><br>Property behavior:<br>- *supported* if the item prototype belongs to a template |
| valuemapid | ID | ID связанной карты значений.<br><br>Property behavior:<br>- *supported* if `value_type` is set to ”numeric float”, ”character”, or ”numeric unsigned”<br>- *read-only* for inherited objects |
| verify_host | integer | Следует ли проверять, что имя узла сети для соединения совпадает с именем в сертификате узла сети.<br><br>Possible values:<br>0 - *(default)* Do not validate;<br>1 - Validate.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| verify_peer | integer | Следует ли проверять, что сертификат узла сети является подлинным.<br><br>Possible values:<br>0 - *(default)* Do not validate;<br>1 - Validate.<br><br>Property behavior:<br>- *supported* if `type` is set to ”HTTP agent”<br>- *read-only* for inherited objects |
| discover | integer | Статус обнаружения прототипа элемента данных.<br><br>Possible values:<br>0 - *(default)* new items will be discovered;<br>1 - new items will not be discovered and existing items will be marked as lost. |

HTTP-заголовок

Объект заголовка имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя HTTP-заголовка.<br>Поведение свойства:<br>- *required* |
| value | string | Значение заголовка.<br>Поведение свойства:<br>- *required* |

Поле HTTP-запроса

Объект поля запроса определяет имя и значение, которые используются для указания параметра URL. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя параметра.<br>Property behavior:<br>- *required* |
| value | string | Значение параметра.<br>Property behavior:<br>- *required* |

Тег прототипа элемента данных

Объект тега прототипа элемента данных имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега прототипа элемента данных.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение тега прототипа элемента данных. |

Предобработка прототипа элемента данных

Объект предобработки прототипа элемента данных имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип опции предобработки.<br><br>Возможные значения:<br>1 - Пользовательский множитель;<br>2 - Обрезка справа;<br>3 - Обрезка слева;<br>4 - Обрезка;<br>5 - Регулярное выражение;<br>6 - Булево в десятичное;<br>7 - Восьмеричное в десятичное;<br>8 - Шестнадцатеричное в десятичное;<br>9 - Простое изменение;<br>10 - Изменение за секунду;<br>11 - XML XPath;<br>12 - JSONPath;<br>13 - В диапазоне;<br>14 - Соответствует регулярному выражению;<br>15 - Не соответствует регулярному выражению;<br>16 - Проверка ошибки в JSON;<br>17 - Проверка ошибки в XML;<br>18 - Проверка ошибки с использованием регулярного выражения;<br>19 - Отбрасывать неизменившееся;<br>20 - Отбрасывать неизменившееся с heartbeat;<br>21 - JavaScript;<br>22 - Шаблон Prometheus;<br>23 - Prometheus в JSON;<br>24 - CSV в JSON;<br>25 - Замена;<br>26 - Проверка неподдерживаемого;<br>27 - XML в JSON;<br>28 - Значение обхода SNMP;<br>29 - Обход SNMP в JSON;<br>30 - Получение значения SNMP.<br><br>Property behavior:<br>- *required* |
| params | string | Дополнительные параметры, используемые опцией предобработки.<br>Несколько параметров разделяются символом новой строки (\n).<br><br>Если `type` установлен в ”Check unsupported”, параметры следуют синтаксису `<scope>[\n<pattern>]`, где *pattern* - регулярное выражение, а *scope* - одно из:<br>-1 - сопоставить любую ошибку;<br>0 - проверить, совпадает ли сообщение об ошибке с *pattern*;<br>1 - проверить, не совпадает ли сообщение об ошибке с *pattern*.<br><br>Property behavior:<br>- *required* if `type` is set to ”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), or ”SNMP get value” (30) |

| Property | Type | Description |
|---|---|---|
| error_handler | integer | Тип действия, используемый в случае сбоя шага предобработки.<br><br>Возможные значения:<br>0 - Сообщение об ошибке задается сервером Zabbix;<br>1 - Отбросить значение;<br>2 - Установить пользовательское значение;<br>3 - Установить пользовательское сообщение об ошибке.<br><br>Возможные значения, если `type` установлен в ”Check unsupported”:<br>1 - Отбросить значение;<br>2 - Установить пользовательское значение;<br>3 - Установить пользовательское сообщение об ошибке.<br><br>Property behavior:<br>- *required* if `type` is set to ”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), or ”SNMP get value” (30) |
| error_handler_params | string | Параметры обработчика ошибок.<br><br>Property behavior:<br>- *required* if `error_handler` is set to ”Set custom value” or ”Set custom error message” |

Для каждого типа предобработки поддерживаются следующие параметры и обработчики ошибок.

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 1 | Custom multiplier | number^1, 6^ | | | 0, 1, 2, 3 |
| 2 | Right trim | list of characters[2] | | | |
| 3 | Left trim | list of characters[2] | | | |
| 4 | Trim | list of characters[2] | | | |
| 5 | Regular expression | pattern[3] | output[2] | | 0, 1, 2, 3 |
| 6 | Boolean to decimal | | | | 0, 1, 2, 3 |
| 7 | Octal to decimal | | | | 0, 1, 2, 3 |
| 8 | Hexadecimal to decimal | | | | 0, 1, 2, 3 |

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 9 | Simple change | | | | 0, 1, 2, 3 |
| 10 | Change per second | | | | 0, 1, 2, 3 |
| 11 | XML XPath | path[4] | | | 0, 1, 2, 3 |
| 12 | JSONPath | path[4] | | | 0, 1, 2, 3 |
| 13 | In range | min^1, 6^ | max^1, 6^ | | 0, 1, 2, 3 |
| 14 | Matches regular expression | pattern[3] | | | 0, 1, 2, 3 |
| 15 | Does not match regular expression | pattern[3] | | | 0, 1, 2, 3 |
| 16 | Check for error in JSON | path[4] | | | 0, 1, 2, 3 |
| 17 | Check for error in XML | path[4] | | | 0, 1, 2, 3 |
| 18 | Check for error using regular expression | pattern[3] | output[2] | | 0, 1, 2, 3 |
| 19 | Discard unchanged | | | | |
| 20 | Discard unchanged with heartbeat | seconds^5, 6^ | | | |
| 21 | JavaScript | script[2] | | | |
| 22 | Prometheus pattern | pattern^6, 7^ | `value`, `label`, `function` | output^8, 9^ | 0, 1, 2, 3 |
| 23 | Prometheus to JSON | pattern^6, 7^ | | | 0, 1, 2, 3 |

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 24 | CSV to JSON | character[2] | character[2] | 0,1 | 0, 1, 2, 3 |
| 25 | Replace | search string[2] | replacement[2] | | |
| 26 | Check un-sup-ported | scope[1] | pattern^3, 6^ | | 1, 2, 3 |
| 27 | XML to JSON | | | | 0, 1, 2, 3 |
| 28 | SNMP walk value | OID[2] | Format:<br>0 - Unchanged<br>1 - UTF-8 from Hex-STRING<br>2 - MAC from Hex-STRING<br>3 - Integer from BITS | | 0, 1, 2, 3 |
| 29 | SNMP walk to JSON[10] | Field name[2] | OID prefix[2] | Format:<br>0 - Unchanged<br>1 - UTF-8 from Hex-STRING<br>2 - MAC from Hex-STRING<br>3 - Integer from BITS | 0, 1, 2, 3 |
| 30 | SNMP get value | Format:<br>1 - UTF-8 from Hex-STRING<br>2 - MAC from Hex-STRING<br>3 - Integer from BITS | | | 0, 1, 2, 3 |

[1] целое или число с плавающей точкой<br> [2] строка<br> [3] регулярное выражение<br> [4] JSONPath или XML XPath<br> [5] положительное целое число (с поддержкой суффиксов времени, например 30s, 1m, 2h, 1d)<br> [6] пользовательский макрос, макрос LLD<br> [7] шаблон Prometheus в синтаксисе: `<metric name>{<label name>="<label value>", ...} == <value>`. Каждый компонент шаблона Prometheus (метрика, имя метки, значение метки и значение метрики) может быть пользовательским макросом или макросом LLD.<br> [8] вывод Prometheus в синтаксисе: `<label name>` (может быть пользовательским макросом или макросом LLD), если `label` выбран в качестве второго параметра.<br> [9] Одна из функций агрегации: `sum`, `min`, `max`, `avg`, `count`, если `function` выбран в качестве второго параметра.<br> [10] Поддерживает несколько записей "Field name,OID prefix,Format records", разделенных символом новой строки.

**itemprototype.create**

Описание

`object itemprototype.create( / itemPrototypes)`

Этот метод позволяет создавать новые прототипы элементов данных.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Прототип элемента данных для создания.

Помимо стандартных свойств прототипа элемента данных, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| ruleid | ID | ID правила LLD, которому принадлежит элемент данных.<br><br>Поведение параметра:<br>*- required* |
| preprocessing | array | Параметры предобработки прототипа элемента данных. |
| tags | array | Теги прототипа элемента данных. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных прототипов элементов данных, указанных в свойстве `itemids`. Порядок возвращаемых ID совпадает с порядком переданных прототипов элементов данных.

Примеры

Создание прототипа элемента данных

Создайте прототип элемента данных для мониторинга свободного места на диске в обнаруженной файловой системе. Обнаруженные элементы должны быть числовыми элементами данных агента Zabbix, обновляемыми каждые 30 секунд.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "                  {#FSNAME}",
        "key_": "vfs.fs.size[{#FSNAME},free]",
        "hostid": "10197",
        "ruleid": "27665",
        "type": 0,
        "value_type": 3,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27666"
        ]
    },
    "id": 1
}
```

Создание прототипа элемента данных с предварительной обработкой

Создайте элемент данных, используя изменение за секунду и пользовательский множитель в качестве второго шага.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Создание прототипа зависимого элемента данных

Создайте прототип зависимого элемента данных для прототипа мастер-элемента с ID 44211. Разрешены только зависимости на одном и том же узле сети (шаблон/правило обнаружения), поэтому мастер-элемент и зависимый элемент данных должны иметь одинаковые hostid и ruleid.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44212"
        ]
    },
    "id": 1
}
```

Создание прототипа элемента данных HTTP-агента

Создайте прототип элемента данных с URL, используя пользовательский макрос, поля запроса и пользовательские заголовки.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Создание прототипа элемента данных типа script

Создайте простую сборку данных с помощью прототипа элемента данных типа script.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Источник

CItemPrototype::create() в *ui/include/classes/api/services/CItemPrototype.php*.

**itemprototype.delete**

Описание

`object itemprototype.delete(    itemPrototypeIds)`

Этот метод позволяет удалять прототипы элементов данных.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых прототипов элементов данных.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных прототипов элементов данных, указанных в свойстве `prototypeids`.

Примеры

Удаление нескольких прототипов элементов данных

Удаление двух прототипов элементов данных.<br> Зависимые прототипы элементов данных удаляются автоматически, если удален мастер-элемент данных или прототип элемента данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.delete",
    "params": [
        "27352",
        "27356"
    ],
```

```
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "prototypeids": [
            "27352",
            "27356"
        ]
    },
    "id": 1
}
```

Источник

CItemPrototype::delete() в *ui/include/classes/api/services/CItemPrototype.php*.

**itemprototype.get**

Описание

`integer/array itemprototype.get(          )`

Этот метод позволяет получать прототипы элементов данных в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| discoveryids | ID/array | Возвращать только прототипы элементов данных, которые принадлежат указанным правилам LLD. |
| graphids | ID/array | Возвращать только прототипы элементов данных, которые используются в указанных прототипах графиков. |
| hostids | ID/array | Возвращать только прототипы элементов данных, которые принадлежат указанным узлам сети. |
| inherited | boolean | Если задано `true`, возвращать только прототипы элементов данных, унаследованные от шаблона. |
| itemids | ID/array | Возвращать только прототипы элементов данных с указанными ID. |
| monitored | boolean | Если задано `true`, возвращать только включенные прототипы элементов данных, которые принадлежат контролируемым узлам сети. |
| templated | boolean | Если задано `true`, возвращать только прототипы элементов данных, которые принадлежат шаблонам. |
| templateids | ID/array | Возвращать только прототипы элементов данных, которые принадлежат указанным шаблонам. |
| triggerids | ID/array | Возвращать только прототипы элементов данных, которые используются в указанных прототипах триггеров. |

| Parameter | Type | Description |
|---|---|---|
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения прототипа элемента данных. Объект обнаружения прототипа элемента данных связывает обнаруженный прототип элемента данных с прототипом элемента данных, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`parent_itemid` - `(string)` ID прототипа элемента данных, из которого был создан прототип элемента данных;<br>`key_` - `(string)` ключ прототипа элемента данных;<br>`status` - `(int)` статус обнаружения прототипа элемента данных:<br>0 - *(по умолчанию)* прототип элемента данных обнаружен,<br>1 - прототип элемента данных больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда прототип элемента данных, который больше не обнаружен, будет удален;<br>`ts_disable` - `(timestamp)` время, когда прототип элемента данных, который больше не обнаружен, будет отключен;<br>`disable_source` - `(int)` признак того, был ли прототип элемента данных отключен правилом LLD или вручную:<br>0 - *(по умолчанию)* отключен автоматически,<br>1 - отключен правилом LLD. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с правилом низкоуровневого обнаружения, к которому принадлежит прототип элемента данных. |
| selectDiscoveryRulePrototype | query | Возвращать свойство `discoveryRulePrototype` с родительским прототипом правила LLD, к которому принадлежит прототип элемента данных. |
| selectGraphs | query | Возвращать свойство `graphs` с прототипами графиков, в которых используется прототип элемента данных.<br><br>Поддерживает `count`. |
| selectHosts | query | Возвращать свойство `hosts` с массивом узлов сети, к которым принадлежит прототип элемента данных. |
| selectTags | query | Возвращать теги прототипа элемента данных в свойстве `tags`. |
| selectTriggers | query | Возвращать свойство `triggers` с прототипами триггеров, в которых используется прототип элемента данных.<br><br>Поддерживает `count`. |
| selectPreprocessing | query | Возвращать свойство `preprocessing` с параметрами предварительной обработки прототипа элемента данных. |
| selectValueMap | query | Возвращать свойство `valuemap` с картой значений прототипа элемента данных. |
| filter | object | Возвращать только те результаты, которые в точности соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит прототип элемента данных. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подвыборками.<br><br>Применяется к следующим подвыборкам:<br>`selectGraphs` - результаты будут отсортированы по `name`;<br>`selectTriggers` - результаты будут отсортированы по `description`. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `itemid`, `name`, `key_`, `delay`, `type`, `status`, `history`, `trends`, `discovered`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение прототипов элементов данных из правила LLD

Получить все прототипы элементов данных для указанного ID правила LLD.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23077",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10079",
            "name": "                  en0",
            "key_": "net.if.in[en0]",
            "delay": "1m",
            "history": "1w",
            "trends": "365d",
            "status": "0",
            "value_type": "3",
            "trapper_hosts": "",
            "units": "bps",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
```

```
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "2",
        "interfaceid": "0",
        "description": "",
        "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": "",
        "parameters": []
    },
    {
        "itemid": "10010",
        "type": "0",
        "snmp_oid": "",
        "hostid": "10001",
        "name": "              (       1         )",
        "key_": "system.cpu.load[percpu,avg1]",
        "delay": "1m",
        "history": "1w",
        "trends": "365d",
        "status": "0",
        "value_type": "0",
        "trapper_hosts": "",
        "units": "",
        "logtimefmt": "",
        "templateid": "0",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "2",
        "interfaceid": "0",
        "description": "                              CPU     ,                  CPU.",
        "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": "",
            "parameters": []
        }
    ],
    "id": 1
}
```

Поиск зависимого элемента данных

Найдите один зависимый элемент данных для указанного ID элемента данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "filter": {
            "type": 18,
            "master_itemid": "25545"
        },
        "limit": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "evaltype": "0",
            "master_itemid": "25545",
            "jmx_endpoint": "",
            "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": "",
            "parameters": []
        }
    ],
    "id": 1
}
```

Поиск прототипа элемента данных HTTP-агента

Найдите прототип элемента данных HTTP-агента с методом запроса HEAD для указанного ID узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "hostids": "10254",
        "filter": {
            "type": 19,
            "request_method": 3
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "",
            "logtimefmt": "",
            "templateid": "28255",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "2",
            "interfaceid": "2",
            "description": "",
            "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": "",
            "parameters": []
        }
    ],
    "id": 1
}
```


См. также

- Узел сети
- Прототип графика

- Прототип триггера

Источник

CItemPrototype::get() в *ui/include/classes/api/services/CItemPrototype.php*.

### itemprototype.update

Описание

`object itemprototype.update( / itemPrototypes)`

Этот метод позволяет обновлять существующие прототипы элементов данных.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства прототипа элемента данных, которые необходимо обновить.

Для каждого прототипа элемента данных должно быть определено свойство `itemid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств прототипа элемента данных, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| preprocessing | array | Параметры предобработки прототипа элемента данных для замены текущих параметров предобработки.<br><br>Поведение параметра:<br>- *только для чтения* для наследуемых объектов |
| tags | array | Теги прототипа элемента данных. |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID обновленных прототипов элементов данных, указанных в свойстве `itemids`.

Примеры

Изменение интерфейса прототипа элемента данных

Измените интерфейс узла сети, который будет использоваться обнаруженными элементами данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "27428",
        "interfaceid": "132"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27428"
        ]
    },
```

```
    "id": 1
}
```

Обновление прототипа зависимого элемента данных

Обновляет прототип зависимого элемента данных с новым ID прототипа мастер-элемента. Допускаются только зависимости на одном и том же узле сети (шаблон/правило обнаружения), поэтому мастер-элемент и зависимый элемент должны иметь одинаковые hostid и ruleid.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "master_itemid": "25570",
        "itemid": "189030"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189030"
        ]
    },
    "id": 1
}
```

Обновление прототипа элемента данных HTTP-агента

Измените поля запроса и удалите все пользовательские заголовки.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid":"28305",
        "query_fields": [
            {
                "name": "random",
                "value": "qwertyuiopasdfghjklzxcvbnm"
            }
        ],
        "headers": []
    }
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Обновление параметров предварительной обработки элемента данных

Обновите прототип элемента данных с правилом предварительной обработки элемента данных “Custom multiplier”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "44211",
        "preprocessing": [
            {
                "type": 1,
                "params": "4",
                "error_handler": 2,
                "error_handler_params": "5"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Обновление прототипа элемента данных типа скрипт

Обновите прототип элемента данных типа скрипт, указав другой скрипт, и удалите ненужные параметры, которые использовались предыдущим скриптом.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Источник

CItemPrototype::update() в *ui/include/classes/api/services/CItemPrototype.php*.

### Регулярное выражение

Этот класс предназначен для работы с глобальными регулярными выражениями.

Ссылки на объекты:

- Регулярное выражение
- Выражения

Доступные методы:

- regexp.create - создать новые регулярные выражения
- regexp.delete - удалить регулярные выражения
- regexp.get - получить регулярные выражения
- regexp.update - обновить регулярные выражения

### Объект регулярного выражения

Следующие объекты напрямую связаны с API `regexp`.

Регулярное выражение

Глобальный объект регулярного выражения имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| regexpid | ID | ID регулярного выражения.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя регулярного выражения.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| test_string | string | Тестовая строка. |

Выражения

Объект expressions имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| expression | string | Регулярное выражение.<br><br>Поведение свойства:<br>- *required* |
| expression_type | integer | Тип регулярного выражения.<br><br>Возможные значения:<br>0 - Включена символьная строка;<br>1 - Включена любая символьная строка;<br>2 - Символьная строка не включена;<br>3 - Результат TRUE;<br>4 - Результат FALSE.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| exp_delimiter | string | Разделитель выражения.<br><br>Значение по умолчанию: ",".<br><br>Возможные значения: "," или ".", или "/".<br><br>Поведение свойства:<br>- *supported* if expression_type is set to ”Any character string included” |
| case_sensitive | integer | Учет регистра.<br><br>Значение по умолчанию: 0.<br><br>Возможные значения:<br>0 - Без учета регистра;<br>1 - С учетом регистра. |

**regexp.create**

Описание

`object regexp.create(  /  regularExpressions)`

Этот метод позволяет создавать новые глобальные регулярные выражения.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Регулярные выражения для создания.

Помимо стандартных свойств, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| expressions | array | Параметры Expressions.<br><br>Parameter behavior:<br>- *required* |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID созданных регулярных выражений, указанных в свойстве`regexpids`.

Примеры

Создание нового глобального регулярного выражения.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.create",
    "params": {
      "name": "Storage devices for SNMP discovery",
      "test_string": "/boot",
      "expressions": [
        {
          "expression": "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$",
          "expression_type": "4",
          "case_sensitive": "1"
```

```
        }
      ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "16"
        ]
    },
    "id": 1
}
```

Источник

CRegexp::create() в *ui/include/classes/api/services/CRegexp.php*.

**regexp.delete**

Описание

`object regexp.delete(    regexpids)`

Этот метод позволяет удалять глобальные регулярные выражения.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(  ) ID удаляемых регулярных выражений.

Возвращаемые значения

(  ) Возвращает объект, который содержит ID удалённых регулярных выражений, указанных в свойстве`regexpids`.

Примеры

Удаление нескольких глобальных регулярных выражений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.delete",
    "params": [
        "16",
        "17"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "16",
            "17"
        ]
    },
```

```
    "id": 1
}
```

Источник

CRegexp::delete() в *ui/include/classes/api/services/CRegexp.php*.

**regexp.get**

Описание

`integer/array regexp.get(  )`

Метод позволяет получать глобальные регулярные выражения по заданным параметрам.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| regexpids | ID/array | Возвращать только регулярные выражения с указанными ID. |
| selectExpressions | query | Возвращать свойство `expressions`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `regexpid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(  /  )` Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение глобальных регулярных выражений.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.get",
    "params": {
        "output": ["regexpid", "name"],
        "selectExpressions": ["expression", "expression_type"],
        "regexpids": [1, 2],
        "preservekeys": true
```

```
    },
    "id": 1
}
```


Ответ:


```
{
  "jsonrpc": "2.0",
  "result": {
    "1": {
      "regexpid": "1",
      "name": "File systems for discovery",
      "expressions": [
        {
          "expression": "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|
          "expression_type": "3"
        }
      ]
    },
    "2": {
      "regexpid": "2",
      "name": "Network interfaces for discovery",
      "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
}
```


Источник

CRegexp::get() в *ui/include/classes/api/services/CRegexp.php*.

**regexp.update**

Описание

`object regexp.update(    /    RegularExpressions)`

Этот метод позволяет обновлять существующие глобальные регулярные выражения.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства регулярного выражения, которые нужно обновить.

Свойство `regexpid` должно быть определено для каждого объекта, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| expressions | array | Параметры Expressions. |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных регулярных выражений, указанных в свойстве`regexpids`.

Примеры

Обновление глобального регулярного выражения для обнаружения файловых систем.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.update",
    "params": {
      "regexpid": "1",
      "name": "File systems for discovery",
      "test_string": "",
      "expressions": [
        {
          "expression": "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|zfs)$",
          "expression_type": "3",
          "exp_delimiter": ",",
          "case_sensitive": "0"
        },
        {
          "expression": "^(ntfs|fat32|fat16)$",
          "expression_type": "3",
          "exp_delimiter": ",",
          "case_sensitive": "0"
        }
      ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "1"
        ]
    },
    "id": 1
}
```

Источник

CRegexp::update() в *ui/include/classes/api/services/CRegexp.php*.

### Роль

Этот класс предназначен для работы с ролями пользователей.

Ссылки на объекты:

- Role
- Role rules
  - UI element
  - Service
  - Service tag
  - Module
  - Action

Доступные методы:

- role.create - создать новые роли пользователей
- role.delete - удалить роли пользователей
- role.get - получить роли пользователей
- role.update - обновить роли пользователей

### Объект роли

Следующие объекты напрямую связаны с API `role`.

Роль

Объект role имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| roleid | ID | ID роли.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя роли.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| type | integer | Тип пользователя.<br><br>Возможные значения:<br>1 - *(по умолчанию)* Пользователь;<br>2 - Администратор;<br>3 - Суперадминистратор.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| readonly | integer | Указывает, является ли роль доступной только для чтения.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Нет;<br>1 - Да.<br><br>Поведение свойства:<br>- *только для чтения* |

Правила роли

Объект правил роли имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| ui | array | Массив объектов элемент интерфейса пользователя. |

| Property | Type | Description |
|---|---|---|
| ui.default_access | integer | Включен ли доступ к новым элементам интерфейса пользователя.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| services.read.mode | integer | Доступ только для чтения к сервисам.<br><br>Возможные значения:<br>0 - Доступ только для чтения к сервисам, указанным в `services.read.list` или совпадающим со свойством `services.read.tag`;<br>1 - *(по умолчанию)* Доступ только для чтения ко всем сервисам. |
| services.read.list | array | Массив объектов Service.<br><br>Указанным сервисам, включая дочерние сервисы, будет предоставлен доступ только для чтения для пользовательской роли. Доступ только для чтения не переопределяет доступ на чтение и запись к сервисам.<br><br>Property behavior:<br>- *supported* if `services.read.mode` is set to "0" |
| services.read.tag | object | Массив объектов Service tag.<br><br>Сервисам, совпавшим по тегу, включая дочерние сервисы, будет предоставлен доступ только для чтения для пользовательской роли. Доступ только для чтения не переопределяет доступ на чтение и запись к сервисам.<br><br>Property behavior:<br>- *supported* if `services.read.mode` is set to "0" |
| services.write.mode | integer | Доступ на чтение и запись к сервисам.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Доступ на чтение и запись к сервисам, указанным в `services.write.list` или совпадающим со свойством `services.write.tag`;<br>1 - Доступ на чтение и запись ко всем сервисам. |
| services.write.list | array | Массив объектов Service.<br><br>Указанным сервисам, включая дочерние сервисы, будет предоставлен доступ на чтение и запись для пользовательской роли. Доступ на чтение и запись переопределяет доступ только для чтения к сервисам.<br><br>Property behavior:<br>- *supported* if `services.write.mode` is set to "0" |
| services.write.tag | object | Массив объектов Service tag.<br><br>Сервисам, совпавшим по тегу, включая дочерние сервисы, будет предоставлен доступ на чтение и запись для пользовательской роли. Доступ на чтение и запись переопределяет доступ только для чтения к сервисам.<br><br>Property behavior:<br>- *supported* if `services.write.mode` is set to "0" |
| modules | array | Массив объектов module. |
| modules.default_access | integer | Включен ли доступ к новым модулям.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |

| Property | Type | Description |
|---|---|---|
| api.access | integer | Включен ли доступ к API.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |
| api.mode | integer | Режим обработки методов API, перечисленных в свойстве `api`.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Список запрещенных;<br>1 - Список разрешенных. |
| api | array | Массив методов API. |
| actions | array | Массив объектов action. |
| actions.default_access | integer | Включен ли доступ к новым действиям.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |

UI element

Объект UI element имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя UI element.<br><br>Возможные значения, если type объекта Role задан как "User", "Admin" или "Super admin":<br>monitoring.dashboard - *Панели*;<br>monitoring.problems - *Мониторинг → Проблемы*;<br>monitoring.hosts - *Мониторинг → Узлы сети*;<br>monitoring.latest_data - *Мониторинг → Последние данные*;<br>monitoring.maps - *Мониторинг → Карты*;<br>services.services - *Услуги → Услуги*;<br>services.sla_report - *Услуги → Отчет SLA*;<br>inventory.overview - *Инвентаризация → Обзор*;<br>inventory.hosts - *Инвентаризация → Узлы сети*;<br>reports.availability_report - *Отчеты → Отчет о доступности*;<br>reports.top_triggers - *Отчеты → Триггеры топ 100*.<br><br>Возможные значения, если type объекта Role задан как "Admin" или "Super admin":<br>monitoring.discovery - *Мониторинг → Обнаружение*;<br>services.sla - *Услуги → SLA*;<br>reports.scheduled_reports - *Отчеты → Запланированные отчеты*;<br>reports.notifications - *Отчеты → Уведомления*;<br>configuration.template_groups - *Сбор данных → Группы шаблонов*;<br>configuration.host_groups - *Сбор данных → Группы узлов сети*;<br>configuration.templates - *Сбор данных → Шаблоны*;<br>configuration.hosts - *Сбор данных → Узлы сети*;<br>configuration.maintenance - *Сбор данных → Обслуживание*;<br>configuration.discovery - *Сбор данных → Обнаружение*;<br>configuration.trigger_actions - *Оповещения → Действия → Действия по триггерам*;<br>configuration.service_actions - *Оповещения → Действия → Действия по услугам*;<br>configuration.discovery_actions - *Оповещения → Действия → Действия по обнаружению*;<br>configuration.autoregistration_actions - *Оповещения → Действия → Действия автрорегистрации*;<br>configuration.internal_actions - *Оповещения → Действия → Внутренние действия*.<br><br>Возможные значения, если type объекта Role задан как "Super admin":<br>reports.system_info - *Отчеты → Сведения о системе*;<br>reports.audit - *Отчеты → Журнал аудита*;<br>reports.action_log - *Отчеты → Журнал действий*;<br>configuration.event_correlation - *Сбор данных → Корреляция событий*;<br>administration.media_types - *Оповещения → Типы медиа*;<br>administration.scripts - *Оповещения → Скрипты*;<br>administration.user_groups - *Пользователи → Группы пользователей*;<br>administration.user_roles - *Пользователи → Роли пользователей*;<br>administration.users - *Пользователи → Пользователи*;<br>administration.api_tokens - *Пользователи → API tokens*;<br>administration.authentication - *Пользователи → Аутентификация*;<br>administration.general - *Администрирование → Общие*;<br>administration.audit_log - *Администрирование → Журнал аудита*;<br>administration.housekeeping - *Администрирование → Housekeeping*;<br>administration.proxy_groups - *Администрирование → Группы* |

| Property | Type | Description |
|---|---|---|
| status | integer | Включен ли доступ к UI element.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |

Сервис

| Property | Type | Description |
|---|---|---|
| serviceid | ID | ID службы.<br>Поведение свойства:<br>- *required* |

Тег сервиса

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега.<br><br>Если указана пустая строка, тег сервиса не будет использоваться для сопоставления сервиса.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение тега.<br><br>Если значение не указано или указана пустая строка, для сопоставления сервиса будет использоваться только имя тега. |

Модуль

Объект module имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| moduleid | ID | ID модуля.<br><br>Поведение свойства:<br>- *required* |
| status | integer | Включен ли доступ к модулю.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |

Действие

Объект action имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя действия.<br><br>Возможные значения, если `type` объекта Role задан как ”User”, ”Admin” или ”Super admin”:<br>`edit_dashboards` - Создание и редактирование панелей;<br>`edit_maps` - Создание и редактирование карт;<br>`add_problem_comments` - Добавление комментариев к проблемам;<br>`change_severity` - Изменение серьезности проблемы;<br>`acknowledge_problems` - Подтверждение проблем;<br>`suppress_problems` - Подавление проблем;<br>`close_problems` - Закрытие проблем;<br>`execute_scripts` - Выполнение скриптов;<br>`manage_api_tokens` - Управление API-токенами;<br>`change_problem_ranking` - Изменение ранга проблемы с причины на симптом и наоборот;<br>`edit_own_media` - Разрешить создавать/редактировать собственные медиа.<br><br>Возможные значения, если `type` объекта Role задан как ”Admin” или ”Super admin”:<br>`edit_maintenance` - Создание и редактирование обслуживаний;<br>`manage_scheduled_reports` - Управление запланированными отчетами,<br>`manage_sla` - Управление SLA.<br><br>Возможные значения, если `type` объекта Role задан как ”User” или ”Admin”:<br>`invoke_execute_now` - позволяет выполнять проверки элементов данных для пользователей, у которых есть только права на чтение на узле сети.<br><br>Возможные значения, если `type` объекта Role задан как ”Super admin”:<br>`edit_user_media` - Разрешить создавать/редактировать медиа для пользователей.<br><br>Поведение свойства:<br>- *required* |
| status | integer | Включен ли доступ для выполнения действия.<br><br>Возможные значения:<br>0 - Отключено;<br>1 - *(по умолчанию)* Включено. |

**role.create**

Описание

`object role.create( / roles)`

Этот метод позволяет создавать новые роли.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Роли для создания.

В дополнение к стандартным свойствам роли метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| rules | array | Правила роли, которые будут созданы для роли. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных ролей, указанных в свойстве `roleids`. Порядок возвращаемых ID совпадает с порядком переданных ролей.

Примеры

Создание роли

Создайте роль с типом "User" и запретом доступа к двум элементам интерфейса.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

См. также

- Правила роли
- Элемент интерфейса
- Модуль
- Действие

Источник

CRole::create() в *ui/include/classes/api/services/CRole.php*.

**role.delete**

Описание

`object role.delete( roleids)`

Этот метов позволяет удалить роли.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`( )` ID удаляемых ролей.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных ролей, указанных в свойстве `roleids`.

Примеры

Удаление нескольких ролей пользователей

Удаление двух ролей пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "role.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CRole::delete() в *ui/include/classes/api/services/CRole.php*.

**role.get**

Описание

`integer/array role.get( )`

Этот метод позволяет получать роли в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| roleids | ID/array | Возвращать только роли с указанными ID. |
| selectRules | query | Возвращать свойство `rules` с правилами роли. |
| selectUsers | query | Возвращать свойство `users` с пользователями, которым назначена роль.<br><br>См. `user.get` для ограничений, основанных на типе пользователя. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `roleid`, `name`. |
| countOutput | boolean | Эти параметры описаны в reference commentary. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение данных роли

Получение данных роли ”Super admin role” и ее правил доступа.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "role.get",
    "params": {
        "output": "extend",
        "selectRules": "extend",
        "roleids": "3"
    },
    "id": 1
}
```

Ответ:

```
{
    "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",
        "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"
            }
        ],
        "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
}
```


См. также

- Правила роли
- Пользователь

Источник

CRole::get() в *ui/include/classes/api/services/CRole.php*.

**role.update**

Описание

`object role.update(  /  roles)`

Этот метод позволяет обновлять существующие роли.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства роли, которые необходимо обновить.

Для каждой роли должно быть определено свойство `roleid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам роли метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| rules | array | Правила доступа, которые нужно обновить для роли. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновлённых ролей, указанных в свойстве `roleids`.

Примеры

Отключение возможности выполнять скрипты

Обновить роль с ID ”5”, отключив возможность выполнять скрипты.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "actions": [
                    {
                        "name": "execute_scripts",
                        "status": "0"
                    }
                ]
            }
        }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Ограничение доступа к API

Обновить роль с ID ”5”, запретив вызов любых методов ”create”, ”update” или ”delete”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "api.access": "1",
                "api.mode": "0",
                "api": ["*.create", "*.update", "*.delete"]
            }
        }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CRole::update() в *ui/include/classes/api/services/CRole.php*.

### Сервис

Этот класс предназначен для работы с ИТ-инфраструктурой и бизнес-сервисами.

Ссылки на объекты:

- Сервис
- Правило статуса
- Тег сервиса
- Тревога сервиса
- Тег проблемы

Доступные методы:

- service.create - создать новые сервисы
- service.delete - удалить сервисы
- service.get - получить сервисы
- service.update - обновить сервисы

### Объект service

Следующие объекты напрямую связаны с API `service`.

Сервис

Объект service имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| serviceid | ID | ID сервиса.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| algorithm | integer | Правило расчета статуса. Применимо только при наличии дочерних сервисов.<br><br>Возможные значения:<br>0 - установить статус OK;<br>1 - наиболее критичный, если у всех дочерних сервисов есть проблемы;<br>2 - наиболее критичный из дочерних сервисов.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Имя сервиса.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| sortorder | integer | Позиция сервиса, используемая для сортировки.<br><br>Возможные значения: 0-999.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| weight | integer | Вес сервиса.<br><br>Возможные значения: 0-1000000.<br><br>По умолчанию: 0. |

| Property | Type | Description |
|---|---|---|
| propagation_rule | integer | Правило распространения статуса.<br><br>Возможные значения:<br>0 - *(по умолчанию)* распространять статус сервиса как есть - без каких-либо изменений;<br>1 - увеличить распространяемый статус на заданное значение `propagation_value` (на 1-5 уровней серьезности);<br>2 - уменьшить распространяемый статус на заданное значение `propagation_value` (на 1-5 уровней серьезности);<br>3 - игнорировать этот сервис - статус вообще не распространяется на родительский сервис;<br>4 - установить фиксированный статус сервиса с использованием заданного `propagation_value`.<br><br>Поведение свойства:<br>- *обязательно*, если задан `propagation_value` |
| propagation_value | integer | Значение распространения статуса.<br><br>Возможные значения, если `propagation_rule` установлено в "0" или "3":<br>0 - Не классифицировано.<br><br>Возможные значения, если `propagation_rule` установлено в "1" или "2":<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Чрезвычайная.<br><br>Возможные значения, если `propagation_rule` установлено в "4":<br>-1 - OK;<br>0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Чрезвычайная.<br><br>Поведение свойства:<br>- *обязательно*, если задан `propagation_rule` |
| status | integer | Находится ли сервис в состоянии OK или в состоянии проблемы.<br><br>Если сервис находится в состоянии проблемы, `status` равен либо серьезности наиболее критичной проблемы, либо наивысшему статусу дочернего сервиса, находящегося в состоянии проблемы.<br><br>Если сервис находится в состоянии OK, `status` равен -1.<br><br>Поведение свойства:<br>- *только для чтения* |
| description | string | Описание сервиса. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных сервисов с уже существующими. Генерируется автоматически, если не указан. |
| created_at | integer | Unix timestamp, когда сервис был создан. |

| Property | Type | Description |
|---|---|---|
| readonly | integer | Доступ к сервису.<br><br>Возможные значения:<br>0 - Чтение и запись;<br>1 - Только чтение.<br><br>Поведение свойства:<br>- *только для чтения* |

Правило статуса

Объект правила статуса имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Условие для установки статуса (New status).<br><br>Возможные значения:<br>0 - если как минимум (N) дочерних сервисов имеют статус (Status) или выше;<br>1 - если как минимум (N%) дочерних сервисов имеют статус (Status) или выше;<br>2 - если менее (N) дочерних сервисов имеют статус (Status) или ниже;<br>3 - если менее (N%) дочерних сервисов имеют статус (Status) или ниже;<br>4 - если вес дочерних сервисов со статусом (Status) или выше составляет не менее (W);<br>5 - если вес дочерних сервисов со статусом (Status) или выше составляет не менее (N%);<br>6 - если вес дочерних сервисов со статусом (Status) или ниже меньше (W);<br>7 - если вес дочерних сервисов со статусом (Status) или ниже меньше (N%).<br><br>Где:<br>- N (W) — `limit_value`;<br>- (Status) — `limit_status`;<br>- (New status) — `new_status`.<br><br>Поведение свойства:<br>- *required* |
| limit_value | integer | Предельное значение.<br><br>Возможные значения:<br>- для N и W: 1-100000;<br>- для N%: 1-100.<br><br>Поведение свойства:<br>- *required* |

| Property | Type | Description |
|---|---|---|
| limit_status | integer | Предельный статус.<br><br>Возможные значения:<br>-1 - OK;<br>0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>Поведение свойства:<br>- *required* |
| new_status | integer | Значение нового статуса.<br><br>Возможные значения:<br>0 - Не классифицировано;<br>1 - Информация;<br>2 - Предупреждение;<br>3 - Средняя;<br>4 - Высокая;<br>5 - Катастрофа.<br><br>Поведение свойства:<br>- *required* |

Тег сервиса

Объект тега сервиса имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега сервиса.<br>Поведение свойства:<br>- *required* |
| value | string | Значение тега сервиса. |

Сервисная тревога

**Note:**
Сервисные тревоги нельзя напрямую создавать, обновлять или удалять через Zabbix API.

Объекты сервисной тревоги представляют изменение состояния сервиса. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Время, когда произошло изменение состояния сервиса. |
| value | integer | Статус сервиса.<br><br>См. свойство `status` сервиса для списка возможных значений. |

Тег проблемы

Теги проблемы позволяют связывать услуги с событиями проблемы. Объект тега проблемы имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега проблемы.<br><br>Property behavior:<br>- *required* |
| operator | integer | Оператор условия сопоставления.<br><br>Возможные значения:<br>0 - *(default)* Равно;<br>2 - Содержит. |
| value | string | Значение тега проблемы. |

**service.create**

Описание

`object service.create( / services)`

Этот метод позволяет создавать новые услуги.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` создаваемые services.

В дополнение к стандартным свойствам service, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| children | array | Дочерние services, которые будут привязаны к service.<br><br>У дочерних services должно быть определено только свойство `serviceid`. |
| parents | array | Родительские services, которые будут привязаны к service.<br><br>У родительских services должно быть определено только свойство `serviceid`. |
| tags | array | Service tags, которые будут созданы для service. |
| problem_tags | array | Problem tags, которые будут созданы для service. |
| status_rules | array | Status rules, которые будут созданы для service. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных сервисов в свойстве `serviceids`. Порядок возвращаемых идентификаторов соответствует порядку переданных сервисов.

Примеры

Создание сервиса

Создайте сервис, который будет переведен в состояние проблемы, если хотя бы у одного дочернего элемента возникнет проблема.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "service.create",
    "params": {
        "name": "Server 1",
        "algorithm": 1,
        "sortorder": 1
```

```
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Источник

CService::create() в *ui/include/classes/api/services/CService.php*.

**service.delete**

Описание

`object service.delete(    serviceIds)`

Этот метод позволяет удалять услуги.

**Note:**
Этот метод доступен пользователям любого типа.  Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых услуг.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных услуг, указанных в свойстве `serviceids`.

Примеры

Удаление нескольких сервисов

Удаление двух сервисов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "service.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4",
            "5"
        ]
    },
```

```
    "id": 1
}
```

Источник

CService::delete() в *ui/include/classes/api/services/CService.php*.

**service.get**

Описание

`integer/array service.get(          )`

Метод позволяет получать услуги по заданным параметры.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| serviceids | ID/array | Возвращать только услуги с указанными ID. |
| parentids | ID/array | Возвращать только услуги, связанные с указанными родительскими услугами. |
| deep_parentids | flag | Возвращать все прямые и косвенные дочерние услуги. Используется вместе с `parentids`. |
| childids | ID/array | Возвращать только услуги, связанные с указанными дочерними услугами. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И;<br>2 - Или. |
| tags | object/array | Возвращать только услуги с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все услуги.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| problem_tags | object/array | Возвращать только услуги с указанными тегами проблемы.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все услуги.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| without_problem_tags | flag | Возвращать только услуги без тегов проблемы. |
| slaids | ID/array | Возвращать только услуги, связанные с указанными SLA. |

| Parameter | Type | Description |
|---|---|---|
| selectChildren | query | Возвращать свойство `children` с дочерними услугами.<br><br>Поддерживает `count`. |
| selectParents | query | Возвращать свойство `parents` с родительскими услугами.<br><br>Поддерживает `count`. |
| selectTags | query | Возвращать свойство `tags` с тегами услуги.<br><br>Поддерживает `count`. |
| selectProblemEvents | query | Возвращать свойство `problem_events` с массивом объектов событий проблемы.<br><br>Объект события проблемы имеет следующие свойства:<br>`eventid` - (`ID`) ID события;<br>`severity` - (`string`) Текущая серьезность события;<br>`name` - (`string`) Имя разрешенного события.<br><br>Поддерживает `count`. |
| selectProblemTags | query | Возвращать свойство `problem_tags` с тегами проблемы.<br><br>Поддерживает `count`. |
| selectStatusRules | query | Возвращать свойство `status_rules` с правилами статуса.<br><br>Поддерживает `count`. |
| selectStatusTimeline | object/array | Возвращать свойство `status_timeline`, содержащее изменения состояния услуги за указанные периоды.<br><br>Формат `[{"period_from": "<period_from>", "period_to": "<period_to>"}, ...]` - `period_from` является начальной датой (включительно; целочисленная временная метка), а `period_to` - конечной датой (исключительно; целочисленная временная метка) для интересующего вас периода.<br><br>Возвращает массив записей, содержащих свойство `start_value` и массив `alarms` для изменений состояния в указанных периодах.<br>Поддерживает `count`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `serviceid`, `name`, `status`, `sortorder`, `created_at`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение всех сервисов

Получите все данные обо всех сервисах и их зависимостях.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "service.get",
    "params": {
        "output": "extend",
        "selectChildren": "extend",
        "selectParents": ["serviceid", "name"]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Источник

CService::get() в *ui/include/classes/api/services/CService.php*.

**service.update**

Описание

`object service.update(  /  services)`

Этот метод позволяет обновлять существующие сервисы.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) свойства сервиса, которые необходимо обновить.

Для каждого сервиса должно быть определено свойство serviceid, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Помимо стандартных свойств сервиса, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| children | array | Дочерние сервисы, которые заменят текущие дочерние сервисы.<br><br>У дочерних сервисов должно быть определено только свойство serviceid. |
| parents | array | Родительские сервисы, которые заменят текущие родительские сервисы.<br><br>У родительских сервисов должно быть определено только свойство serviceid. |
| tags | array | Теги сервиса, которые заменят текущие теги сервиса. |
| problem_tags | array | Теги проблемы, которые заменят текущие теги проблемы. |
| status_rules | array | Правила статуса, которые заменят текущие правила статуса. |

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы обновленных служб в свойстве serviceids.

Примеры

Установка родителя для сервиса

Сделать сервис с ID ”3” родительским для сервиса с ID ”5”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "5",
        "parents": [
            {
                "serviceid": "3"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Добавление запланированного периода недоступности

Добавьте период недоступности для сервиса с ID ”4”, запланированный еженедельно с понедельника 22:00 до вторника 10:00.

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "service.update",
    "params": {
        "serviceid": "4",
        "times": [
            {
                "type": "1",
                "ts_from": "165600",
                "ts_to": "201600"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4"
        ]
    },
    "id": 1
}
```

Источник

CService::update() в *ui/include/classes/api/services/CService.php*.

#### Скрипт

Этот класс предназначен для работы со скриптами.

Ссылки на объекты:

- Script
  - Параметры вебхука
- Debug
  - Запись журнала

Доступные методы:

- script.create - создать новые скрипты
- script.delete - удалить скрипты
- script.execute - выполнить скрипты
- script.get - получить скрипты
- script.getscriptsbyevents - получить скрипты для событий
- script.getscriptsbyhosts - получить скрипты для узлов сети
- script.update - обновить скрипты

#### Объект script

Следующие объекты напрямую связаны с API `script`.

Скрипт

Объект script имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| scriptid | ID | ID скрипта.<br><br>Property behavior:<br>- *только для чтения*<br>- *обязательно* для операций обновления |

| Property | Type | Description |
|---|---|---|
| name | string | Имя скрипта.<br><br>Property behavior:<br>- *обязательно* для операций создания |
| type | integer | Тип скрипта.<br><br>Допустимые значения, если `scope` установлен в ”action operation”:<br>0 - Script;<br>1 - IPMI;<br>2 - SSH;<br>3 - TELNET;<br>5 - Webhook.<br><br>Допустимые значения, если `scope` установлен в ”manual host action” или ”manual event action”:<br>6 - URL.<br><br>Property behavior:<br>- *обязательно* для операций создания |
| command | string | Команда для выполнения.<br><br>Property behavior:<br>- *обязательно*, если `type` установлен в ”Script”, ”IPMI”, ”SSH”, ”TELNET” или ”Webhook” |
| scope | integer | Область применения скрипта.<br><br>Допустимые значения:<br>1 - action operation;<br>2 - manual host action;<br>4 - manual event action.<br><br>Property behavior:<br>- *обязательно* для операций создания |
| execute_on | integer | Где запускать скрипт.<br><br>Допустимые значения:<br>0 - запуск на агенте Zabbix;<br>1 - запуск на сервере Zabbix. *Поддерживается* только если на сервере Zabbix включено выполнение глобальных скриптов;<br>2 - *(по умолчанию)* запуск на сервере Zabbix или прокси.<br><br>Property behavior:<br>- *поддерживается*, если `type` установлен в ”Script” |
| menu_path | string | Папки, разделенные слешем, которые формируют навигацию в стиле меню во веб-интерфейсе при нажатии на узел сети или событие.<br><br>Property behavior:<br>- *поддерживается*, если `scope` установлен в ”manual host action” или ”manual event action” |
| authtype | integer | Метод аутентификации, используемый для скрипта типа SSH.<br><br>Допустимые значения:<br>0 - пароль;<br>1 - публичный ключ.<br><br>Property behavior:<br>- *поддерживается*, если `type` установлен в ”SSH” |
| username | string | Имя пользователя, используемое для аутентификации.<br><br>Property behavior:<br>- *обязательно*, если `type` установлен в ”SSH” или ”TELNET” |

| Property | Type | Description |
|---|---|---|
| password | string | Пароль, используемый для скриптов SSH с аутентификацией по паролю и для скриптов TELNET.<br><br>Property behavior:<br>- *поддерживается*, если `type` установлен в ”SSH” и `authtype` установлен в ”password”, или если `type` установлен в ”TELNET” |
| publickey | string | Имя файла публичного ключа, используемого для скриптов SSH с аутентификацией по публичному ключу.<br><br>Property behavior:<br>- *обязательно*, если `type` установлен в ”SSH” и `authtype` установлен в ”public key” |
| privatekey | string | Имя файла закрытого ключа, используемого для скриптов SSH с аутентификацией по публичному ключу.<br><br>Property behavior:<br>- *обязательно*, если `type` установлен в ”SSH” и `authtype` установлен в ”public key” |
| port | string | Номер порта, используемый для скриптов SSH и TELNET.<br><br>Property behavior:<br>- *поддерживается*, если `type` установлен в ”SSH” или ”TELNET” |
| groupid | ID | ID группы узлов сети, на которых можно запускать скрипт.<br><br>Если установлено значение ”0”, скрипт будет доступен для всех групп узлов сети.<br><br>По умолчанию: 0. |
| usrgrpid | ID | ID группы пользователей, которым будет разрешено запускать скрипт.<br><br>Если установлено значение ”0”, скрипт будет доступен для всех групп пользователей.<br><br>По умолчанию: 0.<br><br>Property behavior:<br>- *поддерживается*, если `scope` установлен в ”manual host action” или ”manual event action” |
| host_access | integer | Права на узел сети, необходимые для запуска скрипта.<br><br>Допустимые значения:<br>2 - *(по умолчанию)* чтение;<br>3 - запись.<br><br>Property behavior:<br>- *поддерживается*, если `scope` установлен в ”manual host action” или ”manual event action” |
| confirmation | string | Текст всплывающего окна подтверждения.<br>Всплывающее окно появится при попытке запустить скрипт из веб-интерфейса Zabbix.<br><br>Property behavior:<br>- *поддерживается*, если `scope` установлен в ”manual host action” или ”manual event action” |

| Property | Type | Description |
|---|---|---|
| timeout | string | Тайм-аут выполнения скрипта webhook в секундах.<br>Поддерживаются суффиксы единиц времени (например, 30s, 1m).<br><br>Допустимые значения: 1-60s.<br><br>По умолчанию: 30s.<br><br>Property behavior:<br>- *обязательно*, если `type` установлен в ”Webhook” |
| parameters | array | Массив входных параметров webhook.<br><br>Property behavior:<br>- *поддерживается*, если `type` установлен в ”Webhook” |
| description | string | Описание скрипта. |
| url | string | Пользовательский URL.<br><br>Property behavior:<br>- *обязательно*, если `type` установлен в ”URL” |
| new_window | integer | Открывать URL в новом окне.<br><br>Допустимые значения:<br>0 - Нет;<br>1 - *(по умолчанию)* Да.<br><br>Property behavior:<br>- *поддерживается*, если `type` установлен в ”URL” |
| manualinput | integer | Указывает, принимает ли скрипт ввод, предоставленный пользователем.<br><br>Допустимые значения:<br>0 - *(по умолчанию)* Отключено;<br>1 - Включено;<br><br>Property behavior:<br>- *поддерживается*, если `scope` установлен в ”manual host action” или ”manual event action” |
| manualinput_prompt | string | Текст запроса для ручного ввода.<br><br>Property behavior:<br>- *обязательно*, если `manualinput` установлен в ”Enabled” |
| manualinput_validator | string | Строковое поле, используемое для проверки введенных пользователем данных. Строка состоит либо из регулярного выражения, либо из набора значений, разделенных запятыми.<br><br>Property behavior:<br>- *обязательно*, если `manualinput` установлен в ”Enabled” |
| manualinput_validator_type | integer | Определяет тип ожидаемого пользовательского ввода.<br><br>Допустимые значения:<br>0 - *(по умолчанию)* String. Указывает, что manualinput_validator следует трактовать как регулярное выражение;<br>1 - List. Указывает, что manualinput_validator следует трактовать как список возможных значений ввода, разделенных запятыми.<br><br>Property behavior:<br>- *поддерживается*, если `manualinput` установлен в ”Enabled” |

| Property | Type | Description |
|---|---|---|
| manualinput_default_value | string | Значение по умолчанию для автоматического заполнения пользовательского ввода.<br><br>Property behavior:<br>- *поддерживается*, если `manualinput_validator_type` установлен в "String" |

Параметры вебхука

Параметры, передаваемые в скрипт вебхука при его вызове, имеют следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя параметра.<br>Поведение свойства:<br>- *required* |
| value | string | Значение параметра. Поддерживает макросы. |

Отладка

Отладочная информация о выполненном скрипте вебхука. Объект debug имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| logs | array | Массив записей журнала. |
| ms | string | Длительность выполнения скрипта в миллисекундах. |

Запись журнала

Объект записи журнала имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| level | integer | Уровень журнала. |
| ms | string | Время, прошедшее в миллисекундах с момента запуска скрипта до добавления записи журнала. |
| message | string | Сообщение журнала. |

**script.create**

Описание

`object script.create( / scripts)`

Этот метод позволяет создавать новые сценарии.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( / ) Создаваемые скрипты.

Этот метод принимает скрипты со стандартными свойствами скрипта.

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных скриптов, указанных в свойстве `scriptids`. Порядок возвращаемых ID совпадает с порядком переданных скриптов.

Примеры

Создание скрипта вебхука

Создайте скрипт вебхука, который отправляет HTTP-запрос во внешнюю службу.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "3"
        ]
    },
    "id": 1
}
```

Создание SSH-скрипта

Создайте SSH-скрипт с аутентификацией по публичному ключу, который можно выполнить на узле сети и который имеет контекстное меню.

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "5"
        ]
    },
    "id": 1
}
```

Создание пользовательского скрипта

Создайте пользовательский скрипт, который будет перезагружать сервер. Скрипту потребуется доступ на запись к узлу сети, и он запросит у пользователя ручной ввод. После успешной отправки ввода скрипт отобразит сообщение с подтверждением во веб-интерфейсе.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "4"
        ]
    },
    "id": 1
}
```

Создание скрипта типа URL

Создайте скрипт типа URL для области узла сети, который остается в том же окне и имеет текст подтверждения.

Request:

```
{
    "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": "Edit host {HOST.NAME}?",
        "new_window": 0
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "56"
        ]
    },
    "id": 1
}
```

Создание скрипта типа URL с ручным вводом

Создайте скрипт типа URL для области событий, который открывается в новом окне и поддерживает ручной ввод.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "URL script with manual input",
        "type": 6,
        "scope": 4,
        "url": "http://zabbix/ui/zabbix.php?action={MANUALINPUT}",
        "new_window": 1,
        "manualinput": 1,
        "manualinput_prompt": "Select a page to open:",
        "manualinput_validator": "dashboard.view,script.list,actionlog.list",
        "manualinput_validator_type": 1
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "57"
        ]
    },
    "id": 1
}
```

Источник

CScript::create() в *ui/include/classes/api/services/CScript.php*.

**script.delete**

Описание

`object script.delete(    scriptIds)`

Этот метод позволяет удалять скрипты.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID удаляемых скриптов.

Возвращаемые значения

( ) Возвращает объект, который содержит ID удаленных скриптов, указанных в свойстве `scriptids`.

Примеры

Удаление нескольких скриптов

Удаление двух скриптов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.delete",
    "params": [
        "3",
        "4"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "3",
            "4"
        ]
    },
    "id": 1
}
```

Источник

CScript::delete() в *ui/include/classes/api/services/CScript.php*.

**script.execute**

Описание

`object script.execute( )`

Этот метод позволяет запустить скрипт на хосте или событии. За исключением скриптов типа URL. Они не являются исполняемыми.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие ID скрипта для запуска, а также ID узла сети или ID события и значение manualinput.

| Parameter | Type | Description |
|---|---|---|
| scriptid | ID | ID скрипта для запуска.<br><br>Поведение параметра:<br>- *required* |
| hostid | ID | ID узла сети, на котором нужно запустить скрипт.<br><br>Поведение параметра:<br>- *required* if `eventid` is not set |
| eventid | ID | ID события, на котором нужно запустить скрипт.<br><br>Поведение параметра:<br>- *required* if `hostid` is not set |
| manualinput | string | Значение, предоставленное пользователем, с которым запускается скрипт, с подстановкой макроса {MANUALINPUT}. |

Возвращаемые значения

`(object)` Возвращает результат выполнения скрипта.

| Property | Type | Description |
|---|---|---|
| response | string | Был ли скрипт выполнен успешно.<br><br>Возможное значение - `success`. |
| value | string | Вывод скрипта. |
| debug | object | Содержит объект debug, если выполняется скрипт вебхука. Для других типов скриптов содержит пустой объект. |

Примеры

Запуск вебхука

Запустите вебхук, который отправляет HTTP-запрос во внешнюю службу.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "4",
        "hostid": "30079"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Запуск пользовательского скрипта

Запустите скрипт ”ping” на узле сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "1",
        "hostid": "30079"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Запуск пользовательского скрипта с ручным вводом

Запустите скрипт ”ping” с командой ”ping -c {MANUALINPUT} {HOST.CONN}; case $?  in [01]) true;; *) false;; esac” на узле сети.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "7",
        "hostid": "30079",
        "manualinput": "2"
    },
    "id": 1
}
```

Response:

```
{
    "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
}
```

Источник

CScript::execute() в *ui/include/classes/api/services/CScript.php*.

**script.get**

Описание

`integer/array script.get(  )`

Метод позволяет получать скрипты по заданным параметрам.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groupids | ID/array | Возвращать только скрипты, которые можно запускать на указанных группах узлов сети. |
| hostids | ID/array | Возвращать только скрипты, которые можно запускать на указанных узлах сети. |
| scriptids | ID/array | Возвращать только скрипты с указанными ID. |
| usrgrpids | ID/array | Возвращать только скрипты, которые могут запускаться пользователями из указанных групп пользователей. |
| selectHostGroups | query | Возвращать свойство `hostgroups` с группами узлов сети, на которых можно запускать скрипт. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, на которых можно запускать скрипт. |
| selectActions | query | Возвращать свойство `actions` с действиями, с которыми связан скрипт. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `scriptid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(  /  )` Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получить все скрипты

Получить все настроенные скрипты.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "       {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": "          zabbix {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": "                    :",
            "manualinput_validator": "dashboard.view,script.list,actionlog.list",
            "manualinput_validator_type": "1",
            "parameters": []
        }
    ],
    "id": 1
}
```


См. также

- Узел сети
- Группа узлов сети

Источник

CScript::get() в *ui/include/classes/api/services/CScript.php*.

#### script.getscriptsbyevents

Описание

`object script.getscriptsbyevents(          )`

Этот метод позволяет получить все доступные скрипты для данного события или конкретный скрипт, если указан ID скрипта. Если предоставлено значение manualinput, оно заменяет макрос {MANUALINPUT} на указанное значение.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Метод принимает объект или массив объектов со следующими параметрами.

| Parameter | Type | Description |
|---|---|---|
| eventid | ID | ID события, для которого нужно вернуть скрипты.<br>Должен быть уникальным.<br><br>Поведение параметра:<br>- *required* |
| scriptid | ID | ID скрипта, который нужно вернуть. |
| manualinput | string | Значение макроса {MANUALINPUT}, заданного пользователем. |

Возвращаемые значения

`(object)` Возвращает объект, в котором идентификаторы событий указаны как свойства, а массивы доступных скриптов — как значения. Если указан идентификатор скрипта, соответствующее значение представляет собой массив, содержащий этот конкретный скрипт.

**Note:**
Метод автоматически развернет макросы в тексте `confirmation`, тексте `manualinput prompt` и `url`.
Если указан параметр manualinput, макрос {MANUALINPUT} будет подставлен в указанное значение.

Примеры

Получение скриптов по ID событий

Получить все скрипты, доступные для событий ”632” и ”614”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
         "eventid":  "632"
      },
      {
         "eventid":  "614"
      }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "632": [
            {
                "scriptid": "3",
                "name": "                    ",
                "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": "          Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "      ,                  *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "     Zabbix        :",
            "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": "                    ",
            "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": "          Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "      ,                  *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "4",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "     Zabbix        :",
            "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
}
```


Получение конкретного скрипта со значением manualinput.

Получить скрипт с ID ”4” для события ”632” со значением manualinput ”dashboard.view”.

Запрос:


```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
        "eventid":  "632",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```


Ответ:


```
{
    "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
}
```


Источник

CScript::getScriptsByEvents() в *ui/include/classes/api/services/CScript.php*.

**script.getscriptsbyhosts**

Описание

`object script.getscriptsbyhosts(          )`

Этот метод позволяет получить все доступные скрипты для данного хоста или конкретный скрипт, если указан ID скрипта. Если предоставлено значение manualinput, оно заменяет макрос {MANUALINPUT} на указанное значение.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Метод принимает объект или массив объектов со следующими параметрами.

| Parameter | Type | Description |
|---|---|---|
| hostid | ID | ID узла сети, для которого нужно вернуть скрипты.<br>Должен быть уникальным.<br><br>Поведение параметра:<br>- *required* |
| scriptid | ID | ID скрипта, который нужно вернуть. |
| manualinput | string | Значение макроса {MANUALINPUT}, предоставленного пользователем. |

Возвращаемые значения

`(object)` Возвращает объект, в котором идентификаторы узлов сети являются свойствами, а массивы доступных скриптов — значениями. Если указан идентификатор скрипта, соответствующее значение представляет собой массив, содержащий конкретный скрипт.

> **Note:**
> Метод автоматически развернет макросы в тексте `confirmation`, тексте `manualinput prompt` и `url`.
> Если указан параметр manualinput, макрос {MANUALINPUT} будет подставлен в указанное значение.

Примеры

Получение скриптов по ID узлов сети

Получить все скрипты, доступные на узлах сети ”30079” и ”30073”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
       {
          "hostid":  "30079"
       },
       {
          "hostid":  "30073"
       }
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "30079": [
            {
                "scriptid": "3",
                "name": "Detect operating system",
                "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "7",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            },
            {
                "scriptid": "1",
                "name": "Ping",
                "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "0",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "2",
                "port": "",
```

```
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Open Zabbix page",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "      ,                  *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "     Zabbix        :",
            "manualinput_validator_type": "0",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "2",
            "name": "Traceroute",
            "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "",
            "type": "0",
            "execute_on": "1",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
```

```
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            }
        ],
        "30073": [
            {
                "scriptid": "3",
                "name": "Detect operating system",
                "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "7",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
                "url": "",
                "new_window": "1",
                "manualinput": "0",
                "manualinput_prompt": "",
                "manualinput_validator_type": "0",
                "manualinput_validator": "",
                "manualinput_default_value": "",
                "parameters": []
            },
            {
                "scriptid": "1",
                "name": "Ping",
                "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
                "host_access": "2",
                "usrgrpid": "0",
                "groupid": "0",
                "description": "",
                "confirmation": "",
                "type": "0",
                "execute_on": "1",
                "timeout": "30s",
                "scope": "2",
                "port": "",
                "authtype": "0",
                "username": "",
                "password": "",
                "publickey": "",
                "privatekey": "",
                "menu_path": "",
```

```
            "url": "",
            "new_window": "1",
            "manualinput": "0",
            "manualinput_prompt": "",
            "manualinput_validator_type": "0",
            "manualinput_validator": "",
            "manualinput_default_value": "",
            "parameters": []
        },
        {
            "scriptid": "4",
            "name": "Open Zabbix page",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "      ,                  *UNKNOWN*?",
            "type": "6",
            "execute_on": "2",
            "timeout": "30s",
            "scope": "2",
            "port": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "menu_path": "",
            "url": "http://localhost/ui/zabbix.php?action=*UNKNOWN*",
            "new_window": "1",
            "manualinput": "1",
            "manualinput_prompt": "     Zabbix        :",
            "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
}
```

Получение конкретного скрипта со значением manualinput.

Получить скрипт с ID "4" на узле сети "30079" со значением manualinput "dashboard.view".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
      {
        "hostid":  "30079",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Источник

CScript::getScriptsByHosts() в *ui/include/classes/api/services/CScript.php*.

**script.update**

Описание

`object script.update(  /  scripts)`

Этот метод позволяет обновлять существующие скрипты.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства скрипта, которые необходимо обновить.

Свойство `scriptid` должно быть определено для каждого скрипта, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений. Исключение составляет изменение свойства `type` с 5 (Webhook) на другое значение: свойство `parameters` будет очищено.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных скриптов в свойстве `scriptids`.

Примеры

Изменение команды скрипта

Измените команду скрипта на "/bin/ping -c 10 {HOST.CONN} 2>&1".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
        "scriptid": "1",
        "command": "/bin/ping -c 10 {HOST.CONN} 2>&1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Изменение команды скрипта и добавление ручного ввода

Измените команду скрипта на "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
      "scriptid": "1",
      "command": "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1",
      "manualinput": "1",
      "manualinput_prompt": "           ICMP-    ,                          ping",
      "manualinput_validator": "^(?:[1-9]|10)$",
      "manualinput_validator_type": "0",
      "manualinput_default_value": "10"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Источник

CScript::update() в *ui/include/classes/api/services/CScript.php*.

#### Событие

Этот класс предназначен для работы с событиями.

Ссылки на объекты:

- Событие
- Тег события
- URL типа медиа

Доступные методы:

- event.get - получить события
- event.acknowledge - подтвердить события

#### Объект события

Следующие объекты напрямую связаны с API `event`.

Событие

> **Note:**
> События создаются сервером Zabbix и не могут быть изменены через API.

Объект события имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| eventid | ID | ID события. |

| Property | Type | Description |
|---|---|---|
| source | integer | Тип события.<br><br>Возможные значения:<br>0 - событие, созданное триггером;<br>1 - событие, созданное правилом обнаружения;<br>2 - событие, созданное автоматической регистрацией активного агента;<br>3 - внутреннее событие;<br>4 - событие, созданное при обновлении статуса сервиса. |
| object | integer | Тип объекта, связанного с событием.<br><br>Возможные значения, если `source` установлено в ”event created by a trigger”:<br>0 - триггер.<br><br>Возможные значения, если `source` установлено в ”event created by a discovery rule”:<br>1 - обнаруженный узел сети;<br>2 - обнаруженный сервис.<br><br>Возможные значения, если `source` установлено в ”event created by active agent autoregistration”:<br>3 - автоматически зарегистрированный узел сети.<br><br>Возможные значения, если `source` установлено в ”internal event”:<br>0 - триггер;<br>4 - элемент данных;<br>5 - правило LLD.<br><br>Возможные значения, если `source` установлено в ”event created on service status update”:<br>6 - сервис. |
| objectid | ID | ID связанного объекта. |
| acknowledged | integer | Было ли событие подтверждено. |
| clock | timestamp | Время создания события. |
| ns | integer | Наносекунды в момент создания события. |
| name | string | Разрешенное имя события. |
| value | integer | Состояние связанного объекта.<br><br>Возможные значения, если `source` установлено в ”event created by a trigger” или ”event created on service status update”:<br>0 - OK;<br>1 - проблема.<br><br>Возможные значения, если `source` установлено в ”event created by a discovery rule”:<br>0 - узел сети или сервис в рабочем состоянии;<br>1 - узел сети или сервис недоступен;<br>2 - узел сети или сервис обнаружен;<br>3 - узел сети или сервис потерян.<br><br>Возможные значения, если `source` установлено в ”internal event”:<br>0 - состояние ”нормальное”;<br>1 - состояние ”неизвестно” или ”не поддерживается”.<br><br>Поведение свойства:<br>- *supported*, если `source` установлено в ”event created by a trigger”, ”event created by a discovery rule”, ”internal event” или ”event created on service status update” |

| Property | Type | Description |
|---|---|---|
| severity | integer | Текущая серьезность события.<br><br>Возможные значения:<br>0 - не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа. |
| r_eventid | ID | ID события восстановления. |
| c_eventid | ID | ID события, которое было использовано для переопределения (закрытия) текущего события по правилу глобальной корреляции.<br>См. `correlationid`, чтобы определить точное правило корреляции.<br>Этот параметр определен только тогда, когда событие закрыто правилом глобальной корреляции. |
| cause_eventid | ID | ID события-причины. |
| correlationid | ID | ID правила корреляции, которое вызвало закрытие проблемы.<br>Этот параметр определен только тогда, когда событие закрыто правилом глобальной корреляции. |
| userid | ID | ID пользователя, который закрыл событие (если событие было закрыто вручную). |
| suppressed | integer | Подавлено ли событие.<br><br>Возможные значения:<br>0 - событие находится в нормальном состоянии;<br>1 - событие подавлено. |
| opdata | string | Операционные данные с раскрытыми макросами. |
| urls | array | Активные URL-адреса типа медиа. |

Тег события

Объект тега события имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега события. |
| value | string | Значение тега события. |

URL типа оповещения

Объект URL типа оповещения имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя URL, определенное для типа оповещения. |
| url | string | Значение URL, определенное для типа оповещения. |

Результаты будут содержать записи только для активных типов оповещения с включенным пунктом меню событий. Макрос, используемый в свойствах, будет раскрыт, но если одно из свойств содержит нераскрытый макрос, оба свойства будут исключены из результатов. Поддерживаемые макросы см. в разделе *Поддерживаемые макросы*.

**event.acknowledge**

Описание

`object event.acknowledge(object/array parameters)`

Этот метод позволяет обновлять события. Можно выполнить следующие действия обновления:

- Закрыть событие. Если событие уже разрешено, это действие будет пропущено.
- Подтвердить событие. Если событие уже подтверждено, это действие будет пропущено.

- Снять подтверждение с события. Если событие не подтверждено, это действие будет пропущено.
- Добавить сообщение.
- Изменить серьезность события. Если у события уже такая же серьезность, это действие будет пропущено.
- Подавить событие. Если событие уже подавлено, это действие будет пропущено.
- Снять подавление с события. Если событие не подавлено, это действие будет пропущено.
- Изменить ранг события на cause. Если событие уже имеет ранг cause, это действие будет пропущено.
- Изменить ранг события на symptom. Если событие уже имеет ранг symptom, это действие будет пропущено.

**Attention:**
Можно обновлять только события триггера.<br> Можно обновлять только проблемные события.<br> Для закрытия события или изменения его серьезности требуются права чтения/записи для триггера.<br> Чтобы закрыть событие, в триггере должно быть разрешено ручное закрытие.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода могут быть отозваны в настройках роли пользователя. Дополнительные сведения см. в разделе Роли пользователей.

Параметры

`(object/array)` Параметры, содержащие IDs событий и операций обновления, которые должны быть выполнены.

| Parameter | Type | Description |
|---|---|---|
| eventids | ID/array | IDs событий, для которых нужно выполнить подтверждение.<br><br>Поведение параметра:<br>- *required* |
| action | integer | Действие(я) обновления события.<br><br>Возможные значения битовой карты:<br>1 - закрыть проблему;<br>4 - добавить сообщение;<br>6 - подтвердить событие;<br>8 - изменить серьезность;<br>16 - снять подтверждение события;<br>32 - подавить событие;<br>64 - отменить подавление события;<br>128 - изменить ранг события на cause;<br>256 - изменить ранг события на symptom.<br><br>Это поле битовой маски; допустима любая сумма возможных значений битовой карты (например, 34 для подтверждения и подавления события).<br><br>Поведение параметра:<br>- *required* |
| cause_eventid | ID | ID cause-события.<br><br>Поведение параметра:<br>- *required* если `action` содержит бит ”change event rank to symptom” |
| message | string | Текст сообщения.<br><br>Поведение параметра:<br>- *required* если `action` содержит бит ”add message” |

| Parameter | Type | Description |
|---|---|---|
| severity | integer | Новая серьезность для событий.<br><br>Возможные значения:<br>0 - not classified;<br>1 - information;<br>2 - warning;<br>3 - average;<br>4 - high;<br>5 - disaster.<br><br>Поведение параметра:<br>- *required* если `action` содержит бит "change severity" |
| suppress_until | integer | Unix timestamp, до которого событие должно быть подавлено.<br><br>Если установлено в "0", подавление будет бессрочным.<br><br>Поведение параметра:<br>- *required* если `action` содержит бит "suppress event" |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных событий в свойстве `eventids`.

Примеры

Подтверждение события

Подтвердите одно событие и оставьте сообщение.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 6,
        "message": "Problem resolved."
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427"
        ]
    },
    "id": 1
}
```

Изменение серьезности события

Измените серьезность для нескольких событий и оставьте сообщение.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": ["20427", "20428"],
        "action": 12,
```

```
        "message": "Maintenance required to fix it.",
        "severity": 4
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427",
            "20428"
        ]
    },
    "id": 1
}
```

Подавление события на неопределенный срок

Подавить одно событие без времени истечения.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 32,
        "suppress_until": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427
        ]
    },
    "id": 1
}
```

Снятие подавления события

Снимите подавление с одного ранее подавленного события.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 64
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "eventids": [
            20427
        ]
    },
    "id": 1
}
```

Подтвердить и подавить событие

Подтвердите одно или несколько событий, добавьте сообщение и подавите их в том же запросе.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": [
            "20427",
            "20428"
        ],
        "action": 38,
        "message": "                                .",
        "suppress_until": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427,
            20428
        ]
    },
    "id": 1
}
```

Источник

CEvent::acknowledge() в *ui/include/classes/api/services/CEvent.php*.

**event.get**

Описание

`integer/array event.get(object parameters)`

Метод позволяет получать события в соответствии с заданными параметрами.

**Attention:**
Этот метод может возвращать события удаленной сущности, если эти события еще не были удалены housekeeper.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| eventids | ID/array | Возвращать только события с указанными ID. |
| groupids | ID/array | Возвращать только события, созданные объектами, которые принадлежат указанным группам узлов сети. |
| hostids | ID/array | Возвращать только события, созданные объектами, которые принадлежат указанным узлам сети. |
| objectids | ID/array | Возвращать только события, созданные указанными объектами. |
| source | integer | Возвращать только события с указанным типом.<br><br>Список поддерживаемых типов событий см. на странице объекта события.<br><br>По умолчанию: 0 - события триггера. |
| object | integer | Возвращать только события, созданные объектами указанного типа.<br><br>Список поддерживаемых типов объектов см. на странице объекта события.<br><br>По умолчанию: 0 - триггер. |
| acknowledged | boolean | Если задано `true`, возвращать только подтвержденные события. |
| action | integer | Возвращать только события, для которых были выполнены указанные действия обновления события. Для нескольких действий используйте сумму любых допустимых значений битовой карты в качестве битовой маски (например, 34 для подтверждения и подавления события). |
| action_userids | ID/array | Возвращать только события с указанными ID пользователей, выполнивших действия обновления события. |
| suppressed | boolean | Если задано `true`, возвращать только подавленные события. |
| symptom | boolean | Если задано `true`, возвращать только события-симптомы. |
| severities | integer/array | Возвращать только события с указанными уровнями серьезности события. Применяется только если object - trigger. |
| trigger_severities | integer/array | Возвращать только события с указанными уровнями серьезности триггера. Применяется только если object - trigger. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И;<br>2 - Или. |
| tags | array | Возвращать только события с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все события.<br><br>Возможные значения operator:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| eventid_from | string | Возвращать только события с ID, большим или равным указанному ID. |
| eventid_till | string | Возвращать только события с ID, меньшим или равным указанному ID. |
| time_from | timestamp | Возвращать только события, созданные в указанное время или позже. |
| time_till | timestamp | Возвращать только события, созданные в указанное время или раньше. |

| Parameter | Type | Description |
|---|---|---|
| problem_time_from | timestamp | Возвращает только события, которые находились в состоянии проблемы начиная с `problem_time_from`, независимо от их текущего состояния. Применяется только когда source - событие триггера, а object - trigger. Этот параметр обязателен, если указан `problem_time_till`. |
| problem_time_till | timestamp | Возвращает только события, которые находились в состоянии проблемы до `problem_time_till`, независимо от их текущего состояния. Применяется только когда source - событие триггера, а object - trigger. Этот параметр обязателен, если указан `problem_time_from`. |
| value | integer/array | Возвращать только события с указанными значениями. |
| selectAcknowledges | query | Возвращать свойство `acknowledges` с обновлениями события. Обновления события сортируются в обратном хронологическом порядке.<br><br>Объект обновления события имеет следующие свойства:<br>`acknowledgeid` - `(ID)` ID подтверждения;<br>`userid` - `(ID)` ID пользователя, который обновил событие;<br>`clock` - `(timestamp)` время обновления события;<br>`message` - `(string)` текст сообщения;<br>`action` - `(integer)` выполненное действие обновления, см. `event.acknowledge`;<br>`old_severity` - `(integer)` серьезность события до этого действия обновления;<br>`new_severity` - `(integer)` серьезность события после этого действия обновления;<br>`suppress_until` - `(timestamp)` время, до которого событие будет подавлено;<br>`taskid` - `(ID)` ID задачи, если текущее событие находится в процессе изменения ранга;<br>`username` - `(string)` имя пользователя, который обновил событие;<br>`name` - `(string)` имя пользователя, который обновил событие;<br>`surname` - `(string)` фамилия пользователя, который обновил событие.<br><br>Поддерживает `count`. |
| selectAlerts | query | Возвращать свойство `alerts` с оповещениями, сгенерированными событием. Оповещения сортируются в обратном хронологическом порядке. |
| selectHosts | query | Возвращать свойство `hosts` с узлами сети, содержащими объект, который создал событие. Поддерживается только для событий, сгенерированных триггерами, элементами данных или правилами LLD. |
| selectRelatedObject | query | Возвращать свойство `relatedObject` с объектом, который создал событие. Тип возвращаемого объекта зависит от типа события. |
| selectSuppressionData | query | Возвращать свойство `suppression_data` со списком активных обслуживаний и ручных подавлений:<br>`maintenanceid` - `(ID)` ID обслуживания;<br>`userid` - `(ID)` ID пользователя, который подавил событие;<br>`suppress_until` - `(integer)` время, до которого событие подавлено. |
| selectTags | query | Возвращать свойство `tags` с тегами события. |
| filter | object | Возвращать только те результаты, которые в точности соответствуют заданному фильтру.<br><br>Принимает объект, где ключи - это имена свойств, а значения - либо одно значение, либо массив значений для сравнения.<br><br>Не поддерживает свойства с типом данных `text`. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `eventid`, `objectid`, `clock`.<br><br>Возможные значения при использовании вместе с `groupBy`: `objectid`.<br><br>Возможные значения при использовании вместе с `countOutput` и `groupBy`: `objectid`, `rowscount`. |
| groupBy | string/array | Группировать результаты по указанным свойствам. Указанные свойства будут возвращены в результатах.<br><br>Возможные значения: `objectid`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов
- количество полученных объектов, если был использован параметр `countOutput`, но не был использован параметр `groupBy`
- массив объектов с результатами агрегации, если был использован параметр `groupBy`

Примеры

Получение событий триггера

Получите последние события из триггера "22395".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "selectAcknowledges": "extend",
        "selectSuppressionData": "extend",
        "selectTags": "extend",
        "objectids": "22395",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Ответ:

```
{
    "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",
            "cause_eventid": "0",
            "acknowledges": [],
            "opdata": "Load averages(1m 5m 15m): (1.6328 3.0522 2.2515), # of CPUs: 2",
            "suppression_data": [],
            "suppressed": "0",
            "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",
            "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",
                    "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"
                }
            ],
            "suppressed": "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
}
```


Получение событий по временному периоду

Получите все события, созданные в период с 17 по 18 октября 2012 года, в обратном хронологическом порядке.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "time_from": "1350432000",
        "time_till": "1350518400",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
```

```
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20617",
            "source": "0",
            "object": "0",
            "objectid": "14282",
            "clock": "1350477816",
            "value": "1",
            "acknowledged": "0",
            "ns": "0",
            "name": "Less than 25% free in the history cache",
            "severity": "3",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        },
        {
            "eventid": "20616",
            "source": "0",
            "object": "0",
            "objectid": "14281",
            "clock": "1350477814",
            "value": "0",
            "acknowledged": "0",
            "ns": "0",
            "name": "Zabbix trapper processes more than 75% busy",
            "severity": "0",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        },
        {
            "eventid": "20615",
            "source": "0",
            "object": "0",
            "objectid": "14281",
            "clock": "1350477541",
            "value": "1",
            "acknowledged": "0",
            "ns": "0",
            "name": "Zabbix trapper processes more than 75% busy",
            "severity": "3",
            "r_eventid": "20616",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
```

```
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        }
    ],
    "id": 1
}
```

Получение событий, подтвержденных указанным пользователем

Получение событий, подтвержденных пользователем с ID=10

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Ответ:

```
{
    "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",
            "cause_eventid": "0",
            "acknowledges": [
                {
                    "userid": "10",
                    "action": "2"
                }
            ],
            "opdata": "",
            "suppressed": "0",
            "urls": []
        }
    ],
    "id": 1
}
```

Получение 5 триггеров с наибольшим количеством событий проблемы

Получите 5 триггеров с уровнями серьезности ”Warning”, ”Average”, ”High” или ”Disaster” вместе с количеством событий проблемы за указанный период времени.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "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
}
```

См. также

- Alert
- Элемент данных
- Узел сети
- Правило LLD
- Сервис
- Триггер

Источник

CEvent::get() в *ui/include/classes/api/services/CEvent.php*.

**Тип медиа**

Этот класс предназначен для работы с типами медиа.

Ссылки на объекты:

- Тип медиа
  - Параметры вебхука
  - Параметры скрипта
- Шаблон сообщения

Доступные методы:

- mediatype.create - создать новые типы медиа
- mediatype.delete - удалить типы медиа
- mediatype.get - получить типы медиа
- mediatype.update - обновить типы медиа

**Объект способа оповещения**

Следующие объекты непосредственно связаны с API `mediatype`.

Тип медиа

Объект типа медиа имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| mediatypeid | ID | ID типа медиа.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя типа медиа.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| type | integer | Транспорт, используемый типом медиа.<br><br>Возможные значения:<br>0 - Email;<br>1 - Script;<br>2 - SMS;<br>4 - Webhook.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| exec_path | string | Имя файла скрипта (например, notification.sh), расположенного в каталоге, указанном в параметре конфигурации сервера AlertScriptsPath.<br><br>Поведение свойства:<br>- *обязательно*, если `type` установлен в ”Script” |
| gsm_modem | string | Имя последовательного устройства GSM-модема.<br><br>Поведение свойства:<br>- *обязательно*, если `type` установлен в ”SMS” |
| passwd | string | Пароль аутентификации.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_authentication` установлен в ”Normal password” |

| Property | Type | Description |
|---|---|---|
| provider | integer | Почтовый провайдер.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Generic SMTP;<br>1 - Gmail;<br>2 - Gmail relay;<br>3 - Office365;<br>4 - Office365 relay. |
| smtp_email | string | Адрес электронной почты, с которого будут отправляться уведомления.<br><br>Поведение свойства:<br>- *обязательно*, если `type` установлен в ”Email” |
| smtp_helo | string | SMTP HELO.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Email” |
| smtp_server | string | SMTP-сервер.<br><br>Поведение свойства:<br>- *обязательно*, если `type` установлен в ”Email” |
| smtp_port | integer | Порт SMTP-сервера для подключения.<br><br>По умолчанию: 25.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Email” |
| smtp_security | integer | Уровень безопасности SMTP-соединения, который следует использовать.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Нет;<br>1 - STARTTLS;<br>2 - SSL/TLS.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Email” |
| smtp_verify_host | integer | Проверка SSL host для SMTP.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Нет;<br>1 - Да.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_security` установлен в ”STARTTLS” или ”SSL/TLS” |
| smtp_verify_peer | integer | Проверка SSL peer для SMTP.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Нет;<br>1 - Да.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_security` установлен в ”STARTTLS” или ”SSL/TLS” |

| Property | Type | Description |
|---|---|---|
| smtp_authentication | integer | Метод аутентификации SMTP, который следует использовать.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Нет;<br>1 - Normal password;<br>2 - OAuth token.<br>Аутентификация OAuth не разрешена для почтового провайдера *Office365 relay*.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Email” |
| redirection_url | string | URL веб-интерфейса Zabbix для перенаправления обратно при авторизации OAuth.<br><br>По умолчанию:<br>Значение свойства API settings `url` с частью `zabbix.php?action=oauth.authorize`<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| client_id | string | Идентификатор клиента, зарегистрированный на сервере авторизации OAuth.<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| client_secret | string | Секрет клиента, зарегистрированный на сервере авторизации OAuth. Доступен только пользователю типа Super Admin.<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| authorization_url | string | URL OAuth с параметрами для получения токенов доступа и обновления.<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| token_url | string | URL OAuth для обмена токена авторизации на токены доступа и обновления.<br>Этот URL также используется сервером для обновления недействительного токена доступа.<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| tokens_status | integer | Битовая маска состояния токенов.<br><br>Возможные значения:<br>0 - (по умолчанию) Оба токена содержат недопустимое значение<br>1 - Токен доступа содержит допустимое значение<br>2 - Токен обновления содержит допустимое значение<br>3 - Оба токена содержат допустимое значение.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_authentication` установлен в ”OAuth token” |

| Property | Type | Description |
|---|---|---|
| access_token | string | Значение токена доступа OAuth.<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| access_token_updated | timestamp | Метка времени последнего изменения `access_token`, выполненного сервером при обновлении с помощью `refresh_token` или API при изменении токенов.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_authentication` установлен в ”OAuth token” |
| access_expires_in | integer | Время в секундах, через которое `access_token` станет устаревшим и потребуется выполнить запрос к `refresh_url`. Устанавливается сервером Zabbix при обновлении `access_token` или API при изменении токенов.<br><br>Метка времени вычисляется путем прибавления значения `access_token_updated`.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_authentication` установлен в ”OAuth token” |
| refresh_token | string | Значение токена обновления OAuth.<br><br>Поведение свойства:<br>- *обязательно*, если `smtp_authentication` установлен в ”OAuth token” |
| status | integer | Включен ли тип медиа.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Включен;<br>1 - Отключен. |
| username | string | Имя пользователя.<br><br>Поведение свойства:<br>- *поддерживается*, если `smtp_authentication` установлен в ”Normal password” |
| maxsessions | integer | Максимальное число оповещений, которые могут обрабатываться параллельно.<br><br>Возможные значения, если `type` установлен в ”SMS”: 1.<br><br>Возможные значения, если `type` установлен в ”Email”, ”Script” или ”Webhook”: 0-100.<br><br>По умолчанию: 1. |
| maxattempts | integer | Максимальное число попыток отправки оповещения.<br><br>Возможные значения: 1-100.<br><br>По умолчанию: 3. |
| attempt_interval | string | Интервал между повторными попытками.<br>Принимает секунды и единицу времени с суффиксом.<br><br>Возможные значения: 0-1h.<br><br>По умолчанию: 10s. |

| Property | Type | Description |
|---|---|---|
| message_format | integer | Формат сообщения.<br><br>Возможные значения:<br>0 - Plain text;<br>1 - *(по умолчанию)* HTML.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Email” |
| script | text | Тело скрипта вебхука (JavaScript).<br><br>Поведение свойства:<br>- *обязательно*, если `type` установлен в ”Webhook” |
| timeout | string | Тайм-аут скрипта вебхука.<br>Принимает секунды и единицу времени с суффиксом.<br><br>Возможные значения: 1-60s.<br><br>По умолчанию: 30s.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Webhook” |
| process_tags | integer | Обрабатывать значения свойств JSON в ответе скрипта вебхука как теги. Эти теги добавляются к любым существующим тегам проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Игнорировать ответ скрипта вебхука;<br>1 - Обрабатывать ответ скрипта вебхука как теги.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Webhook” |
| show_event_menu | integer | Включать ли пункт в меню событий, который ведет на пользовательский URL. Также добавляет свойство `urls` в вывод problem.get и event.get.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не включать пункт меню событий или свойство `urls`;<br>1 - Включать пункт меню событий и свойство `urls`.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Webhook” |
| event_menu_url | string | URL, используемый в пункте меню событий и в свойстве `urls`, возвращаемом problem.get и event.get.<br><br>Поведение свойства:<br>- *поддерживается*, если `show_event_menu` установлен в ”Include event menu entry and `urls` property” |
| event_menu_name | string | Имя, используемое в пункте меню событий и в свойстве `urls`, возвращаемом problem.get и event.get.<br><br>Поведение свойства:<br>- *поддерживается*, если `show_event_menu` установлен в ”Include event menu entry and `urls` property” |
| parameters | array | Параметры вебхука или скрипта.<br><br>Поведение свойства:<br>- *поддерживается*, если `type` установлен в ”Webhook” или ”Script” |
| description | text | Описание типа медиа. |

Параметры вебхука

Параметры вебхука имеют следующие свойства.

| Property | Type | Description |
|---|---|---|
| name | string | Имя параметра.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение параметра, поддерживает макросы.<br>Поддерживаемые макросы описаны на странице Поддерживаемые макросы. |

Параметры скрипта

Параметры скрипта имеют следующие свойства.

| Property | Type | Description |
|---|---|---|
| sortorder | integer | Порядок, в котором значения параметров будут передаваться скрипту как аргументы командной строки, начиная с 0 как с первого.<br><br>Property behavior:<br>- *required* |
| value | string | Значение параметра, поддерживает макросы.<br>Поддерживаемые макросы описаны на странице Supported macros. |

Шаблон сообщения

Объект шаблона сообщения определяет шаблон, который будет использоваться как сообщение по умолчанию для операций действия при отправке уведомления. Он имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| eventsource | integer | Источник события.<br><br>Возможные значения:<br>0 - Триггеры;<br>1 - Обнаружение;<br>2 - Автоматическая регистрация;<br>3 - Внутренний;<br>4 - Сервисы.<br><br>Поведение свойства:<br>- *required* |
| recovery | integer | Режим операции.<br><br>Возможные значения:<br>0 - Операции;<br>1 - Операции восстановления;<br>2 - Операции обновления.<br><br>Поведение свойства:<br>- *required* |
| subject | string | Тема сообщения. |
| message | string | Текст сообщения. |

**mediatype.create**

Описание

```
object mediatype.create(    /     mediaTypes)
```

Этот метод позволяет создавать новые способы оповещения.

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Типы оповещений для создания.

Помимо стандартных свойств типа оповещений, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| message_templates | array | Шаблоны сообщений, которые будут созданы для типа оповещений. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных способов оповещения, указанных в свойствеmediatypeids. Порядок возвращаемых ID совпадает с порядком переданных способов оповещения.

Примеры

Создание способа оповещения по электронной почте.

Создать новый способ оповещения по электронной почте с настраиваемым SMTP портом и шаблонами сообщений.

Запрос:

```
{
    "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}\" on host \"{HOST.NAME}\" started at {EVENT.TIME}."
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "Resolved in {EVENT.DURATION}: {EVENT.NAME}",
                "message": "Problem \"{EVENT.NAME}\" on host \"{HOST.NAME}\" has been resolved at {EVENT.R
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "Updated problem in {EVENT.AGE}: {EVENT.NAME}",
                "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem \"{EVENT.NAME}\" on host \"{HOST
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "7"
        ]
    },
    "id": 1
}
```

Создание способа оповещения с использованием скрипта

Создать новый способ оповещения с использованием скрипта с пользовательским значением для количества попыток и интервала между ними.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "8"
        ]
    },
    "id": 1
}
```

Создание способа оповещения с использованием вебхука

Создать новый способ оповещения с использованием вебхука.

Запрос:

```
{
    "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
}
```


Ответ:


```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "9"
        ]
    },
    "id": 1
}
```


Источник

CMediaType::create() в *ui/include/classes/api/services/CMediaType.php*.

**mediatype.delete**

Описание

`object mediatype.delete(    mediaTypeIds)`

Этот метод позволяет удалять способы оповещения.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(  ) ID удаляемых способов оповещения.

Возвращаемые значения

(  ) Возвращает объект, который содержит ID удаленных способов оповещения, указанных в свойстве `mediatypeids`.

Примеры

Удаление нескольких способов оповещения

Удалить два способа оповещения.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.delete",
    "params": [
        "3",
        "5"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "3",
            "5"
        ]
    },
    "id": 1
}
```

Источник

CMediaType::delete() в *ui/include/classes/api/services/CMediaType.php*.

**mediatype.get**

Описание

`integer/array mediatype.get(object parameters)`

Метод позволяет получать типы медиа в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

**Note:**
При запросе информации, связанной с пользователями, о типах медиа *Admin* и *User* могут получать данные только о собственном пользователе. Пример см. в разделе Получение типов медиа как *Admin*.

| Parameter | Type | Description |
|---|---|---|
| mediatypeids | ID/array | Возвращать только типы медиа с указанными ID. |
| mediaids | ID/array | Возвращать только типы медиа, используемые указанным media. |
| userids | ID/array | Возвращать только типы медиа, используемые указанными пользователями. |
| selectActions | query | Возвращать свойство `actions` с действиями, которые используют тип медиа. |
| selectMessageTemplates | query | Возвращать свойство `message_templates` с массивом сообщений типа медиа.<br><br>Поведение параметра:<br>- *supported* для пользователей типа *Super admin* |

| Parameter | Type | Description |
|---|---|---|
| selectUsers | query | Возвращать свойство `users` с пользователями, которые используют тип медиа.<br><br>См. `user.get` для ограничений в зависимости от типа пользователя. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `mediatypeid`. |
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений для сравнения.<br><br>Не поддерживает свойства с типом данных `text`. |
| output | query | Возможные свойства объекта типа медиа для пользователей типа *Admin* и *User*: `mediatypeid`, `name`, `type`, `status`, `maxattempts`.<br>Свойства объекта типа медиа, которые нужно вернуть.<br><br>Пользователи типа *Admin* и *User* могут получать только следующие свойства: `mediatypeid`, `name`, `type`, `status`, `maxattempts`, `description`. Пример см. в разделе Получение типов медиа как *Admin*.<br><br>По умолчанию: `extend`. |
| search | object | Возвращать результаты, соответствующие указанному шаблону (без учета регистра).<br><br>Принимает объект, где ключи — это имена свойств, а значения — строки для поиска. Если дополнительные параметры не указаны, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживает только свойства с типом данных `string` и `text`.<br><br>Возможные свойства объекта типа медиа для пользователей типа *Admin* и *User*: `name`, `description`. |
| countOutput | boolean | Эти параметры описаны в комментарии к справочнику. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| preservekeys | boolean | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

( / ) Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если был использован параметр `countOutput`.

Примеры

Получение типов оповещений

Получить все настроенные типы оповещений. Следующий пример возвращает два типа оповещений:

- тип оповещения по электронной почте;
- тип оповещения по SMS.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "selectMessageTemplates": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "            {EVENT.TIME} {EVENT.DATE}\r\n       : {EVENT.NAME}\r\n      : {
                },
                {
                    "eventsource": "0",
                    "recovery": "1",
                    "subject": "Resolved: {EVENT.NAME}",
                    "message": "               {EVENT.RECOVERY.TIME} {EVENT.RECOVERY.DATE}\r\n       : {EVE
                },
                {
                    "eventsource": "0",
                    "recovery": "2",
                    "subject": "Updated problem: {EVENT.NAME}",
                    "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION}        {EVENT.UPDATE.DATE} {EVENT.UP
                },
```

```
            {
                "eventsource": "1",
                "recovery": "0",
                "subject": "Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}",
                "message": "           : {DISCOVERY.RULE.NAME}\r\n\r\nIP       : {DISCOVERY.DEVICE.IPAD
            },
            {
                "eventsource": "2",
                "recovery": "0",
                "subject": "Autoregistration: {HOST.HOST}",
                "message": "        : {HOST.HOST}\r\nIP       : {HOST.IP}\r\n       : {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\n      : {HOST.NAME}\r\n{EVENT.DATE} {EVEN
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "",
                "message": "RESOLVED: {EVENT.NAME}\r\n      : {HOST.NAME}\r\n{EVENT.DATE} {EVENT.TIME}"
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "",
                "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION}        {EVENT.UPDATE.DATE} {EVENT.UP
```

```
                },
                {
                    "eventsource": "1",
                    "recovery": "0",
                    "subject": "",
                    "message": "Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}"
                },
                {
                    "eventsource": "2",
                    "recovery": "0",
                    "subject": "",
                    "message": "Autoregistration: {HOST.HOST}\r\nIP      : {HOST.IP}\r\n      : {HOST.POR
                }
            ],
            "parameters": []
        }
    ],
    "id": 1
}
```

Получение типов оповещений в роли *Admin*

Пользователь типа *Admin* может получить все включенные типы оповещений вместе с пользователями, которые используют эти типы оповещений. Следующий пример возвращает два типа оповещений:

- тип оповещений email с одним пользователем (только собственный пользователь пользователя типа *Admin*);
- тип оповещений SMS без пользователей.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "filter": {
            "status": 0
        },
        "selectUsers": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


Получение способов оповещения с использованием скриптов и вебхуков

В следующем примере возвращаются три способа оповещения:

- способ оповещения с использованием скрипта с параметрами;
- способ оповещения с использованием скрипта без параметров;
- способ оповещения с использованием вебхука с параметрами.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": ["mediatypeid", "name", "parameters"],
        "filter": {
            "type": [1, 4]
        }
    },
    "id": 1
}
```

Ответ:


```
{
    "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
}
```

Смотрите также

- Пользователь

Источник

CMediaType::get() в *ui/include/classes/api/services/CMediaType.php*.

**mediatype.update**

Описание

`object mediatype.update(    /     mediaTypes)`

Этот метод позволяет обновлять существующие способы оповещения

**Note:**
Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства типа оповещения, которые необходимо обновить.

Свойство `mediatypeid` должно быть определено для каждого типа оповещения, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам типа оповещения, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| message_templates | array | Шаблоны сообщений для замены текущих шаблонов сообщений. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID обновленных способов оповещений, указанных в свойстве `mediatypeids`.

Примеры

Включение типа оповещений

Включите тип оповещений, то есть установите для его статуса значение "0".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.update",
    "params": {
        "mediatypeid": "6",
        "status": "0"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "6"
        ]
    },
    "id": 1
}
```

Источник

CMediaType::update() в *ui/include/classes/api/services/CMediaType.php*.

**Тренд**

Этот класс предназначен для работы с данными трендов.

Ссылки на объекты:

- Float trend
- Integer trend

Доступные методы:

- trend.get - получить тренды

### Объект тренда

Следующие объекты напрямую связаны с API `trend`.

**Note:**
Объекты тренда различаются в зависимости от типа информации элемента данных. Они создаются сервером Zabbix и не могут быть изменены через API.

Объект тренда float

Объект тренда float имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Метка времени часа, для которого было рассчитано значение. Например, метка времени ”04:00:00” означает значения, рассчитанные за период ”04:00:00-04:59:59”. |
| itemid | ID | ID связанного элемента данных. |
| num | integer | Количество значений, доступных за час. |
| value_min | float | Минимальное значение за час. |
| value_avg | float | Среднее значение за час. |
| value_max | float | Максимальное значение за час. |

Целочисленный тренд

Объект целочисленного тренда имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| clock | timestamp | Метка времени часа, для которого было вычислено значение. Например, метка времени ”04:00:00” означает значения, вычисленные за период ”04:00:00-04:59:59”. |
| itemid | ID | ID связанного элемента данных. |
| num | integer | Количество значений, которые были доступны за час. |
| value_min | integer | Минимальное значение за час. |
| value_avg | integer | Среднее значение за час. |
| value_max | integer | Максимальное значение за час. |

### trend.get

Описание

`integer/array trend.get(object parameters)`

Метод позволяет получать данные трендов в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| itemids | ID/array | Возвращать только тренды с указанными ID элементов данных. |
| time_from | timestamp | Возвращать только значения, собранные после указанного времени или в указанное время. |
| time_till | timestamp | Возвращать только значения, собранные до указанного времени или в указанное время. |
| countOutput | boolean | Подсчитать количество полученных объектов. |

| Parameter | Type | Description |
|---|---|---|
| limit | integer | Ограничить количество полученных объектов. |
| output | query | Указать свойства объекта Trend object, которые нужно вернуть. |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение данных тренда элемента данных

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trend.get",
    "params": {
        "output": [
            "itemid",
            "clock",
            "num",
            "value_min",
            "value_avg",
            "value_max"
        ],
        "itemids": [
            "23715"
        ],
        "limit": "1"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23715",
            "clock": "1446199200",
            "num": "60",
            "value_min": "0.165",
            "value_avg": "0.2168",
            "value_max": "0.35"
        }
    ],
    "id": 1
}
```

Источник

CTrend::get() в *ui/include/classes/api/services/CTrend.php*.

**Триггер**

Этот класс предназначен для работы с триггерами.

Ссылки на объекты:

- Trigger
- Trigger tag

Доступные методы:

- trigger.create - создать новые триггеры
- trigger.delete - удалить триггеры
- trigger.get - получить триггеры
- trigger.update - обновить триггеры

**Объект триггера**

Следующие объекты напрямую связаны с API `trigger`.

Триггер

Объект триггера имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| triggerid | ID | ID триггера.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| description | string | Имя триггера.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| expression | string | Выражение триггера.<br><br>Элементы данных с `value_type`, установленным в 5 (binary), нельзя использовать в выражениях триггера.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| event_name | string | Имя события, создаваемого триггером. |
| opdata | string | Операционные данные. |
| comments | string | Дополнительное описание триггера. |
| error | string | Текст ошибки, если при обновлении состояния триггера возникли проблемы.<br><br>Поведение свойства:<br>- *только для чтения* |
| flags | integer | Происхождение триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* обычный триггер;<br>4 - триггер, преобразованный из прототипа.<br><br>Поведение свойства:<br>- *только для чтения* |
| lastchange | timestamp | Время последнего изменения состояния триггера.<br><br>Поведение свойства:<br>- *только для чтения* |
| priority | integer | Серьезность триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не классифицировано;<br>1 - информация;<br>2 - предупреждение;<br>3 - средняя;<br>4 - высокая;<br>5 - катастрофа. |

| Property | Type | Description |
|---|---|---|
| state | integer | Состояние триггера.<br><br>Возможные значения:<br>0 - *(по умолчанию)* состояние триггера актуально;<br>1 - текущее состояние триггера неизвестно.<br><br>Поведение свойства:<br>- *только для чтения* |
| status | integer | Включен или отключен триггер.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включен;<br>1 - отключен. |
| templateid | ID | ID родительского триггера шаблона.<br><br>Поведение свойства:<br>- *только для чтения* |
| type | integer | Может ли триггер генерировать несколько событий проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* не генерировать несколько событий;<br>1 - генерировать несколько событий. |
| url | string | URL, связанный с триггером. |
| url_name | string | Метка для URL, связанного с триггером. |
| value | integer | Находится ли триггер в состоянии OK или проблемы.<br><br>Возможные значения:<br>0 - *(по умолчанию)* OK;<br>1 - проблема.<br><br>Поведение свойства:<br>- *только для чтения* |
| recovery_mode | integer | Режим генерации события OK.<br><br>Возможные значения:<br>0 - *(по умолчанию)* выражение;<br>1 - выражение восстановления;<br>2 - нет. |
| recovery_expression | string | Выражение восстановления триггера.<br><br>Элементы данных с `value_type`, установленным в 5 (binary), нельзя использовать в выражениях восстановления триггера. |
| correlation_mode | integer | Закрытие события OK.<br><br>Возможные значения:<br>0 - *(по умолчанию)* все проблемы;<br>1 - все проблемы, если значения тегов совпадают. |
| correlation_tag | string | Тег для сопоставления. |
| manual_close | integer | Разрешить ручное закрытие.<br><br>Возможные значения:<br>0 - *(по умолчанию)* нет;<br>1 - да. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных триггеров с уже существующими. Если не указан, генерируется автоматически.<br><br>Поведение свойства:<br>- *поддерживается*, если триггер принадлежит шаблону |

Тег триггера

Объект тега триггера имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега триггера.<br>Поведение свойства:<br>- *required* |
| value | string | Значение тега триггера. |

**trigger.create**

Описание

`object trigger.create( / triggers)`

Этот метод позволяет создавать новые триггеры.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Триггеры для создания.

В дополнение к стандартным свойствам триггера метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dependencies | array | Триггеры, от которых зависит триггер.<br><br>У триггеров должно быть определено только свойство `triggerid`. |
| tags | array | Теги триггера. |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID созданных триггеров, указанных в свойстве`triggerids`. Порядок возвращаемых ID совпадает с порядком переданных триггеров.

Примеры

Создание триггера

Создайте два триггера, каждый из которых зависит от одного другого триггера.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.create",
    "params": [
        {
            "description": "                          {HOST.NAME}",
            "expression": "last(/Linux server/system.cpu.load[percpu,avg1])>5",
            "dependencies": [
                {
                    "triggerid": "17367"
                }
            ]
        },
        {
            "description": "          ",
            "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 1
}
```

Источник

CTrigger::create() в *ui/include/classes/api/services/CTrigger.php*.

### trigger.delete

Описание

`object trigger.delete(    triggerIds)`

Этот метод позволяет удалять триггеры.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых триггеров.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных триггеров, указанных в свойстве `triggerids`.

Примеры

Удаление нескольких триггеров

Удаление двух триггеров.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.delete",
    "params": [
```

```
        "12002",
        "12003"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "12002",
            "12003"
        ]
    },
    "id": 1
}
```

Источник

CTrigger::delete() в *ui/include/classes/api/services/CTrigger.php*.

**trigger.get**

Описание

`integer/array trigger.get(          )`

Метод позволяет получать триггеры по заданным параметрам.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| triggerids | ID/array | Возвращать только триггеры с указанными ID. |
| groupids | ID/array | Возвращать только триггеры, которые принадлежат узлам сети или шаблонам из указанных групп узлов сети или групп шаблонов. |
| templateids | ID/array | Возвращать только триггеры, которые принадлежат указанным шаблонам. |
| hostids | ID/array | Возвращать только триггеры, которые принадлежат указанным узлам сети. |
| itemids | ID/array | Возвращать только триггеры, которые содержат указанные элементы данных. |
| functions | string/array | Возвращать только триггеры, которые используют указанные функции.<br><br>Список поддерживаемых функций см. на странице поддерживаемые функции. |
| group | string | Возвращать только триггеры, которые принадлежат узлам сети или шаблонам из группы узлов сети или группы шаблонов с указанным именем. |
| host | string | Возвращать только триггеры, которые принадлежат узлу сети с указанным техническим именем. |
| inherited | boolean | Если установлено значение `true`, возвращать только триггеры, унаследованные из шаблона. |
| templated | boolean | Если установлено значение `true`, возвращать только триггеры, которые принадлежат шаблонам. |

| Parameter | Type | Description |
|---|---|---|
| dependent | boolean | Если установлено значение `true`, возвращать только триггеры, которые имеют зависимости. Если установлено значение `false`, возвращать только триггеры, которые не имеют зависимостей. |
| monitored | flag | Возвращать только включенные триггеры, которые принадлежат контролируемым узлам сети и содержат только включенные элементы данных. |
| active | flag | Возвращать только включенные триггеры, которые принадлежат контролируемым узлам сети. |
| maintenance | boolean | Если установлено значение `true`, возвращать только включенные триггеры, которые принадлежат узлам сети в обслуживании. |
| withUnacknowledgedEvents | flag | Возвращать только триггеры, у которых есть неподтвержденные события. |
| withAcknowledgedEvents | flag | Возвращать только триггеры, у которых все события подтверждены. |
| withLastEventUnacknowledged | flag | Возвращать только триггеры, у которых последнее событие не подтверждено. |
| skipDependent | flag | Пропускать триггеры в состоянии проблемы, которые зависят от других триггеров. Обратите внимание, что другие триггеры игнорируются, если они отключены, имеют отключенные элементы данных или отключенные узлы сети элементов данных. |
| lastChangeSince | timestamp | Возвращать только триггеры, состояние которых изменилось после указанного времени. |
| lastChangeTill | timestamp | Возвращать только триггеры, состояние которых изменилось до указанного времени. |
| only_true | flag | Возвращать только триггеры, которые недавно находились в состоянии проблемы (в пределах `ok_period`). |
| min_severity | integer | Возвращать только триггеры с уровнем важности больше или равным указанному. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И;<br>2 - Или. |
| tags | array | Возвращать только триггеры с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все триггеры.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| expandComment | flag | Разворачивать макросы в описании триггера. |
| expandDescription | flag | Разворачивать макросы в имени триггера. |
| expandExpression | flag | Разворачивать функции и макросы в выражении триггера. |
| selectHostGroups | query | Возвращать группы узлов сети, к которым принадлежит триггер, в свойстве `hostgroups`. |
| selectHosts | query | Возвращать узлы сети, к которым принадлежит триггер, в свойстве `hosts`. |
| selectItems | query | Возвращать элементы данных, содержащиеся в триггере, в свойстве `items`. |

| Parameter | Type | Description |
|---|---|---|
| selectFunctions | query | Возвращать функции, используемые в триггере, в свойстве `functions`.<br><br>Объекты функции представляют функции, используемые в выражении триггера, и имеют следующие свойства:<br>`functionid` - `(ID)` ID функции;<br>`itemid` - `(ID)` ID элемента данных, используемого в функции;<br>`function` - `(string)` имя функции;<br>`parameter` - `(string)` параметр, переданный функции.<br>Параметр запроса в возвращаемой строке заменяется символом $. |
| selectDependencies | query | Возвращать триггеры, от которых зависит триггер, в свойстве `dependencies`. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения триггера. Объект обнаружения триггера связывает обнаруженный триггер с прототипом триггера, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`parent_triggerid` - `(ID)` ID прототипа триггера, из которого был создан триггер;<br>`status` - `(int)` статус обнаружения триггера:<br>0 - *(по умолчанию)* триггер обнаружен,<br>1 - триггер больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда триггер, который больше не обнаружен, будет удален;<br>`ts_disable` - `(timestamp)` время, когда триггер, который больше не обнаружен, будет отключен;<br>`disable_source` - `(int)` признак того, был ли триггер отключен правилом LLD или вручную:<br>0 - *(по умолчанию)* отключен автоматически,<br>1 - отключен правилом LLD. |
| selectDiscoveryRule | query | Возвращать правило обнаружения низкого уровня, которое создало триггер, в свойстве `discoveryRule`. |
| selectLastEvent | query | Возвращать последнее значимое событие триггера в свойстве `lastEvent`. |
| selectTags | query | Возвращать теги триггера в свойстве `tags`. |
| selectTemplateGroups | query | Возвращать группы шаблонов, к которым принадлежит триггер, в свойстве `templategroups`. |
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит триггер;<br>`hostid` - ID узла сети, к которому принадлежит триггер. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подвыборками.<br><br>Применяется к следующим подвыборкам:<br>`selectHosts` - результаты будут отсортированы по `host`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `triggerid`, `description`, `status`, `priority`, `lastchange`, `hostname`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |

| Parameter | Type | Description |
|---|---|---|
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |
| selectTriggerDiscovery | query | Возвращать объект обнаружения триггера в свойстве `triggerDiscovery`. Объекты обнаружения триггера связывают триггер с прототипом триггера, из которого он был создан.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryData`. |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество полученных объектов, если использован параметр `countOutput`.

Примеры

Получение данных по ID триггера

Получите все данные и функции, используемые в триггере ”14062”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "triggerids": "14062",
        "output": "extend",
        "selectFunctions": "extend"
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "                    10    ",
            "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",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```

Получение триггеров в состоянии проблемы

Получите ID, имя и уровень серьезности всех триггеров в состоянии проблемы и отсортируйте их по уровню серьезности по убыванию.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description",
            "priority"
        ],
        "filter": {
            "value": 1
        },
        "sortfield": "priority",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "13907",
            "description": "Zabbix self-monitoring processes < 100% busy",
            "priority": "4"
        },
        {
            "triggerid": "13824",
            "description": "Zabbix discoverer processes more than 75% busy",
            "priority": "3"
        }
    ],
    "id": 1
}
```

Получение конкретного триггера с тегами

Получите конкретный триггер с тегами.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17578"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "17370",
            "description": "Service status",
            "tags": [
                {
                    "tag": "service",
                    "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\1\")}"
                },
                {
                    "tag": "error",
                    "value": ""
                }
            ]
        }
    ],
    "id": 1
}
```

См. также

- Правило обнаружения
- Элемент данных
- Узел сети
- Группа узлов сети
- Группа шаблонов

Источник

CTrigger::get() в *ui/include/classes/api/services/CTrigger.php*.

**trigger.update**

Описание

`object trigger.update( / triggers)`

Этот метод позволяет обновлять существующие триггеры.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Свойства триггера, которые необходимо обновить.

Для каждого триггера должно быть определено свойство triggerid, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

Дополнительно к стандартным свойствам триггера метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dependencies | array | Триггеры, от которых зависит триггер.<br><br>У триггеров должно быть определено только свойство triggerid. |
| tags | array | Теги триггера. |

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы обновленных триггеров в свойстве triggerids.

Примеры

Включение триггера

Включите триггер, то есть установите для него статус "0".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Замена тегов триггера

Замените теги для триггера.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Замена зависимостей

Заменить зависимости для триггера.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "22713",
        "dependencies": [
            {
                "triggerid": "22712"
            },
            {
                "triggerid": "22772"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "22713"
        ]
    },
    "id": 1
}
```

Источник

CTrigger::update() в *ui/include/classes/api/services/CTrigger.php*.

**Узел высокой доступности**

Этот класс предназначен для работы с узлами сервера, которые входят в кластер высокой доступности, или с отдельным экземпляром сервера.

Ссылки на объекты:

- Узел высокой доступности

Доступные методы:

- hanode.get - получить узлы

### Объект узла высокой доступности

Следующий объект связан с эксплуатацией кластера высокой доступности серверов Zabbix.

Узел высокой доступности

> **Note:**
> Узлы создаются сервером Zabbix и не могут быть изменены через API.

Объект узла высокой доступности имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| ha_nodeid | ID | ID узла. |
| name | string | Имя, назначенное узлу, с использованием параметра конфигурации HANodeName в zabbix_server.conf. Пусто для сервера, работающего в автономном режиме. |
| address | string | IP-адрес или DNS-имя, с которого узел подключается. |
| port | integer | Порт, на котором работает узел. |
| lastaccess | integer | Время heartbeat, то есть время последнего обновления от узла. Метка времени UTC. |
| status | integer | Состояние узла.<br><br>Возможные значения:<br>0 - standby;<br>1 - остановлен вручную;<br>2 - недоступен;<br>3 - активен. |

### hanode.get

Описание

`integer/array hanode.get(`  `)`

Метод позволяет получить список узлов кластера высокой доступности по заданным параметрам.

> **Note:**
> Этот метод доступен только пользователю с типом доступа *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| ha_nodeids | ID/array | Возвращать только узлы сети с указанными ID узлов сети. |
| filter | object | Возвращать только те результаты, которые точно соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Поддерживаемые свойства: `name`, `address`, `status`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `name`, `lastaccess`, `status`. |
| countOutput | flag | Эти параметры описаны в справочном комментарии. |
| limit | integer | |

| Parameter | Type | Description |
|---|---|---|
| output | query | |
| preservekeys | boolean | |
| sortorder | string/array | |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получить список узлов сети, упорядоченных по статусу

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "preservekeys": true,
        "sortfield": "status",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Ответ:

```
{
  "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
}
```

Получение списка конкретных узлов по их ID

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "ha_nodeids": ["ckuo7i1nw000e0sajwfttc1mp", "ckuo7i1nv000c0sajz85xcrtt"]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Получить список остановленных узлов сети

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["ha_nodeid", "address", "port"],
        "filter": {
            "status": 1
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Получить количество резервных узлов сети

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "countOutput": true,
        "filter": {
            "status": 0
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": "3",
    "id": 1
}
```

Проверка статуса узлов сети по указанным IP-адресам

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["name", "status"],
        "filter": {
            "address": ["192.168.1.7", "192.168.1.13"]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "name": "node3",
            "status": "0"
        },
        {
            "name": "node-active",
```

```
            "status": "3"
        }
    ],
    "id": 1
}
```

Источник

CHaNode::get() в *ui/include/classes/api/services/CHaNode.php*.

#### Узел сети

Этот класс предназначен для работы с узлами сети.

Ссылки на объекты:

- Узел сети
- Инвентарь узла сети
- Тег узла сети

Доступные методы:

- host.create - создать новые узлы сети
- host.delete - удалить узлы сети
- host.get - получить узлы сети
- host.massadd - добавить связанные объекты к узлам сети
- host.massremove - удалить связанные объекты из узлов сети
- host.massupdate - заменить или удалить связанные объекты из узлов сети
- host.update - обновить узлы сети

#### Объект узла сети

Следующие объекты напрямую связаны с API host.

Узел сети

Объект узла сети имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| hostid | ID | ID узла сети.<br><br>Property behavior:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| host | string | Техническое имя узла сети.<br><br>Property behavior:<br>- *обязательно* для операций создания |
| description | text | Описание узла сети. |
| flags | integer | Origin узла сети.<br><br>Possible values:<br>0 - a plain host;<br>4 - a host converted from prototype.<br><br>Property behavior:<br>- *только для чтения* |
| inventory_mode | integer | Режим заполнения инвентаря узла сети.<br><br>Possible values:<br>-1 - *(default)* disabled;<br>0 - manual;<br>1 - automatic. |

| Property | Type | Description |
|---|---|---|
| ipmi_authtype | integer | Алгоритм аутентификации IPMI.<br><br>Possible values:<br>-1 - *(default)* default;<br>0 - none;<br>1 - MD2;<br>2 - MD5<br>4 - straight;<br>5 - OEM;<br>6 - RMCP+. |
| ipmi_password | string | Пароль IPMI. |
| ipmi_privilege | integer | Уровень привилегий IPMI.<br><br>Possible values:<br>1 - callback;<br>2 - *(default)* user;<br>3 - operator;<br>4 - admin;<br>5 - OEM. |
| ipmi_username | string | Имя пользователя IPMI. |
| maintenance_from | timestamp | Время начала текущего обслуживания.<br><br>Property behavior:<br>- *только для чтения* |
| maintenance_status | integer | Текущий статус обслуживания.<br><br>Possible values:<br>0 - *(default)* no maintenance;<br>1 - maintenance in effect.<br><br>Property behavior:<br>- *только для чтения* |
| maintenance_type | integer | Текущий тип обслуживания.<br><br>Possible values:<br>0 - *(default)* maintenance with data collection;<br>1 - maintenance without data collection.<br><br>Property behavior:<br>- *только для чтения* |
| maintenanceid | ID | ID обслуживания, которое в данный момент действует для узла сети.<br><br>Property behavior:<br>- *только для чтения* |
| name | string | Отображаемое имя узла сети.<br><br>Default: `host` property value. |
| monitored_by | integer | Источник, который используется для мониторинга узла сети.<br><br>Possible values:<br>0 - *(default)* Zabbix server;<br>1 - Proxy;<br>2 - Proxy group. |
| proxyid | ID | ID прокси, который используется для мониторинга узла сети.<br><br>Property behavior:<br>- *обязательно* если `monitored_by` set to ”Proxy” |

| Property | Type | Description |
|---|---|---|
| proxy_groupid | ID | ID группы прокси, которая используется для мониторинга узла сети.<br><br>Property behavior:<br>- *обязательно* если `monitored_by` set to ”Proxy group” |
| status | integer | Статус и функция узла сети.<br><br>Possible values:<br>0 - *(default)* enabled;<br>1 - disabled. |
| tls_connect | integer | Подключения к узлу сети.<br><br>Possible values:<br>1 - *(default)* No encryption;<br>2 - PSK;<br>4 - certificate. |
| tls_accept | integer | Подключения от узла сети.<br><br>Possible bitmap values:<br>1 - *(default)* No encryption;<br>2 - PSK;<br>4 - certificate.<br><br>This is a bitmask field; any sum of possible bitmap values is acceptable (for example, 6 for PSK and certificate). |
| tls_issuer | string | Издатель сертификата. |
| tls_subject | string | Тема сертификата. |
| tls_psk_identity | string | Идентификатор PSK; должен быть связан только с одним PSK (в autoregistration, hosts, and proxies).<br><br>Не включайте конфиденциальную информацию в идентификатор PSK, так как он передается по сети без шифрования, чтобы получатель знал, какой PSK использовать.<br><br>Property behavior:<br>- *только для записи*<br>- *обязательно* если `tls_connect` set to ”PSK”, or `tls_accept` contains the ”PSK” bit |
| tls_psk | string | Предварительно общий ключ (PSK); должен содержать не менее 32 шестнадцатеричных цифр.<br><br>Property behavior:<br>- *только для записи*<br>- *обязательно* если `tls_connect` set to ”PSK”, or `tls_accept` contains the ”PSK” bit |
| active_available | integer | Статус доступности активного интерфейса узла сети.<br><br>Possible values:<br>0 - interface status is unknown;<br>1 - interface is available;<br>2 - interface is not available.<br><br>Property behavior:<br>- *только для чтения* |
| assigned_proxyid | ID | ID прокси, назначенного сервером Zabbix, если узел сети мониторится группой прокси.<br><br>Property behavior:<br>- *только для чтения* |

Инвентаризация узла сети

Объект инвентаризации узла сети имеет следующие свойства.

**Note:**
Каждое свойство имеет свой уникальный идентификационный номер, который используется для связывания полей инвентаризации узла сети с элементами данных.

| ID | Свойство | Тип | Описание | Максимальная длина |
|---|---|---|---|---|
| 4 | alias | string | Псевдоним. | 128 characters |
| 11 | asset_tag | string | Инвентарный номер. | 64 characters |
| 28 | chassis | string | Шасси. | 64 characters |
| 23 | contact | string | Контактное лицо. | 65535 characters for SQL databases |
| 32 | contract_number | string | Номер контракта. | 64 characters |
| 47 | date_hw_decomm | string | Дата вывода оборудования из эксплуатации. | 64 characters |
| 46 | date_hw_expiry | string | Дата окончания срока обслуживания оборудования. | 64 characters |
| 45 | date_hw_install | string | Дата установки оборудования. | 64 characters |
| 44 | date_hw_purchase | string | Дата покупки оборудования. | 64 characters |
| 34 | deployment_status | string | Статус развертывания. | 64 characters |
| 14 | hardware | string | Оборудование. | 255 characters |
| 15 | hardware_full | string | Подробные сведения об оборудовании. | 65535 characters for SQL databases |
| 39 | host_netmask | string | Маска подсети узла сети. | 39 characters |
| 38 | host_networks | string | Сети узла сети. | 65535 characters for SQL databases |
| 40 | host_router | string | Маршрутизатор узла сети. | 39 characters |
| 30 | hw_arch | string | Архитектура оборудования. | 32 characters |
| 33 | installer_name | string | Имя установщика. | 64 characters |
| 24 | location | string | Местоположение. | 65535 characters for SQL databases |
| 25 | location_lat | string | Широта местоположения. | 16 characters |
| 26 | location_lon | string | Долгота местоположения. | 16 characters |
| 12 | macaddress_a | string | MAC-адрес A. | 64 characters |
| 13 | macaddress_b | string | MAC-адрес B. | 64 characters |
| 29 | model | string | Модель. | 64 characters |
| 3 | name | string | Имя. | 128 characters |
| 27 | notes | string | Заметки. | 65535 characters for SQL databases |
| 41 | oob_ip | string | IP-адрес OOB. | 39 characters |
| 42 | oob_netmask | string | Маска подсети узла сети OOB. | 39 characters |
| 43 | oob_router | string | Маршрутизатор OOB. | 39 characters |
| 5 | os | string | Имя ОС. | 128 characters |
| 6 | os_full | string | Подробное имя ОС. | 255 characters |
| 7 | os_short | string | Краткое имя ОС. | 128 characters |
| 61 | poc_1_cell | string | Мобильный номер основного POC. | 64 characters |
| 58 | poc_1_email | string | Основной email. | 128 characters |
| 57 | poc_1_name | string | Имя основного POC. | 128 characters |
| 63 | poc_1_notes | string | Заметки по основному POC. | 65535 characters for SQL databases |
| 59 | poc_1_phone_a | string | Телефон A основного POC. | 64 characters |
| 60 | poc_1_phone_b | string | Телефон B основного POC. | 64 characters |
| 62 | poc_1_screen | string | Имя в мессенджере основного POC. | 64 characters |
| 68 | poc_2_cell | string | Мобильный номер вторичного POC. | 64 characters |
| 65 | poc_2_email | string | Email вторичного POC. | 128 characters |
| 64 | poc_2_name | string | Имя вторичного POC. | 128 characters |
| 70 | poc_2_notes | string | Заметки по вторичному POC. | 65535 characters for SQL databases |
| 66 | poc_2_phone_a | string | Телефон A вторичного POC. | 64 characters |
| 67 | poc_2_phone_b | string | Телефон B вторичного POC. | 64 characters |
| 69 | poc_2_screen | string | Имя в мессенджере вторичного POC. | 64 characters |
| 8 | serialno_a | string | Серийный номер A. | 64 characters |
| 9 | serialno_b | string | Серийный номер B. | 64 characters |
| 48 | site_address_a | string | Адрес площадки A. | 128 characters |
| 49 | site_address_b | string | Адрес площадки B. | 128 characters |
| 50 | site_address_c | string | Адрес площадки C. | 128 characters |
| 51 | site_city | string | Город площадки. | 128 characters |
| 53 | site_country | string | Страна площадки. | 64 characters |

| ID | Свойство | Тип | Описание | Максимальная длина |
|---|---|---|---|---|
| 56 | site_notes | string | Заметки о площадке. | 65535 characters for SQL databases |
| 55 | site_rack | string | Расположение стойки на площадке. | 128 characters |
| 52 | site_state | string | Регион/штат площадки. | 64 characters |
| 54 | site_zip | string | Почтовый индекс площадки. | 64 characters |
| 16 | software | string | Программное обеспечение. | 255 characters |
| 18 | software_app_a | string | Программное приложение A. | 64 characters |
| 19 | software_app_b | string | Программное приложение B. | 64 characters |
| 20 | software_app_c | string | Программное приложение C. | 64 characters |
| 21 | software_app_d | string | Программное приложение D. | 64 characters |
| 22 | software_app_e | string | Программное приложение E. | 64 characters |
| 17 | software_full | string | Сведения о программном обеспечении. | 65535 characters for SQL databases |
| 10 | tag | string | Тег. | 64 characters |
| 1 | type | string | Тип. | 64 characters |
| 2 | type_full | string | Сведения о типе. | 64 characters |
| 35 | url_a | string | URL A. | 2048 characters |
| 36 | url_b | string | URL B. | 2048 characters |
| 37 | url_c | string | URL C. | 2048 characters |
| 31 | vendor | string | Поставщик. | 64 characters |

Тег узла сети

Объект тега узла сети имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега узла сети.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение тега узла сети. |
| automatic | integer | Тип тега узла сети.<br><br>Возможные значения:<br>0 - *(default)* manual (тег, созданный пользователем);<br>1 - automatic (тег, созданный при обнаружении на низком уровне)<br><br>Поведение свойства:<br>- *read-only* |

**host.create**

Описание

`object host.create( / hosts)`

Этот метод позволяет создавать новые хосты.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Узлы сети для создания.

В дополнение к стандартным свойствам узла сети метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы узлов сети, к которым нужно добавить узел сети.<br><br>Группы узлов сети должны содержать только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* |
| interfaces | object/array | Интерфейсы, которые нужно создать для узла сети. |
| tags | object/array | Теги узла сети. |
| templates | object/array | Шаблоны, которые нужно привязать к узлу сети.<br><br>Шаблоны должны содержать только свойство `templateid`. |
| macros | object/array | Пользовательские макросы, которые нужно создать для узла сети. |
| inventory | object | Свойства инвентаря узла сети. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы созданных хостов в свойстве`hostids`. Порядок возвращаемых идентификаторов соответствует порядку переданных хостов.

Примеры

Создание узла сети

Создайте узел сети с именем ”Linux server” с IP-интерфейсом и тегами, добавьте его в группу, привяжите к нему шаблон и задайте MAC-адреса в инвентаре узла сети.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "107819"
        ]
    },
    "id": 1
}
```

Создание узла сети с интерфейсом SNMP

Создайте узел сети под названием ”SNMP host”, указав интерфейс SNMPv3 и детальную информацию.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10658"
        ]
    },
    "id": 1
}
```

Создание узла сети с шифрованием PSK

Создайте узел сети с именем "PSK host" и настроенным шифрованием PSK. Обратите внимание, что узел сети должен быть предварительно настроен для использования PSK.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

Создание узла сети, контролируемого прокси

Создайте узел сети, который будет контролироваться прокси с ID "1".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
```

```
        "host": "Host monitored by proxy",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 1,
        "proxyid": 1
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10591"
        ]
    },
    "id": 1
}
```

Создание узла сети, контролируемого группой прокси

Создайте узел сети, который будет контролироваться группой прокси с ID ”1”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Host monitored by proxy group",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 2,
        "proxy_groupid": 1
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10592"
        ]
    },
    "id": 1
}
```

См. также

- Группа узлов сети
- Шаблон
- Пользовательский макрос
- Интерфейс узла сети
- Инвентаризация узла сети
- Тег узла сети
- Прокси

- Группа прокси

Источник

CHost::create() в *ui/include/classes/api/services/CHost.php*.

**host.delete**

Описание

`object host.delete( hosts)`

Этот метод позволяет удалять узлы сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(array)` ID хостов для удаления.

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы удаленных хостов под свойством hostids.

Примеры

Удаление нескольких узлов сети

Удалите два узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.delete",
    "params": [
        "13",
        "32"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "13",
            "32"
        ]
    },
    "id": 1
}
```

Источник

CHost::delete() в *ui/include/classes/api/services/CHost.php*.

**host.get**

Описание

`integer/array host.get( )`

Метод позволяет получить хосты по заданным параметрам.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groupids | ID/array | Возвращать только узлы сети, которые принадлежат указанным группам. |
| dserviceids | ID/array | Возвращать только узлы сети, связанные с указанными обнаруженными службами. |
| graphids | ID/array | Возвращать только узлы сети, у которых есть указанные графики. |
| hostids | ID/array | Возвращать только узлы сети с указанными ID узлов сети. |
| httptestids | ID/array | Возвращать только узлы сети, у которых есть указанные веб-проверки. |
| interfaceids | ID/array | Возвращать только узлы сети, которые используют указанные интерфейсы. |
| itemids | ID/array | Возвращать только узлы сети, у которых есть указанные элементы данных. |
| maintenanceids | ID/array | Возвращать только узлы сети, на которые влияют указанные обслуживания. |
| monitored_hosts | flag | Возвращать только контролируемые узлы сети. |
| proxyids | ID/array | Возвращать только узлы сети, которые контролируются указанными прокси. |
| proxy_groupids | ID/array | Возвращать только узлы сети, которые контролируются указанными группами прокси. |
| templated_hosts | flag | Возвращать и узлы сети, и шаблоны. |
| templateids | ID/array | Возвращать только узлы сети, связанные с указанными шаблонами. |
| triggerids | ID/array | Возвращать только узлы сети, у которых есть указанные триггеры. |
| with_items | flag | Возвращать только узлы сети, у которых есть элементы данных.<br><br>Переопределяет параметры `with_monitored_items` и `with_simple_graph_items`. |
| with_item_prototypes | flag | Возвращать только узлы сети, у которых есть прототипы элементов данных.<br><br>Переопределяет параметр `with_simple_graph_item_prototypes`. |

| Parameter | Type | Description |
|---|---|---|
| with_simple_graph_item_prototypes | flag | Возвращать только узлы сети, у которых есть прототипы элементов данных, включенные для создания и имеющие числовой тип информации. |
| with_graphs | flag | Возвращать только узлы сети, у которых есть графики. |
| with_graph_prototypes | flag | Возвращать только узлы сети, у которых есть прототипы графиков. |
| with_httptests | flag | Возвращать только узлы сети, у которых есть веб-проверки.<br><br>Переопределяет параметр `with_monitored_httptests`. |
| with_monitored_httptests | flag | Возвращать только узлы сети, у которых есть включенные веб-проверки. |
| with_monitored_items | flag | Возвращать только узлы сети, у которых есть включенные элементы данных.<br><br>Переопределяет параметр `with_simple_graph_items`. |
| with_monitored_triggers | flag | Возвращать только узлы сети, у которых есть включенные триггеры. Все элементы данных, используемые в триггере, также должны быть включены. |
| with_simple_graph_items | flag | Возвращать только узлы сети, у которых есть элементы данных с числовым типом информации. |
| with_triggers | flag | Возвращать только узлы сети, у которых есть триггеры.<br><br>Переопределяет параметр `with_monitored_triggers`. |
| withProblemsSuppressed | boolean | Если установлено в `true`, возвращать только узлы сети с подавленными проблемами. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| severities | integer/array | Возвращать узлы сети, у которых есть только проблемы с указанными уровнями серьезности.<br>Применяется только если объект проблемы является триггером. |

| Parameter | Type | Description |
|---|---|---|
| tags | object/array | Возвращать только узлы сети с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все узлы сети.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| inheritedTags | boolean | Возвращать узлы сети, у которых указанные `tags` также присутствуют во всех связанных шаблонах.<br><br>Возможные значения:<br>`true` - связанные шаблоны также должны иметь указанные `tags`;<br>`false` - *(по умолчанию)* теги связанных шаблонов игнорируются. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения узла сети. Объект обнаружения узла сети связывает обнаруженный узел сети с прототипом узла сети, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`host` - `(string)` ID узла сети;<br>`parent_hostid` - `(string)` ID прототипа узла сети, из которого был создан узел сети;<br>`status` - `(int)` статус обнаружения узла сети:<br>0 - *(по умолчанию)* узел сети обнаружен,<br>1 - узел сети больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда узел сети, который больше не обнаружен, будет удален;<br>`ts_disable` - `(timestamp)` время, когда узел сети, который больше не обнаружен, будет отключен;<br>`disable_source` - `(int)` признак того, был ли узел сети отключен правилом LLD или вручную:<br>0 - *(по умолчанию)* отключен автоматически,<br>1 - отключен правилом LLD. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с правилом низкоуровневого обнаружения, которое создало узел сети (из прототипа узла сети в мониторинге VMware). |

| Parameter | Type | Description |
|---|---|---|
| selectDiscoveryRules | query | Возвращать свойство `discoveryRules` с правилами LLD узла сети.<br><br>Поддерживает `count`. |
| selectGraphs | query | Возвращать свойство `graphs` с графиками узла сети.<br><br>Поддерживает `count`. |
| selectHostGroups | query | Возвращать свойство `hostgroups` с данными групп узлов сети, к которым принадлежит узел сети. |
| selectHttpTests | query | Возвращать свойство `httpTests` с веб-сценариями узла сети.<br><br>Поддерживает `count`. |
| selectInterfaces | query | Возвращать свойство `interfaces` с интерфейсами узла сети.<br><br>Поддерживает `count`. |
| selectInventory | query | Возвращать свойство `inventory` с данными инвентаря узла сети. |
| selectItems | query | Возвращать свойство `items` с элементами данных узла сети.<br><br>Поддерживает `count`. |
| selectMacros | query | Возвращать свойство `macros` с макросами узла сети. |
| selectParentTemplates | query | Возвращать свойство `parentTemplates` с шаблонами, к которым привязан узел сети.<br><br>Помимо полей объекта Template, содержит `link_type` - `(integer)` способ привязки шаблона к узлу сети.<br>Возможные значения:<br>0 - *(по умолчанию)* привязан вручную;<br>1 - автоматически привязан LLD.<br><br>Поддерживает `count`. |
| selectDashboards | query | Возвращать свойство `dashboards`.<br><br>Поддерживает `count`. |
| selectTags | query | Возвращать свойство `tags` с тегами узла сети. |
| selectInheritedTags | query | Возвращать свойство `inheritedTags` с тегами, которые присутствуют во всех шаблонах, связанных с узлом сети. |
| selectTriggers | query | Возвращать свойство `triggers` с триггерами узла сети.<br><br>Поддерживает `count`. |
| selectValueMaps | query | Возвращать свойство `valuemaps` с картами значений узла сети. |

| Parameter | Type | Description |
|---|---|---|
| filter | object | Возвращать только те результаты, которые точно соответствуют указанному фильтру.<br><br>Принимает объект, где ключи - это имена свойств, а значения - либо одно значение, либо массив значений для сопоставления.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства: свойства интерфейса узла сети. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подвыборами.<br><br>Применяется к следующим подвыборам:<br>`selectParentTemplates` - результаты будут отсортированы по `host`;<br>`selectInterfaces`;<br>`selectItems` - сортировка по `name`;<br>`selectDiscoveryRules` - сортировка по `name`;<br>`selectTriggers` - сортировка по `description`;<br>`selectGraphs` - сортировка по `name`;<br>`selectDashboards` - сортировка по `name`. |
| search | object | Возвращать результаты, соответствующие указанному шаблону (без учета регистра).<br><br>Принимает объект, где ключи - это имена свойств, а значения - строки для поиска. Если дополнительные параметры не указаны, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживает только свойства с типом данных `string` и `text`.<br><br>Поддерживает дополнительные свойства: свойства интерфейса узла сети. |
| searchInventory | object | Возвращать узлы сети, у которых данные инвентаря соответствуют указанному шаблону (без учета регистра).<br><br>Принимает объект, где ключи - это имена свойств, а значения - строки для поиска. Если дополнительные параметры не указаны, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживает только свойства с типом данных `string` и `text`. |

| Parameter | Type | Description |
|---|---|---|
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `hostid`, `host`, `name`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |
| selectDiscoveries | query | Возвращать свойство `discoveries` с правилами низкоуровневого обнаружения узла сети.<br><br>Поддерживает `count`.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryRules`. |
| selectHostDiscovery | query | Возвращать свойство `hostDiscovery` с данными объекта обнаружения узла сети.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryData`. |

Возвращаемые значения

`(integer/array)` Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если параметр `countOutput` был использован.

Примеры

Получение данных по имени

Получите все данные о двух узлах сети с именами ”Zabbix server” и ”Linux server”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "filter": {
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "           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
}
```


Получение групп узлов сети

Получите группы узлов сети, членом которых является узел сети "Zabbix server".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectHostGroups": "extend",
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Получение связанных шаблонов

Получите идентификаторы и имена шаблонов, связанных с узлом сети ”10084”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectParentTemplates": [
            "templateid",
            "name"
        ],
        "hostids": "10084"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
```

```
        {
            "hostid": "10084",
            "parentTemplates": [
                {
                    "name": "Linux",
                    "templateid": "10001"
                },
                {
                    "name": "Zabbix Server",
                    "templateid": "10047"
                }
            ]
        }
    ],
    "id": 1
}
```

Получение узлов сети по шаблону

Получите узлы сети, к которым привязан шаблон "10001" (*Linux by Zabbix agent*).

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "name"],
        "templateids": "10001"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Поиск по инвентарным данным узла сети

Получение узлов сети, которые содержат "Linux" в поле "OS" инвентаря узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "host.get",
    "params": {
        "output": [
            "host"
        ],
        "selectInventory": [
            "os"
        ],
        "searchInventory": {
            "os": "Linux"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "inventory": {
                "os": "Linux Ubuntu"
            }
        },
        {
            "hostid": "10107",
            "host": "Linux server",
            "inventory": {
                "os": "Linux Mint"
            }
        }
    ],
    "id": 1
}
```

Поиск по тегам узла сети

Получить узлы сети, у которых тег ”host-name” равен ”linux-server”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "linux-server",
                "operator": 1
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
```

```
    "result": [
        {
            "hostid": "10085",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "linux-server",
                    "automatic": "0"
                },
                {
                    "tag": "os",
                    "value": "rhel-7",
                    "automatic": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Получить узлы сети, у которых эти теги заданы не только на уровне узла сети, но и в связанных родительских шаблонах.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "tags": [
            {
                "tag": "os",
                "value": "rhel-7",
                "operator": 1
            }
        ],
        "inheritedTags": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10623",
            "name": "PC room 1"
        },
        {
            "hostid": "10601",
            "name": "Office"
        }
    ],
    "id": 1
}
```

Получить узлы сети, исключая те, у которых есть тег ”component”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
```

```
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "component",
                "value": "",
                "operator": 5
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Поиск узла сети с тегами и тегами шаблона

Получение узла сети с тегами и всеми тегами, связанными с родительскими шаблонами.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "hostids": 10502,
```

```
        "selectTags": ["tag", "value"],
        "selectInheritedTags": ["tag", "value"]
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Поиск хостов по серьезности проблемы

Получить хосты, у которых есть проблемы «катастрофа» (Disaster).

Запрос:

```
{
   "jsonrpc": "2.0",
   "method": "host.get",
   "params": {
   "output": ["name"],
   "severities": 5
   },
   "auth": "038e1d7b1735c6a5436ee9eae095879e",
   "id": 1
}
```

Ответ:

```
{
   "jsonrpc": "2.0",
   "result": [
   {
```

```
   "hostid": "10160",
   "name": "Zabbix server"
   }
   ],
   "id": 1
}
```

Получить хосты с проблемами уровня "средние" (Average) и "высокие" (High).

Запрос:

```
{
   "jsonrpc": "2.0",
   "method": "host.get",
   "params": {
   "output": ["name"],
   "severities": [3, 4]
   },
   "auth": "038e1d7b1735c6a5436ee9eae095879e",
   "id": 1
}
```

Ответ:

```
{
   "jsonrpc": "2.0",
   "result": [
   {
   "hostid": "20170",
   "name": "Database"
   },
   {
   "hostid": "20183",
   "name": "workstation"
   }
   ],
   "id": 1
}
```

Получение элементов данных узла сети

Получите элементы данных для узла сети "Zabbix server". Запрос ограничен свойствами элемента данных `itemid`, `name` и `status`. Из-за большого размера ответа в примере показана только часть элементов данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectItems": ["itemid","name","status"],
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
  "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
}
```


См. также

- Группа узлов сети
- Шаблон
- Пользовательский макрос
- Интерфейс узла сети
- Прокси
- Группа прокси

Источник

CHost::get() в *ui/include/classes/api/services/CHost.php*.

#### host.massadd

Описание

`object host.massadd(          )`

Этот метод позволяет одновременно добавлять несколько связанных объектов во все заданные хосты.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs узлов сети, которые нужно обновить, и объекты, которые нужно добавить ко всем узлам сети.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hosts | object/array | Узлы сети, которые нужно обновить.<br><br>У узлов сети должно быть определено только свойство `hostid`.<br><br>Поведение параметра:<br>- *required* |
| groups | object/array | Группы узлов сети, которые нужно добавить к указанным узлам сети.<br><br>У групп узлов сети должно быть определено только свойство `groupid`. |
| interfaces | object/array | Интерфейсы узлов сети, которые нужно создать для указанных узлов сети. |
| macros | object/array | Пользовательские макросы, которые нужно создать для указанных узлов сети. |

| Parameter | Type | Description |
|---|---|---|
| templates | object/array | Шаблоны, которые нужно привязать к указанным узлам сети.<br><br>У шаблонов должно быть определено только свойство `templateid`. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных хостов. под свойством `hostids`.

Примеры

Добавление макросов

Добавьте два новых макроса к двум узлам сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.massadd",
    "params": {
        "hosts": [
            {
                "hostid": "10160"
            },
            {
                "hostid": "10167"
            }
        ],
        "macros": [
            {
                "macro": "{$TEST1}",
                "value": "MACROTEST1"
            },
            {
                "macro": "{$TEST2}",
                "value": "MACROTEST2",
                "description": "Test description"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10160",
            "10167"
        ]
    },
    "id": 1
}
```

См. также

- host.update
- Группа узлов сети
- Шаблон
- Пользовательский макрос
- Интерфейс узла сети

Источник

CHost::massAdd() в *ui/include/classes/api/services/CHost.php*.

**host.massremove**

Описание

`object host.massremove( )`

Этот метод позволяет удалять связанные объекты с нескольких хостов.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие ID узлов сети, которые нужно обновить, и объекты, которые следует удалить.

| Parameter | Type | Description |
|---|---|---|
| hostids | ID/array | ID узлов сети, которые нужно обновить.<br><br>Поведение параметра:<br>- *required* |
| groupids | ID/array | ID групп узлов сети, из которых нужно удалить указанные узлы сети. |
| interfaces | object/array | Интерфейсы узла сети, которые нужно удалить у указанных узлов сети.<br><br>Объект интерфейса узла сети должен содержать только свойства `ip`, `dns` и `port`. |
| macros | string/array | Пользовательские макросы, которые нужно удалить у указанных узлов сети. |
| templateids | ID/array | ID шаблонов, которые нужно отвязать от указанных узлов сети. |
| templateids_clear | ID/array | ID шаблонов, которые нужно отвязать и очистить у указанных узлов сети. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных хостов. под свойством `hostids`.

Примеры

Отвязка шаблонов

Отвяжите шаблон от двух узлов сети и удалите все сущности, унаследованные от шаблона.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.massremove",
    "params": {
        "hostids": ["69665", "69666"],
        "templateids_clear": "325"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "69665",
            "69666"
        ]
    },
    "id": 1
}
```

См. также

- host.update
- Пользовательский макрос
- Интерфейс узла сети

Источник

CHost::massRemove() в *ui/include/classes/api/services/CHost.php*.

### host.massupdate

Описание

`object host.massupdate(          )`

Этот метод позволяет одновременно заменять или удалять связанные объекты и обновлять свойства на нескольких хостах.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs узлов сети, которые нужно обновить, и свойства, которые следует обновить.

Помимо стандартных свойств узла сети, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| hosts | object/array | Узлы сети, которые нужно обновить.<br><br>Узлы сети должны иметь определенным только свойство `hostid`.<br><br>Поведение параметра:<br>- *required* |
| groups | object/array | Группы узлов сети, которыми нужно заменить текущие группы узлов сети, к которым принадлежат узлы сети.<br><br>У групп узлов сети должно быть определено только свойство `groupid`. |
| interfaces | object/array | Интерфейсы узлов сети, которыми нужно заменить текущие интерфейсы узлов сети на указанных узлах сети. |
| inventory | object | Свойства инвентаря узла сети.<br><br>Режим инвентаря узла сети нельзя обновить с помощью параметра `inventory`, используйте вместо него `inventory_mode`. |
| macros | object/array | Пользовательские макросы, которыми нужно заменить текущие пользовательские макросы на указанных узлах сети. |

| Parameter | Type | Description |
|---|---|---|
| templates | object/array | Шаблоны, которыми нужно заменить текущие связанные шаблоны на указанных узлах сети.<br><br>У шаблонов должно быть определено только свойство `templateid`. |
| templates_clear | object/array | Шаблоны, которые нужно отвязать и очистить на указанных узлах сети.<br><br>У шаблонов должно быть определено только свойство `templateid`. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных хостов. под свойством `hostids`.

Примеры

Включение нескольких узлов сети

Включите мониторинг двух узлов сети, то есть установите для их статуса значение "0".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.massupdate",
    "params": {
        "hosts": [
            {
                "hostid": "69665"
            },
            {
                "hostid": "69666"
            }
        ],
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "69665",
            "69666"
        ]
    },
    "id": 1
}
```

Смотрите также

- host.update
- host.massadd
- host.massremove
- Группа узлов сети
- Шаблон
- Пользовательский макрос
- Интерфейс узла сети

Источник

CHost::massUpdate() в *ui/include/classes/api/services/CHost.php*.

**host.update**

Описание

`object host.update( / hosts)`

Этот метод позволяет обновлять существующие узлы сети.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства узла сети, которые необходимо обновить.

Свойство `hostid` должно быть определено для каждого узла сети, все остальные свойства необязательны. Будут обновлены только указанные свойства, все остальные останутся без изменений.

Однако обратите внимание, что при обновлении технического имени узла сети также будет обновлено видимое имя узла сети (если оно не задано отдельно) на значение технического имени узла сети.

Помимо стандартных свойств узла сети, метод принимает следующие параметры.

| Параметр | Тип | Описание |
|---|---|---|
| groups | object/array | Группы узлов сети, которые заменят текущие группы узла сети, к которым он принадлежит.<br>Все группы узлов сети, не указанные в запросе, будут отвязаны.<br><br>Группы узлов сети должны содержать только свойство `groupid`. |
| interfaces | object/array | Интерфейсы узла сети, которые заменят текущие интерфейсы узла сети.<br>Все интерфейсы, не указанные в запросе, будут удалены. |
| tags | object/array | Теги узла сети, которые заменят текущие теги узла сети.<br>Все теги, не указанные в запросе, будут удалены. |
| inventory | object | Свойства инвентаря узла сети. |
| macros | object/array | Пользовательские макросы, которые заменят текущие пользовательские макросы.<br>Все макросы, не указанные в запросе, будут удалены. |
| templates | object/array | Шаблоны, которые заменят текущие привязанные шаблоны.<br>Все шаблоны, не указанные в запросе, будут только отвязаны.<br><br>Шаблоны должны содержать только свойство `templateid`. |
| templates_clear | object/array | Шаблоны, которые нужно отвязать и очистить с узла сети.<br><br>Шаблоны должны содержать только свойство `templateid`. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных хостов. под свойством `hostids`.

Примеры

Активация узла сети

Активируйте мониторинг узла сети, установив для него статус ”0”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "status": 0
    },
```

```
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Отвязка шаблонов

Отвязать и очистить два шаблона от узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "templates_clear": [
            {
                "templateid": "10124"
            },
            {
                "templateid": "10125"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Обновление макросов узла сети

Замените все макросы узла сети двумя новыми.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "macros": [
            {
                "macro": "{$PASS}",
                "value": "password"
            },
            {
                "macro": "{$DISC}",
```

```
                "value": "sda",
                "description": "Updated description"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Обновление инвентаря узла сети

Изменение режима инвентаря и добавление местоположения

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "inventory_mode": 0,
        "inventory": {
            "location": "Latvia, Riga"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Обновление тегов узла сети

Замените все теги узла сети новым тегом.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "tags": {
            "tag": "os",
            "value": "rhel-7"
        }
    },
```

```
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Обновление обнаруженных макросов узла сети

Преобразование созданного правилом обнаружения ”автоматического” макроса в ”ручной” и изменение его значения на ”new-value”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "macros": {
            "hostmacroid": "5541",
            "value": "new-value",
            "automatic": "0"
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Обновление шифрования узла сети

Обновите узел сети ”10590”, чтобы использовать только PSK-шифрование для соединений от узла сети к серверу Zabbix, и измените идентификатор PSK и ключ PSK. Обратите внимание, что узел сети должен быть предварительно настроен на использование PSK.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

См. также

- host.massadd
- host.massupdate
- host.massremove
- Группа узлов сети
- Шаблон
- Пользовательский макрос
- Интерфейс узла сети
- Инвентаризация узла сети
- Тег узла сети
- Прокси
- Группа прокси

Источник

CHost::update() в *ui/include/classes/api/services/CHost.php*.

### Шаблон

Этот класс предназначен для работы с шаблонами.

Ссылки на объекты:

- Template
- Template tag

Доступные методы:

- template.create - создать новые шаблоны
- template.delete - удалить шаблоны
- template.get - получить шаблоны
- template.massadd - добавить связанные объекты к шаблонам
- template.massremove - удалить связанные объекты из шаблонов
- template.massupdate - заменить или удалить связанные объекты из шаблонов
- template.update - обновить шаблоны

### Объект шаблона

Следующие объекты напрямую связаны с API `template`.

Шаблон

Объект шаблона имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| templateid | ID | ID шаблона.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| description | text | Описание шаблона. |

| Property | Type | Description |
|---|---|---|
| host | string | Техническое имя шаблона.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| name | string | Отображаемое имя шаблона.<br><br>По умолчанию: значение свойства `host`. |
| readme | text | Инструкции по настройке, специфичные для шаблона, для отображения в Мастере узла сети. Поддерживает форматирование Markdown. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных шаблонов с уже существующими. Если не указан, генерируется автоматически. |
| vendor_name | string | Имя поставщика шаблона.<br><br>Для операций создания оба свойства `vendor_name` и `vendor_version` должны быть либо заданы, либо оставлены пустыми. Для операций обновления `vendor_version` можно оставить пустым, если в базе данных у него есть значение. |
| vendor_version | string | Версия поставщика шаблона.<br><br>Для операций создания оба свойства `vendor_name` и `vendor_version` должны быть либо заданы, либо оставлены пустыми. Для операций обновления `vendor_name` можно оставить пустым, если в базе данных у него есть значение. |
| wizard_ready | integer | Доступен ли шаблон для выбора в Мастере узла сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Недоступен.<br>1 - Доступен. |

Тег шаблона

Объект тега шаблона имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега шаблона.<br>Поведение свойства:<br>- *required* |
| value | string | Значение тега шаблона. |

**template.create**

Описание

`object template.create( / templates)`

Этот метод позволяет создавать новые шаблоны.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Шаблоны для создания.

Дополнительно к стандартным свойствам шаблона метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы шаблонов, к которым нужно добавить шаблон.<br><br>Группы шаблонов должны содержать только свойство `groupid`.<br><br>Поведение параметра:<br>- *required* |
| tags | object/array | Теги шаблона. |
| templates | object/array | Шаблоны, которые нужно связать с шаблоном.<br><br>Шаблоны должны содержать только свойство `templateid`. |
| macros | object/array | Пользовательские макросы, которые нужно создать для шаблона. |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных шаблонов, указанных в свойстве `templateids`. Порядок возвращаемых идентификаторов соответствует порядку переданных шаблонов.

Примеры

Создание шаблона

Создайте шаблон с тегами и свяжите с этим шаблоном два шаблона.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "11117"
        ]
    },
    "id": 1
}
```

Источник

CTemplate::create() в *ui/include/classes/api/services/CTemplate.php*.

**template.delete**

Описание

`object template.delete( templateIds)`

Этот метод позволяет удалять шаблоны.

Удаление шаблона приведет к удалению всех сущностей шаблона (элементы, триггеры, графики и т.д.). Чтобы оставить сущности шаблона на хостах, но удалить сам шаблон, сначала отвяжите шаблон от нужных хостов, используя один из следующих методов: template.update, template.massupdate, host.update, host.massupdate.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

( ) ID удаляемых шаблонов.

Возвращаемые значения

( ) Возвращает объект, который содержит ID удалённых шаблонов, указанных в свойстве `templateids`.

Примеры

Удаление нескольких шаблонов

Удаление двух шаблонов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.delete",
    "params": [
        "13",
        "32"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "13",
            "32"
        ]
    },
    "id": 1
}
```

Источник

CTemplate::delete() в *ui/include/classes/api/services/CTemplate.php*.

**template.get**

Описание

`integer/array template.get( )`

Этот метод позволяет получать шаблоны в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| templateids | ID/array | Возвращать только шаблоны с указанными ID шаблонов. |
| groupids | ID/array | Возвращать только шаблоны, которые принадлежат указанным группам шаблонов. |
| parentTemplateids | ID/array | Возвращать только шаблоны, к которым привязан указанный шаблон. |
| hostids | ID/array | Возвращать только шаблоны, которые привязаны к указанным узлам сети/шаблонам. |
| graphids | ID/array | Возвращать только шаблоны, которые содержат указанные графики. |
| itemids | ID/array | Возвращать только шаблоны, которые содержат указанные элементы данных. |
| triggerids | ID/array | Возвращать только шаблоны, которые содержат указанные триггеры. |
| with_items | flag | Возвращать только шаблоны, у которых есть элементы данных. |
| with_triggers | flag | Возвращать только шаблоны, у которых есть триггеры. |
| with_graphs | flag | Возвращать только шаблоны, у которых есть графики. |
| with_httptests | flag | Возвращать только шаблоны, у которых есть веб-сценарии. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* И/ИЛИ;<br>2 - ИЛИ. |
| tags | object/array | Возвращать только шаблоны с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все шаблоны.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Содержит;<br>1 - Равно;<br>2 - Не содержит;<br>3 - Не равно;<br>4 - Существует;<br>5 - Не существует. |
| selectTags | query | Возвращать теги шаблона в свойстве `tags`. |
| selectDiscoveryRules | query | Возвращать свойство `discoveryRules` с правилами LLD шаблона.<br><br>Поддерживает `count`. |
| selectHosts | query | Возвращать узлы сети, привязанные к шаблону, в свойстве `hosts`.<br><br>Поддерживает `count`. |
| selectTemplateGroups | query | Возвращать группы шаблонов, к которым принадлежит шаблон, в свойстве `templategroups`. |
| selectTemplates | query | Возвращать шаблоны, к которым привязан указанный шаблон, в свойстве `templates`.<br><br>Поддерживает `count`. |
| selectParentTemplates | query | Возвращать шаблоны, которые привязаны к указанному шаблону, в свойстве `parentTemplates`.<br><br>Поддерживает `count`. |

| Parameter | Type | Description |
|---|---|---|
| selectHttpTests | query | Возвращать веб-сценарии из шаблона в свойстве `httpTests`.<br><br>Поддерживает `count`. |
| selectItems | query | Возвращать элементы данных из шаблона в свойстве `items`.<br><br>Поддерживает `count`. |
| selectTriggers | query | Возвращать триггеры из шаблона в свойстве `triggers`.<br><br>Поддерживает `count`. |
| selectGraphs | query | Возвращать графики из шаблона в свойстве `graphs`.<br><br>Поддерживает `count`. |
| selectMacros | query | Возвращать макросы из шаблона в свойстве `macros`. |
| selectDashboards | query | Возвращать дашборды из шаблона в свойстве `dashboards`.<br><br>Поддерживает `count`. |
| selectValueMaps | query | Возвращать свойство `valuemaps` с картами значений шаблона. |
| limitSelects | integer | Ограничивает число записей, возвращаемых подвыборками.<br><br>Применяется к следующим подвыборкам:<br>`selectTemplates` - результаты будут отсортированы по `name`;<br>`selectHosts` - по `host`;<br>`selectParentTemplates` - по `host`;<br>`selectItems` - по `name`;<br>`selectDiscoveryRules` - по `name`;<br>`selectTriggers` - по `description`;<br>`selectGraphs` - по `name`;<br>`selectDashboards` - по `name`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `hostid`, `host`, `name`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| 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 | Возвращать низкоуровневые обнаружения из шаблона в свойстве `discoveries`.<br><br>Поддерживает `count`.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryRules`. |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение шаблонов по имени

Получите все данные о двух шаблонах с именами ”Linux” и ”Windows”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent",
                "Windows by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "              Linux.                Zabbix 7.4       .\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": "7.4-2",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "wizard_ready": "1",
            "readme": "##    \r\n\r\n              Linux.                Zabbix 7.4       .\r\n\r\n####
        },
        {
            "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": "              Windows.                 Zabbix 7.4       .\r\n\r\n
            "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": "7.4-2",
            "proxy_groupid": "0",
            "monitored_by": "0",
            "wizard_ready": "1",
            "readme": "##    \r\n\r\n              Windows.                 Zabbix 7.4       ."
        }
    ],
    "id": 1
}
```


Получение групп шаблонов

Получите группы шаблонов, членом которых является шаблон "Linux by Zabbix agent".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTemplateGroups": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Ответ:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "templategroups": [
                {
                    "groupid": "10",
                    "name": "Templates/Operating systems",
                    "uuid": "846977d1dfed4968bc5f8bdb363285bc"
                }
            ]
        }
    ],
    "id": 1
}
```


Получение узлов сети по шаблону

Получите узлы сети, к которым привязан шаблон ”10001” (*Linux by Zabbix agent*).

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "templateid",
        "templateids": "10001",
        "selectHosts": ["hostid", "name"]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Поиск по тегам шаблона

Получить шаблоны, у которых тег ”host-name” равен ”{HOST.NAME}”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}",
                "operator": 1
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10402",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "{HOST.NAME}"
                }
            ]
        }
    ],
    "id": 1
}
```

См. также

- Группа шаблонов
- Шаблон
- Пользовательский макрос
- Интерфейс узла сети

Источник

CTemplate::get() в *ui/include/classes/api/services/CTemplate.php*.

**template.massadd**

Описание

`object template.massadd(  )`

Этот метод позволяет одновременно добавить несколько связанных объектов к указанным шаблонам.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs шаблонов для обновления и объекты, которые нужно добавить в шаблоны.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| templates | object/array | Шаблоны, которые нужно обновить.<br><br>Шаблоны должны содержать только свойство `templateid`.<br><br>Поведение параметра:<br>- *required* |
| groups | object/array | Группы шаблонов, в которые нужно добавить указанные шаблоны.<br><br>Группы шаблонов должны содержать только свойство `groupid`. |
| macros | object/array | Пользовательские макросы, которые нужно создать для указанных шаблонов. |
| templates_link | object/array | Шаблоны, которые нужно связать с указанными шаблонами.<br><br>Шаблоны должны содержать только свойство `templateid`. |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных шаблонов, указанных в свойстве `templateids`.

Примеры

Привязать группу к шаблонам

Добавить группу шаблонов ”2” к двум шаблонам.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10085"
            },
            {
                "templateid": "10086"
            }
        ],
        "groups": [
            {
                "groupid": "2"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Связать два шаблона с шаблоном

Связать шаблоны ”10106” и ”10104” с шаблоном ”10073”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10073"
            }
        ],
        "templates_link": [
            {
                "templateid": "10106"
            },
            {
                "templateid": "10104"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10073"
        ]
    },
    "id": 1
}
```

См. также

- template.update
- Узел сети
- Группа шаблонов
- Пользовательский макрос

Источник

CTemplate::massAdd() в *ui/include/classes/api/services/CTemplate.php*.

**template.massremove**

Описание

`object templategroup.massremove(          )`

Этот метод позволяет удалить связанные объекты из нескольких шаблонов.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs шаблонов, которые нужно обновить, и объекты, которые следует удалить.

| Parameter | Type | Description |
|---|---|---|
| templateids | ID/array | IDs шаблонов, которые нужно обновить.<br><br>Поведение параметра:<br>- *required* |
| groupids | ID/array | IDs групп шаблонов, из которых нужно удалить указанные шаблоны. |
| macros | string/array | IDs пользовательских макросов, которые нужно удалить из указанных шаблонов. |
| templateids_clear | ID/array | IDs шаблонов, которые нужно отвязать и очистить в указанных шаблонах (upstream). |
| templateids_link | ID/array | IDs шаблонов, которые нужно отвязать от указанных шаблонов (upstream). |

Возвращаемые значения

`(     )` Возвращает объект, который содержит ID обновленных шаблонов, указанных в свойстве `templateids`.

Примеры

Удаление шаблонов из группы

Удалите два шаблона из группы ”2”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": [
            "10085",
            "10086"
        ],
        "groupids": "2"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Отвязка шаблонов от узла сети

Отвязать шаблоны ”10106” и ”10104” от шаблона ”10085”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": "10085",
        "templateids_link": [
            "10106",
            "10104"
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085"
        ]
    },
    "id": 1
}
```

См. также

- template.update
- Пользовательский макрос

Источник

CTemplate::massRemove() в *ui/include/classes/api/services/CTemplate.php*.

**template.massupdate**

Описание

`object template.massupdate( )`

Этот метод позволяет одновременно заменить или удалить связанные объекты и обновить свойства в нескольких шаблонах.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, содержащие IDs шаблонов для обновления и объекты для замены для шаблонов.

Метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| templates | object/array | Шаблоны для обновления.<br><br>Шаблоны должны содержать только свойство `templateid`.<br><br>Поведение параметра:<br>- *required* |
| groups | object/array | Группы шаблонов для замены текущих групп шаблонов, к которым принадлежат шаблоны.<br><br>Группы шаблонов должны содержать только свойство `groupid`. |
| macros | object/array | Пользовательские макросы для замены всех текущих пользовательских макросов в указанных шаблонах. |
| templates_clear | object/array | Шаблоны для отвязки и очистки от указанных шаблонов.<br><br>Шаблоны должны содержать только свойство `templateid`. |
| templates_link | object/array | Шаблоны для замены текущих связанных шаблонов.<br><br>Шаблоны должны содержать только свойство `templateid`. |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID обновленных шаблонов, указанных в свойстве `templateids`.

Примеры

Отвязка шаблона

Отвяжите и очистите шаблон ”10091” от указанных шаблонов.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.massupdate",
    "params": {
        "templates": [
            {
                "templateid": "10085"
            },
            {
                "templateid": "10086"
            }
        ],
        "templates_clear": [
            {
                "templateid": "10091"
            }
```

```
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Замена пользовательских макросов

Замените все пользовательские макросы указанным пользовательским макросом в нескольких шаблонах.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.massupdate",
    "params": {
        "templates": [
            {
                "templateid": "10074"
            },
            {
                "templateid": "10075"
            },
            {
                "templateid": "10076"
            },
            {
                "templateid": "10077"
            }
        ],
        "macros": [
            {
                "macro": "{$AGENT.TIMEOUT}",
                "value": "5m",
                "description": "  -  ,                          .                ,          /    Zabbix
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10074",
            "10075",
            "10076",
            "10077"
        ]
    },
    "id": 1
```

```
}
```

См. также

- template.update
- template.massadd
- Группа шаблонов
- Пользовательский макрос

Источник

CTemplate::massUpdate() в *ui/include/classes/api/services/CTemplate.php*.

**template.update**

Описание

`object template.update(  /  templates)`

Этот метод позволяет обновлять существующие шаблоны.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства шаблона, которые необходимо обновить.

Для каждого шаблона должно быть определено свойство `templateid`, все остальные свойства необязательны. Будут обновлены только указанные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам шаблона, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| groups | object/array | Группы шаблонов, которые заменят текущие группы шаблонов, к которым принадлежат шаблоны.<br><br>Группы шаблонов должны иметь определенным только свойство `groupid`. |
| tags | object/array | Теги шаблона, которые заменят текущие теги шаблона. |
| macros | object/array | Пользовательские макросы, которые заменят текущие пользовательские макросы для указанных шаблонов. |
| templates | object/array | Шаблоны, которые заменят текущие связанные шаблоны. Шаблоны, которые не переданы, будут только отвязаны.<br><br>У шаблонов должно быть определено только свойство `templateid`. |
| templates_clear | object/array | Шаблоны, которые нужно отвязать и очистить от указанных шаблонов.<br><br>У шаблонов должно быть определено только свойство `templateid`. |

Возвращаемые значения

`(  )` Возвращает объект, который содержит ID обновленных шаблонов, указанных в свойстве `templateids`.

Примеры

Изменение стандартных свойств шаблона

Измените техническое имя шаблона на ”Linux by Zabbix agent Custom”, видимое имя на ”My template” и обновите описание шаблона.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Обновление групп шаблона

Заменяет все группы шаблона для указанного шаблона на другую.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "groups": [
            {
                "groupid": "24"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Обновление тегов шаблона

Замените все теги шаблона на другие.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "tags": [
```

```
            {
                "tag": "host-name",
                "value": "{HOST.NAME}"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Обновление макросов шаблона

Заменяет все макросы шаблона на другой.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "macros": [
            {
                "macro": "{$MY_MACRO}",
                "value": "new_value"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Обновление связанных шаблонов

Отвяжите (без очистки) все шаблоны от указанного шаблона и привяжите к нему другой шаблон.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "templates": [
            {
                "templateid": "10087"
```

```
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Очистка связанных шаблонов шаблона

Отвязать и очистить указанный шаблон от конкретного связанного шаблона.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "templates_clear": [
            {
                "templateid": "10087"
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Источник

CTemplate::update() в *ui/include/classes/api/services/CTemplate.php*.

**Шаблон дашборда**

Этот класс предназначен для работы с шаблонными дашбордами.

Ссылки на объекты:

- Шаблон дашборда
- Страница шаблонного дашборда
  - Виджет шаблонного дашборда
    - Поле виджета шаблонного дашборда

Доступные методы:

- templatedashboard.create - создать новые шаблонные дашборды

- templatedashboard.delete - удалить шаблонные дашборды
- templatedashboard.get - получить шаблонные дашборды
- templatedashboard.update - обновить шаблонные дашборды

**Объект шаблона панели**

Следующие объекты напрямую связаны с API `templatedashboard`.

Панель шаблона

Объект панели шаблона имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dashboardid | ID | ID панели шаблона.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| name | string | Имя панели шаблона.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| templateid | ID | ID шаблона, которому принадлежит панель.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |
| display_period | integer | Период отображения страницы по умолчанию (в секундах).<br><br>Возможные значения: 10, 30, 60, 120, 600, 1800, 3600.<br><br>По умолчанию: 30. |
| auto_start | integer | Автоматически запускать слайд-шоу.<br><br>Возможные значения:<br>0 - не запускать слайд-шоу автоматически;<br>1 - *(по умолчанию)* запускать слайд-шоу автоматически. |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированных панелей шаблонов с уже существующими. Если не указан, генерируется автоматически. |

Страница панели мониторинга шаблона

Объект страницы панели мониторинга шаблона имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| dashboard_pageid | ID | ID страницы панели мониторинга.<br><br>Поведение свойства:<br>- *только для чтения* |
| name | string | Имя страницы панели мониторинга.<br><br>По умолчанию: пустая строка. |
| display_period | integer | Период отображения страницы панели мониторинга (в секундах).<br><br>Допустимые значения: 0, 10, 30, 60, 120, 600, 1800, 3600.<br><br>По умолчанию: 0 (будет использоваться период отображения страницы по умолчанию). |
| widgets | array | Массив объектов виджета панели мониторинга шаблона. |

Виджет панели мониторинга шаблона

Объект виджета панели мониторинга шаблона имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| widgetid | ID | ID виджета панели мониторинга.<br><br>Поведение свойства:<br>- *только для чтения* |
| type | string | Тип виджета панели мониторинга.<br><br>Возможные значения:<br>actionlog - Журнал действий;<br>clock - Часы;<br>discovery - Статус обнаружения;<br>favgraphs - Избранные графики;<br>favmaps - Избранные карты;<br>gauge - Индикатор;<br>graph - График (классический);<br>graphprototype - Прототип графика;<br>honeycomb - Honeycomb;<br>hostavail - Доступность узла сети;<br>hostcard - Карточка узла сети;<br>hostnavigator - Навигатор узлов сети;<br>itemcard - Карточка элемента данных;<br>itemnavigator - Навигатор элементов данных;<br>item - Значение элемента данных;<br>map - Карта;<br>navtree - Дерево навигации по картам;<br>piechart - Круговая диаграмма;<br>plaintext - Обычный текст;<br>problemhosts - Узлы сети с проблемами;<br>problems - Проблемы;<br>problemsbysv - Проблемы по степени важности;<br>slareport - Отчет SLA;<br>svggraph - График;<br>systeminfo - Системная информация;<br>tophosts - Топ узлов сети;<br>topitems - Топ элементов данных;<br>toptriggers - Топ триггеров;<br>trigover - Обзор триггеров;<br>url - URL;<br>web - Веб-мониторинг.<br><br>Поведение свойства:<br>- *обязательно* |
| name | string | Пользовательское имя виджета. |
| x | integer | Горизонтальная позиция от левой стороны панели мониторинга.<br><br>Возможные значения: от 0 до 71. |
| y | integer | Вертикальная позиция от верхней стороны панели мониторинга.<br><br>Возможные значения: от 0 до 63. |
| width | integer | Ширина виджета.<br><br>Возможные значения: от 1 до 72. |
| height | integer | Высота виджета.<br><br>Возможные значения: от 1 до 64. |
| view_mode | integer | Режим отображения виджета.<br><br>Возможные значения:<br>0 - *(по умолчанию)* стандартный вид виджета;<br>1 - со скрытым заголовком; |

| Property | Type | Description |
|---|---|---|
| fields | array | Массив объектов поле виджета панели мониторинга шаблона. |

Поле виджета панели шаблона

Объект поля виджета панели шаблона имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип поля виджета.<br><br>Возможные значения:<br>0 - Integer;<br>1 - String;<br>4 - Item;<br>5 - Item prototype;<br>6 - Graph;<br>7 - Graph prototype;<br>8 - Map;<br>9 - Service;<br>10 - SLA;<br>11 - User;<br>12 - Action;<br>13 - Media type.<br><br>Property behavior:<br>- *required* |
| name | string | Имя поля виджета.<br><br>Возможные значения: см. Dashboard widget fields. Обратите внимание, что некоторые параметры, связанные с узлом сети (например, *Host groups*, *Exclude host groups* и *Hosts* в виджете *Problems*, *Host groups* в виджете *Host availability* и т. д.), недоступны при настройке виджета на панели шаблона. Это связано с тем, что панели шаблона отображают данные только с узла сети, к которому привязан шаблон.<br><br>Property behavior:<br>- *required* |
| value | mixed | Значение поля виджета в зависимости от типа.<br><br>Возможные значения: см. Dashboard widget fields. Обратите внимание, что некоторые параметры, связанные с узлом сети (например, *Host groups*, *Exclude host groups* и *Hosts* в виджете *Problems*, *Host groups* в виджете *Host availability* и т. д.), недоступны при настройке виджета на панели шаблона. Это связано с тем, что панели шаблона отображают данные только с узла сети, к которому привязан шаблон.<br><br>Property behavior:<br>- *required* |

**templatedashboard.create**

Описание

`object templatedashboard.create( / templateDashboards)`

Этот метод позволяет создавать новые шаблоны информационных панелей.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Шаблоны дашбордов для создания.

В дополнение к стандартным свойствам шаблона дашборда, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| pages | array | Страницы шаблона дашборда для создания дашборда. Страницы дашборда будут упорядочены в том же порядке, в котором они указаны.<br><br>Поведение параметра:<br>- *required* |

Возвращаемые значения

( ) Возвращает объект, который содержит ID созданных шаблонов панелей, указанных в свойстве dashboardids. Порядок возвращаемых идентификаторов соответствует порядку переданных шаблонов панелей.

Примеры

Создание шаблона панели

Создайте шаблон панели с именем “Graphs” с одним виджетом Graph на одной странице панели.

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "32"
        ]
    },
    "id": 1
}
```

См. также

- Страница панели шаблона
- Виджет панели шаблона
- Поле виджета панели шаблона

Источник

CTemplateDashboard::create() в *ui/include/classes/api/services/CTemplateDashboard.php*.

**templatedashboard.delete**

Описание

`object templatedashboard.delete(  templateDashboardIds)`

Этот метод позволяет удалять панели шаблонов.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(  ) ID удаляемых шаблонов панелей.

Возвращаемые значения

(  ) Возвращает объект, который содержит ID удаленных шаблонов панелей, указанных в свойстве `dashboardids`.

Примеры

Удаление нескольких панелей шаблона

Удаление двух панелей шаблона.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.delete",
    "params": [
        "45",
        "46"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "45",
            "46"
        ]
    },
```

```
    "id": 1
}
```

Источник

CTemplateDashboard::delete() в *ui/include/classes/api/services/CTemplateDashboard.php*.

### templatedashboard.get

Описание

`integer/array templatedashboard.get(          )`

Метод позволяет получить шаблоны панелей мониторинга в соответствии с заданными параметрами.

> **Note:**
> Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| dashboardids | ID/array | Возвращать только шаблонные панели мониторинга с указанными ID. |
| templateids | ID/array | Возвращать только шаблонные панели мониторинга, принадлежащие указанным шаблонам. |
| selectPages | query | Возвращать свойство `pages` с страницами шаблонной панели мониторинга, корректно упорядоченными. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `dashboardid`, `name`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| filter | object | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |

Возвращаемые значения

`(        /     )` Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение дашбордов шаблона

Получить все дашборды шаблона с виджетами для указанного шаблона.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.get",
```

```
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "templateids": "10001"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```


См. также

- Страница панели шаблона
- Виджет панели шаблона
- Поле виджета панели шаблона

Источник

CTemplateDashboard::get() в *ui/include/classes/api/services/CTemplateDashboard.php*.

**templatedashboard.update**

Описание

`object templatedashboard.update( / templateDashboards)`

Этот метод позволяет обновлять существующие шаблоны информационных панелей.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства шаблона панели мониторинга, которые необходимо обновить.

Для каждой панели мониторинга необходимо указать свойство `dashboardid`, все остальные свойства необязательны. Будут обновлены только указанные свойства.

Дополнительно к стандартным свойствам шаблона панели мониторинга, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| pages | array | Страницы шаблона панели мониторинга для замены существующих страниц панели мониторинга.<br><br>Страницы панели мониторинга обновляются по свойству `dashboard_pageid`. Новые страницы панели мониторинга будут созданы для объектов без свойства `dashboard_pageid`, а существующие страницы панели мониторинга будут удалены, если не используются повторно. Страницы панели мониторинга будут упорядочены в том же порядке, в котором они указаны. Будут обновлены только указанные свойства страниц панели мониторинга. Для свойства `pages` требуется как минимум один объект страницы панели мониторинга. |

Возвращаемые значения

`(object)` Возвращает объект, содержащий идентификаторы обновленных панелей шаблона в свойстве `dashboardids`.

Примеры

Переименование панели шаблона

Переименуйте панель шаблона в ”Performance graphs”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.update",
    "params": {
        "dashboardid": "23",
        "name": "Performance graphs"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "23"
```

```
        ]
    },
    "id": 1
}
```

Обновление страниц панели шаблона

Переименуйте первую страницу панели, замените виджеты на второй странице панели и добавьте новую страницу в качестве третьей. Удалите все остальные страницы панели.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

См. также

- Виджет панели шаблона
- Поле виджета панели шаблона

Источник

CTemplateDashboard::update() в *ui/include/classes/api/services/CTemplateDashboard.php*.

**Элемент данных**

Этот класс предназначен для работы с элементами данных.

Ссылки на объекты:

- Элемент данных
  - HTTP header
  - HTTP query field
- Тег элемента данных
- Предобработка элемента данных

Доступные методы:

- item.create - создать новые элементы данных
- item.delete - удалить элементы данных
- item.get - получить элементы данных
- item.update - обновить элементы данных

#### Объект элемента данных

Следующие объекты напрямую связаны с API `item`.

Элемент данных

**Note:**
Веб-элементы данных нельзя напрямую создавать, обновлять или удалять через Zabbix API.

Объект элемента данных имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| itemid | ID | ID элемента данных.<br><br>Поведение свойства:<br>- *только для чтения*<br>- *обязательно* для операций обновления |
| delay | string | Интервал обновления элемента данных.<br><br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m, 2h, 1d) и, дополнительно, один или несколько пользовательских интервалов, разделенных точкой с запятой. Пользовательские интервалы могут представлять собой сочетание гибких и расписанных интервалов.<br><br>Принимает пользовательские макросы. Если используется, значение должно быть одним макросом. Несколько макросов или макросы, смешанные с текстом, не поддерживаются. Гибкие интервалы можно записывать как два макроса, разделенные косой чертой (например, `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`).<br><br>Пример:<br>`1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr`<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”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), или если `type` установлен в ”Zabbix agent (active)” (7) и `key_` не содержит ”mqtt.get” |
| hostid | ID | ID узла сети или шаблона, которому принадлежит элемент данных.<br><br>Поведение свойства:<br>- *константа*<br>- *обязательно* для операций создания |

| Property | Type | Description |
|---|---|---|
| interfaceid | ID | ID интерфейса узла сети для элемента данных.<br><br>Поведение свойства:<br>- *обязательно* если элемент данных принадлежит узлу сети и `type` установлен в ”Zabbix agent”, ”IPMI agent”, ”JMX agent”, ”SNMP trap” или ”SNMP agent”<br>- *поддерживается* если элемент данных принадлежит узлу сети и `type` установлен в ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” или ”HTTP agent”<br>- *только для чтения* для обнаруженных объектов |
| key_ | string | Ключ элемента данных.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| name | string | Имя элемента данных.<br>Поддерживает пользовательские макросы.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| name_resolved | string | Имя элемента данных с разрешенными пользовательскими макросами.<br><br>Поведение свойства:<br>- *только для чтения* |
| type | integer | Тип элемента данных.<br><br>Возможные значения:<br>0 - Zabbix agent;<br>2 - Zabbix trapper;<br>3 - Simple check;<br>5 - Zabbix internal;<br>7 - Zabbix agent (active);<br>9 - Web item;<br>10 - External check;<br>11 - Database monitor;<br>12 - IPMI agent;<br>13 - SSH agent;<br>14 - TELNET agent;<br>15 - Calculated;<br>16 - JMX agent;<br>17 - SNMP trap;<br>18 - Dependent item;<br>19 - HTTP agent;<br>20 - SNMP agent;<br>21 - Script;<br>22 - Browser.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |

| Property | Type | Description |
|---|---|---|
| url | string | Строка URL.<br>Поддерживает пользовательские макросы, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| value_type | integer | Тип информации элемента данных.<br><br>Возможные значения:<br>0 - numeric float;<br>1 - character;<br>2 - log;<br>3 - numeric unsigned;<br>4 - text;<br>5 - binary.<br><br>Поведение свойства:<br>- *обязательно* для операций создания<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| allow_traps | integer | Разрешить заполнять значение аналогично элементу данных типа trapper.<br><br>0 - *(по умолчанию)* Не разрешать принимать входящие данные;<br>1 - Разрешить принимать входящие данные.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для обнаруженных объектов |
| authtype | integer | Метод аутентификации.<br><br>Возможные значения если `type` установлен в ”SSH agent”:<br>0 - *(по умолчанию)* пароль;<br>1 - публичный ключ.<br><br>Возможные значения если `type` установлен в ”HTTP agent”:<br>0 - *(по умолчанию)* нет;<br>1 - basic;<br>2 - NTLM;<br>3 - Kerberos;<br>4 - Digest.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”SSH agent” или ”HTTP agent”<br>- *только для чтения* для унаследованных объектов (если `type` установлен в ”HTTP agent”) или обнаруженных объектов |
| description | string | Описание элемента данных.<br><br>Поведение свойства:<br>- *только для чтения* для обнаруженных объектов |
| error | string | Текст ошибки, если возникли проблемы при обновлении значения элемента данных.<br><br>Поведение свойства:<br>- *только для чтения* |

| Property | Type | Description |
|---|---|---|
| flags | integer | Происхождение элемента данных.<br><br>Возможные значения:<br>0 - обычный элемент данных;<br>4 - элемент данных, преобразованный из прототипа.<br><br>Поведение свойства:<br>- *только для чтения* |
| follow_redirects | integer | Следовать перенаправлениям ответа при опросе данных.<br><br>Возможные значения:<br>0 - Не следовать перенаправлениям;<br>1 - *(по умолчанию)* Следовать перенаправлениям.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| headers | array | Массив заголовков, которые будут отправлены при выполнении HTTP-запроса.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| history | string | Единица времени, определяющая, как долго должны храниться данные истории.<br>Также принимает пользовательский макрос.<br><br>По умолчанию: 31d.<br><br>Поведение свойства:<br>- *только для чтения* для обнаруженных объектов |
| http_proxy | string | Строка подключения к HTTP(S) proxy.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| inventory_link | integer | ID поля инвентаря узла сети, которое заполняется элементом данных.<br><br>См. страницу инвентаря узла сети для списка поддерживаемых полей инвентаря узла сети и их ID.<br><br>По умолчанию: 0.<br><br>Поведение свойства:<br>- *поддерживается* если `value_type` установлен в ”numeric float”, ”character”, ”numeric unsigned” или ”text”<br>- *только для чтения* для обнаруженных объектов |
| ipmi_sensor | string | Датчик IPMI.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”IPMI agent” и `key_` не установлен в ”ipmi.get”<br>- *поддерживается* если `type` установлен в ”IPMI agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |

| Property | Type | Description |
|---|---|---|
| jmx_endpoint | string | Пользовательская строка подключения JMX agent.<br><br>Значение по умолчанию:<br>service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в "JMX agent"<br>- *только для чтения* для обнаруженных объектов |
| lastclock | timestamp | Время, когда значение элемента данных было обновлено в последний раз.<br><br>По умолчанию отображаются только значения, попадающие в последние 24 часа. Вы можете увеличить этот период, изменив значение параметра *Max history display period* в разделе меню *Administration → General*.<br><br>Поведение свойства:<br>- *только для чтения* |
| lastns | integer | Наносекунды, когда значение элемента данных было обновлено в последний раз.<br><br>По умолчанию отображаются только значения, попадающие в последние 24 часа. Вы можете увеличить этот период, изменив значение параметра *Max history display period* в разделе меню *Administration → General*.<br><br>Поведение свойства:<br>- *только для чтения* |
| lastvalue | string | Последнее значение элемента данных.<br><br>По умолчанию отображаются только значения, попадающие в последние 24 часа. Вы можете увеличить этот период, изменив значение параметра *Max history display period* в разделе меню *Administration → General*.<br><br>Поведение свойства:<br>- *только для чтения* |
| logtimefmt | string | Формат времени в записях журнала.<br><br>Поведение свойства:<br>- *поддерживается* если `value_type` установлен в "log"<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| master_itemid | ID | ID мастер-элемента данных.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в "Dependent item"<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| output_format | integer | Следует ли преобразовывать ответ в JSON.<br><br>0 - *(по умолчанию)* Сохранять как есть;<br>1 - Преобразовать в JSON.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в "HTTP agent"<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |

| Property | Type | Description |
|---|---|---|
| params | string | Дополнительные параметры в зависимости от типа элемента данных:<br>- выполняемый сценарий для элементов данных SSH agent и TELNET agent;<br>- SQL-запрос для элементов данных database monitor;<br>- формула для вычисляемых элементов данных;<br>- сценарий для элементов данных script и browser.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”Database monitor”, ”SSH agent”, ”TELNET agent”, ”Calculated”, ”Script” или ”Browser”<br>- *только для чтения* для унаследованных объектов (если `type` установлен в ”Script” или ”Browser”) или обнаруженных объектов |
| parameters | object/array | Дополнительные параметры, если `type` установлен в ”Script” или ”Browser”. Массив объектов со свойствами `name` и `value`, где `name` должен быть уникальным.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”Script” или ”Browser”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| password | string | Пароль для аутентификации.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”JMX agent” и `username` задан<br>- *поддерживается* если `type` установлен в ”Simple check”, ”SSH agent”, ”TELNET agent”, ”Database monitor” или ”HTTP agent”<br>- *только для чтения* для унаследованных объектов (если `type` установлен в ”HTTP agent”) или обнаруженных объектов |
| post_type | integer | Тип тела POST-данных, хранящихся в свойстве `posts`.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Raw data;<br>2 - JSON data;<br>3 - XML data.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| posts | string | Данные тела HTTP(S)-запроса.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”HTTP agent” и `post_type` установлен в ”JSON data” или ”XML data”<br>- *поддерживается* если `type` установлен в ”HTTP agent” и `post_type` установлен в ”Raw data”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| prevvalue | string | Предыдущее значение элемента данных.<br><br>По умолчанию отображаются только значения, попадающие в последние 24 часа. Вы можете увеличить этот период, изменив значение параметра *Max history display period* в разделе меню *Administration → General*.<br><br>Поведение свойства:<br>- *только для чтения* |

| Property | Type | Description |
|---|---|---|
| privatekey | string | Имя файла закрытого ключа.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”SSH agent” и `authtype` установлен в ”public key”<br>- *только для чтения* для обнаруженных объектов |
| publickey | string | Имя файла открытого ключа.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”SSH agent” и `authtype` установлен в ”public key”<br>- *только для чтения* для обнаруженных объектов |
| query_fields | array | Массив полей запроса, которые будут отправлены при выполнении HTTP-запроса.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| request_method | integer | Тип метода запроса.<br><br>Возможные значения:<br>0 - *(по умолчанию)* GET;<br>1 - POST;<br>2 - PUT;<br>3 - HEAD.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| retrieve_mode | integer | Какая часть ответа должна быть сохранена.<br><br>Возможные значения если `request_method` установлен в ”GET”, ”POST” или ”PUT”:<br>0 - *(по умолчанию)* Тело;<br>1 - Заголовки;<br>2 - Будут сохранены и тело, и заголовки.<br><br>Возможные значения если `request_method` установлен в ”HEAD”:<br>1 - Заголовки.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| snmp_oid | string | SNMP OID.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”SNMP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| ssl_cert_file | string | Путь к файлу открытого SSL-ключа.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |

| Property | Type | Description |
|---|---|---|
| ssl_key_file | string | Путь к файлу закрытого SSL-ключа.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| ssl_key_password | string | Пароль для файла SSL-ключа.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| state | integer | Состояние элемента данных.<br><br>Возможные значения:<br>0 - *(по умолчанию)* нормальное;<br>1 - не поддерживается.<br><br>Поведение свойства:<br>- *только для чтения* |
| status | integer | Статус элемента данных.<br><br>Возможные значения:<br>0 - *(по умолчанию)* включенный элемент данных;<br>1 - отключенный элемент данных. |
| status_codes | string | Диапазоны требуемых HTTP-кодов состояния, разделенные запятыми.<br>Также поддерживает пользовательские макросы как часть списка, разделенного запятыми.<br><br>Пример: 200,200-{$M},{$M},200-400<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| templateid | ID | ID родительского элемента данных шаблона.<br><br>*Подсказка*: используйте свойство `hostid`, чтобы указать шаблон, которому принадлежит элемент данных.<br><br>Поведение свойства:<br>- *только для чтения* |
| timeout | string | Тайм-аут запроса опроса данных элемента данных.<br>Принимает секунды или единицу времени с суффиксом (например, 30s, 1m). Также принимает пользовательские макросы.<br><br>Диапазон возможных значений: 1-600s.<br><br>По умолчанию: ”” - использовать настройки proxy/глобальные настройки.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”Zabbix agent” (0), ”Simple check” (3) и `key_` не начинается с ”`vmware.`” и ”`icmpping`”, ”Zabbix agent (active)” (7), ”External check” (10), ”Database monitor” (11), ”SSH agent” (13), ”TELNET agent” (14), ”HTTP agent” (19), ”SNMP agent” (20) и `snmp_oid` начинается с ”`walk[`” или ”`get[`”, ”Script” (21), ”Browser” (22)<br>- *только для чтения* для унаследованных и обнаруженных объектов |

| Property | Type | Description |
|---|---|---|
| trapper_hosts | string | Разрешенные узлы.<br><br>Поведение свойства:<br>- *readonly* для обнаруженных объектов<br>- *поддерживается* если `type` установлен в ”Zabbix trapper”, или если `type` установлен в ”HTTP agent” и `allow_traps` установлен в ”Allow to accept incoming data” |
| trends | string | Единица времени, определяющая, как долго должны храниться данные трендов.<br>Также принимает пользовательский макрос.<br><br>По умолчанию: 365d.<br><br>Поведение свойства:<br>- *поддерживается* если `value_type` установлен в ”numeric float” или ”numeric unsigned”<br>- *только для чтения* для обнаруженных объектов |
| units | string | Единицы значения.<br><br>Поведение свойства:<br>- *поддерживается* если `value_type` установлен в ”numeric float” или ”numeric unsigned”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| username | string | Имя пользователя для аутентификации.<br><br>Поведение свойства:<br>- *обязательно* если `type` установлен в ”SSH agent”, ”TELNET agent” или если `type` установлен в ”JMX agent” и `password` задан<br>- *поддерживается* если `type` установлен в ”Simple check”, ”Database monitor” или ”HTTP agent”<br>- *только для чтения* для унаследованных объектов (если `type` установлен в ”HTTP agent”) или обнаруженных объектов |
| uuid | string | Универсальный уникальный идентификатор, используемый для связывания импортированного элемента данных с уже существующими. Автоматически генерируется, если не указан.<br><br>Поведение свойства:<br>- *поддерживается* если элемент данных принадлежит шаблону |
| valuemapid | ID | ID связанной карты значений.<br><br>Поведение свойства:<br>- *поддерживается* если `value_type` установлен в ”numeric float”, ”character” или ”numeric unsigned”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| verify_host | integer | Следует ли проверять, что имя узла сети для соединения совпадает с именем в сертификате узла сети.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не проверять;<br>1 - Проверять.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в ”HTTP agent”<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |

| Property | Type | Description |
|---|---|---|
| verify_peer | integer | Следует ли проверять, что сертификат узла сети является подлинным.<br><br>Возможные значения:<br>0 - *(по умолчанию)* Не проверять;<br>1 - Проверять.<br><br>Поведение свойства:<br>- *поддерживается* если `type` установлен в "HTTP agent"<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |

HTTP-заголовок

Объект заголовка имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя HTTP-заголовка.<br>Поведение свойства:<br>- *required* |
| value | string | Значение заголовка.<br>Поведение свойства:<br>- *required* |

Поле HTTP-запроса

Объект поля запроса определяет имя и значение, которые используются для указания параметра URL. Он имеет следующие свойства:

| Property | Type | Description |
|---|---|---|
| name | string | Имя параметра.<br>Property behavior:<br>- *required* |
| value | string | Значение параметра.<br>Property behavior:<br>- *required* |

Тег элемента данных

Объект тега элемента данных имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| tag | string | Имя тега элемента данных.<br><br>Поведение свойства:<br>- *required* |
| value | string | Значение тега элемента данных. |

Предобработка элемента данных

Объект предобработки элемента данных имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| type | integer | Тип опции предобработки.<br><br>Возможные значения:<br>1 - Пользовательский множитель;<br>2 - Обрезка справа;<br>3 - Обрезка слева;<br>4 - Обрезка;<br>5 - Регулярное выражение;<br>6 - Булево в десятичное;<br>7 - Восьмеричное в десятичное;<br>8 - Шестнадцатеричное в десятичное;<br>9 - Простое изменение;<br>10 - Изменение за секунду;<br>11 - XML XPath;<br>12 - JSONPath;<br>13 - В диапазоне;<br>14 - Соответствует регулярному выражению;<br>15 - Не соответствует регулярному выражению;<br>16 - Проверка на ошибку в JSON;<br>17 - Проверка на ошибку в XML;<br>18 - Проверка на ошибку с использованием регулярного выражения;<br>19 - Отбрасывать неизменившееся;<br>20 - Отбрасывать неизменившееся с heartbeat;<br>21 - JavaScript;<br>22 - Шаблон Prometheus;<br>23 - Prometheus в JSON;<br>24 - CSV в JSON;<br>25 - Замена;<br>26 - Проверка неподдерживаемого;<br>27 - XML в JSON;<br>28 - Значение SNMP walk;<br>29 - SNMP walk в JSON;<br>30 - Значение SNMP get.<br><br>Property behavior:<br>- *required* |
| params | string | Дополнительные параметры, используемые опцией предобработки.<br>Несколько параметров разделяются символом новой строки (\n).<br><br>Если `type` установлен в ”Check unsupported”, параметры используют синтаксис `<scope>[\n<pattern>]`, где *pattern* - это регулярное выражение, а *scope* - одно из следующих значений:<br>-1 - совпадение с любой ошибкой;<br>0 - проверить, совпадает ли сообщение об ошибке с *pattern*;<br>1 - проверить, не совпадает ли сообщение об ошибке с *pattern*.<br><br>Property behavior:<br>- *required* если `type` установлен в ”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), or ”SNMP get value” (30) |

| Property | Type | Description |
|---|---|---|
| error_handler | integer | Тип действия, используемый в случае сбоя шага предобработки.<br><br>Возможные значения:<br>0 - Сообщение об ошибке задается сервером Zabbix;<br>1 - Отбросить значение;<br>2 - Установить пользовательское значение;<br>3 - Установить пользовательское сообщение об ошибке.<br><br>Возможные значения, если `type` установлен в ”Check unsupported”:<br>1 - Отбросить значение;<br>2 - Установить пользовательское значение;<br>3 - Установить пользовательское сообщение об ошибке.<br><br>Property behavior:<br>- *required* если `type` установлен в ”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), or ”SNMP get value” (30) |
| error_handler_params | string | Параметры обработчика ошибок.<br><br>Property behavior:<br>- *required* если `error_handler` установлен в ”Set custom value” или ”Set custom error message” |

Для каждого типа предобработки поддерживаются следующие параметры и обработчики ошибок.

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 1 | Custom multiplier | number^1, 6^ | | | 0, 1, 2, 3 |
| 2 | Right trim | list of characters[2] | | | |
| 3 | Left trim | list of characters[2] | | | |
| 4 | Trim | list of characters[2] | | | |
| 5 | Regular expression | pattern[3] | output[2] | | 0, 1, 2, 3 |
| 6 | Boolean to decimal | | | | 0, 1, 2, 3 |
| 7 | Octal to decimal | | | | 0, 1, 2, 3 |
| 8 | Hexadecimal to decimal | | | | 0, 1, 2, 3 |

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 9 | Simple change | | | | 0, 1, 2, 3 |
| 10 | Change per second | | | | 0, 1, 2, 3 |
| 11 | XML XPath | path[4] | | | 0, 1, 2, 3 |
| 12 | JSONPath | path[4] | | | 0, 1, 2, 3 |
| 13 | In range | min^1, 6^ | max^1, 6^ | | 0, 1, 2, 3 |
| 14 | Matches regular expression | pattern[3] | | | 0, 1, 2, 3 |
| 15 | Does not match regular expression | pattern[3] | | | 0, 1, 2, 3 |
| 16 | Check for error in JSON | path[4] | | | 0, 1, 2, 3 |
| 17 | Check for error in XML | path[4] | | | 0, 1, 2, 3 |
| 18 | Check for error using regular expression | pattern[3] | output[2] | | 0, 1, 2, 3 |
| 19 | Discard unchanged | | | | |
| 20 | Discard unchanged with heartbeat | seconds^5, 6^ | | | |
| 21 | JavaScript | script[2] | | | |
| 22 | Prometheus pattern | pattern^6, 7^ | `value`, `label`, `function` | output^8, 9^ | 0, 1, 2, 3 |
| 23 | Prometheus to JSON | pattern^6, 7^ | | | 0, 1, 2, 3 |

| Preprocessing type | Name | Parameter 1 | Parameter 2 | Parameter 3 | Supported error handlers |
|---|---|---|---|---|---|
| 24 | CSV to JSON | character[2] | character[2] | 0,1 | 0, 1, 2, 3 |
| 25 | Replace | search string[2] | replacement[2] | | |
| 26 | Check un-sup-ported | scope[1] | pattern^3, 6^ | | 1, 2, 3 |
| 27 | XML to JSON | | | | 0, 1, 2, 3 |
| 28 | SNMP walk value | OID[2] | Format:<br>0 - Unchanged<br>1 - UTF-8 from Hex-STRING<br>2 - MAC from Hex-STRING<br>3 - Integer from BITS | | 0, 1, 2, 3 |
| 29 | SNMP walk to JSON[10] | Field name[2] | OID prefix[2] | Format:<br>0 - Unchanged<br>1 - UTF-8 from Hex-STRING<br>2 - MAC from Hex-STRING<br>3 - Integer from BITS | 0, 1, 2, 3 |
| 30 | SNMP get value | Format:<br>1 - UTF-8 from Hex-STRING<br>2 - MAC from Hex-STRING<br>3 - Integer from BITS | | | 0, 1, 2, 3 |

[1] число с плавающей точкой (целые числа неявно преобразуются в значения с плавающей точкой)<br> [2] строка<br> [3] регулярное выражение<br> [4] JSONPath или XML XPath<br> [5] положительное целое число (с поддержкой временных суффиксов, например 30s, 1m, 2h, 1d)<br> [6] пользовательский макрос<br> [7] шаблон Prometheus, соответствующий синтаксису: `<metric name>{<label name>="<label value>", ...} == <value>`. Каждый компонент шаблона Prometheus (метрика, имя метки, значение метки и значение метрики) может быть пользовательским макросом.<br> [8] вывод Prometheus, соответствующий синтаксису: `<label name>` (может быть пользовательским макросом), если в качестве второго параметра выбрано `label`.<br> [9] одна из агрегирующих функций: `sum`, `min`, `max`, `avg`, `count`, если в качестве второго параметра выбрано `function`.<br> [10] поддерживает несколько записей "Field name,OID prefix,Format records", разделенных символом новой строки.

**item.create**

Описание

`object item.create(object/array items)`

Этот метод позволяет создавать новые элементы данных.

**Note:**
Веб-элементы данных нельзя создавать через Zabbix API.

**Note:**
Этот метод доступен только для типов пользователей *Admin* и *Super admin*. Разрешение на вызов метода можно отозвать в настройках роли пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(object/array) Элементы данных для создания.

Помимо стандартных свойств элемента данных, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| preprocessing | array | Предобработка элемента данных options. |
| tags | array | Теги элемента данных. |

Возвращаемые значения

(object) Возвращает объект, содержащий идентификаторы созданных элементов данных в свойстве itemids. Порядок возвращаемых идентификаторов соответствует порядку переданных элементов данных.

Примеры

Создание элемента данных

Создайте числовой элемент данных Zabbix agent с 2 тегами элемента данных для мониторинга свободного места на диске на узле сети с ID "30074".

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24758"
        ]
    },
    "id": 1
}
```

Создание элемента инвентаря узла сети

Создайте элемент данных агента Zabbix, чтобы заполнить поле инвентаря узла сети "OS".

Request:

```
{
    "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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 1
}
```

Создание элемента данных с предварительной обработкой

Создайте элемент данных с использованием пользовательского множителя.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44210"
        ]
```

```
    },
    "id": 1
}
```

Создание зависимого элемента данных

Создайте зависимый элемент данных для master-элемента с ID 24759. Разрешены только зависимости на одном и том же узле сети, поэтому master-элемент и зависимый элемент данных должны иметь одинаковый hostid.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Создание элемента данных HTTP-агента

Создайте элемент данных с методом запроса POST и предварительной обработкой ответа JSON.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Создание элемента данных типа script

Создайте простую сборку данных с помощью элемента данных типа script.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Источник

CItem::create() в *ui/include/classes/api/services/CItem.php*.

**item.delete**

Описание

`object item.delete(    itemIds)`

Этот метод позволяет удалять элементы данных.

> **Note:**
> Веб элементы данных нельзя удалить через Zabbix API.

> **Note:**
> Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

(    ) ID удаляемых элементов данных.

Возвращаемые значения

(    ) Возвращает объект, который содержит ID удаленных элементов данных, указанных в свойстве `itemids`.

Примеры

Удаление нескольких элементов данных

Удаление двух элементов данных.<br> Зависимые элементы данных и прототипы элементов данных удаляются автоматически, если удаляется основной элемент данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.delete",
    "params": [
        "22982",
        "22986"
    ],
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22982",
            "22986"
        ]
    },
    "id": 1
}
```

Источник

CItem::delete() в *ui/include/classes/api/services/CItem.php*.

**item.get**

Описание

`integer/array item.get(          )`

Этот метод позволяет получать элементы данных в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| itemids | ID/array | Возвращать только элементы данных с указанными ID. |
| groupids | ID/array | Возвращать только элементы данных, которые принадлежат узлам сети из указанных групп. |
| templateids | ID/array | Возвращать только элементы данных, которые принадлежат указанным шаблонам. |
| hostids | ID/array | Возвращать только элементы данных, которые принадлежат указанным узлам сети. |
| proxyids | ID/array | Возвращать только элементы данных, которые контролируются указанными прокси. |
| interfaceids | ID/array | Возвращать только элементы данных, которые используют указанные интерфейсы узлов сети. |
| graphids | ID/array | Возвращать только элементы данных, которые используются в указанных графиках. |
| triggerids | ID/array | Возвращать только элементы данных, которые используются в указанных триггерах. |
| webitems | flag | Включать веб-элементы данных в результат. |
| inherited | boolean | Если установлено значение `true`, возвращать только элементы данных, унаследованные от шаблона. |
| templated | boolean | Если установлено значение `true`, возвращать только элементы данных, которые принадлежат шаблонам. |
| monitored | boolean | Если установлено значение `true`, возвращать только включенные элементы данных, которые принадлежат контролируемым узлам сети. |
| group | string | Возвращать только элементы данных, которые принадлежат группе с указанным именем. |
| host | string | Возвращать только элементы данных, которые принадлежат узлу сети с указанным именем. |
| evaltype | integer | Метод оценки тегов.<br><br>Возможные значения:<br>0 - *(по умолчанию)* And/Or;<br>2 - Or. |
| tags | array | Возвращать только элементы данных с указанными тегами.<br>Формат: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`.<br>Пустой массив возвращает все элементы данных.<br><br>Возможные значения оператора:<br>0 - *(по умолчанию)* Contains;<br>1 - Equals;<br>2 - Does not contain;<br>3 - Does not equal;<br>4 - Exists;<br>5 - Does not exist. |
| with_triggers | boolean | Если установлено значение `true`, возвращать только элементы данных, которые используются в триггерах. |
| selectHosts | query | Возвращать свойство `hosts` с массивом узлов сети, к которым принадлежит элемент данных. |
| selectInterfaces | query | Возвращать свойство `interfaces` с массивом интерфейсов узлов сети, используемых элементом данных. |

| Parameter | Type | Description |
|---|---|---|
| selectTriggers | query | Возвращать свойство `triggers` с триггерами, в которых используется элемент данных.<br><br>Поддерживает `count`. |
| selectGraphs | query | Возвращать свойство `graphs` с графиками, которые содержат элемент данных.<br><br>Поддерживает `count`. |
| selectDiscoveryData | query | Возвращать свойство `discoveryData` с данными объекта обнаружения элемента данных. Объект обнаружения элемента данных связывает обнаруженный элемент данных с прототипом элемента данных, из которого он был обнаружен.<br><br>Он имеет следующие свойства:<br>`parent_itemid` - `(string)` ID прототипа элемента данных, из которого был создан элемент данных;<br>`key_` - `(string)` ключ прототипа элемента данных;<br>`status` - `(int)` статус обнаружения элемента данных:<br>0 - *(по умолчанию)* элемент данных обнаружен,<br>1 - элемент данных больше не обнаружен;<br>`ts_delete` - `(timestamp)` время, когда элемент данных, который больше не обнаружен, будет удален;<br>`ts_disable` - `(timestamp)` время, когда элемент данных, который больше не обнаружен, будет отключен;<br>`disable_source` - `(int)` признак того, был ли элемент данных отключен правилом LLD или вручную:<br>0 - *(по умолчанию)* отключен автоматически,<br>1 - отключен правилом LLD. |
| selectDiscoveryRule | query | Возвращать свойство `discoveryRule` с правилом LLD, которое создало элемент данных. |
| selectPreprocessing | query | Возвращать свойство `preprocessing` с параметрами предварительной обработки элемента данных. |
| selectTags | query | Возвращать теги элемента данных в свойстве `tags`. |
| selectValueMap | query | Возвращать свойство `valuemap` с картой значений элемента данных. |
| filter | object | Возвращать только те результаты, которые точно соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств, а значения — либо одно значение, либо массив значений, с которыми выполняется сравнение.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Поддерживает дополнительные свойства:<br>`host` - техническое имя узла сети, к которому принадлежит элемент данных. |
| limitSelects | integer | Ограничивает количество записей, возвращаемых подвыборами.<br><br>Применяется к следующим подвыборам:<br>`selectGraphs` - результаты будут отсортированы по `name`;<br>`selectTriggers` - результаты будут отсортированы по `description`. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `itemid`, `name`, `key_`, `delay`, `history`, `trends`, `type`, `status`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| excludeSearch | boolean | |
| limit | integer | |

| Parameter | Type | Description |
|---|---|---|
| output | query | |
| preservekeys | boolean | |
| search | object | |
| searchByAny | boolean | |
| searchWildcardsEnabled | boolean | |
| sortorder | string/array | |
| startSearch | boolean | |
| selectItemDiscovery | query | Возвращать свойство `itemDiscovery` с объектом обнаружения элемента данных. Объект обнаружения элемента данных связывает элемент данных с прототипом элемента данных, из которого он был создан.<br><br>Этот запрос **устарел**, вместо него используйте `selectDiscoveryData`. |

Возвращаемые значения

( / ) Возвращает либо:

- ·массив объектов;
- ·количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Поиск элементов данных по ключу

Получить все элементы данных, используемые в триггерах для указанного ID узла сети, у которых в ключе элемента данных есть слово ”system.cpu”, и отсортировать результаты по имени.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10084",
        "with_triggers": true,
        "search": {
            "key_": "system.cpu"
        },
        "sortfield": "name"
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "%",
```

```
        "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",
        "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": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "CPU utilization"
    },
    {
        "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": "",
        "logtimefmt": "",
        "templateid": "42219",
        "valuemapid": "0",
```

```
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
        "description": "",
        "inventory_link": "0",
        "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": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Load average (15m avg)"
    },
    {
        "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": "",
        "logtimefmt": "",
        "templateid": "42209",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
```

```
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
        "description": "",
        "inventory_link": "0",
        "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": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Load average (1m avg)"
    },
    {
        "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": "",
        "logtimefmt": "",
        "templateid": "42217",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
```

```
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
        "description": "",
        "inventory_link": "0",
        "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": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Load average (5m avg)"
    },
    {
        "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": "",
        "logtimefmt": "",
        "templateid": "42220",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "1",
```

```
            "description": "",
            "inventory_link": "0",
            "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": "",
            "state": "0",
            "error": "",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0",
            "name_resolved": "Number of CPUs"
        }
    ],
    "id": 1
}
```


Поиск зависимых элементов данных по ключу

Получите все зависимые элементы данных с узла сети с ID "10116", у которых в ключе есть слово "apache".

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10116",
        "search": {
            "key_": "apache"
        },
        "filter": {
            "type": 18
        }
    },
    "id": 1
}
```

Ответ:


```
{
    "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": "",
        "logtimefmt": "",
        "templateid": "0",
        "valuemapid": "0",
        "params": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "flags": "0",
        "interfaceid": "0",
        "description": "",
        "inventory_link": "0",
        "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": "",
        "state": "0",
        "error": "",
        "parameters": [],
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "name_resolved": "Days"
    },
    {
        "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": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "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": "",
            "state": "0",
            "error": "",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "0",
            "prevvalue": "0",
            "name_resolved": "Hours"
        }
    ],
    "id": 1
}
```


Найти элемент данных HTTP agent

Найти элемент данных HTTP agent с типом тела POST XML для указанного ID узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "hostids": "10255",
        "filter": {
            "type": 19,
            "post_type": 3
        }
    },
    "id": 1
}
```

Ответ:

```
{
    "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": "",
            "logtimefmt": "",
            "templateid": "0",
            "valuemapid": "0",
            "params": "",
            "ipmi_sensor": "",
            "authtype": "0",
            "username": "",
            "password": "",
            "publickey": "",
            "privatekey": "",
            "flags": "0",
            "interfaceid": "0",
            "description": "",
            "inventory_link": "0",
            "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": "",
            "state": "0",
            "error": "",
            "parameters": [],
            "lastclock": "0",
            "lastns": "0",
            "lastvalue": "",
            "prevvalue": "",
            "name_resolved": "template item"
        }
    ],
    "id": 1
}
```

Получение элементов данных с правилами предварительной обработки

Получите все элементы данных и их правила предварительной обработки для указанного ID узла сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": ["itemid", "name", "key_"],
        "selectPreprocessing": "extend",
        "hostids": "10254"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

См. также

- Правило обнаружения
- График
- Узел сети
- Интерфейс узла сети

- Триггер

Источник

CItem::get() в *ui/include/classes/api/services/CItem.php*.

**item.update**

Описание

`object item.update( / items)`

Этот метод позволяет обновлять существующие элементы данных.

**Note:**
Веб элементы данных нельзя обновлять через Zabbix API.

**Note:**
Этот метод доступен только пользователям с типом доступа *Администратор* и *Супер-администратор*. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object/array)` Свойства элемента данных, которые необходимо обновить.

Для каждого элемента данных должно быть определено свойство `itemid`, все остальные свойства необязательны. Будут обновлены только переданные свойства, все остальные останутся без изменений.

В дополнение к стандартным свойствам элемента данных, метод принимает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| preprocessing | array | Параметры предобработки элемента данных для замены текущих параметров предобработки.<br><br>Поведение параметра:<br>- *только для чтения* для унаследованных объектов или обнаруженных объектов |
| tags | array | Теги элемента данных.<br><br>Поведение параметра:<br>- *только для чтения* для обнаруженных объектов |

Возвращаемые значения

`( )` Возвращает объект, который содержит ID обновленных элементов данных, указанных в свойстве `itemids`.

Примеры

Включение элемента данных

Включите элемент данных, то есть установите для его статуса значение ”0”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "10092",
        "status": 0
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 1
}
```

Обновление зависимого элемента данных

Обновите имя зависимого элемента данных и ID мастер-элемента. Разрешены только зависимости на одном узле сети, поэтому мастер-элемент и зависимый элемент данных должны иметь одинаковый hostid.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "name": "Dependent item updated name",
        "master_itemid": "25562",
        "itemid": "189019"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189019"
        ]
    },
    "id": 1
}
```

Обновление элемента данных HTTP agent

Включить перехват значений элемента данных.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "allow_traps": 1
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Обновление элемента данных с предварительной обработкой

Обновите элемент данных с правилом предварительной обработки элемента данных ”In range”.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "preprocessing": [
            {
                "type": 13,
                "params": "\n100",
                "error_handler": 1,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Обновление элемента данных типа скрипт

Обновите элемент данных типа скрипт, указав другой скрипт, и удалите ненужные параметры, которые использовались предыдущим скриптом.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "params": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Источник

CItem::update() в *ui/include/classes/api/services/CItem.php*.

### Элемент данных графика

Этот класс предназначен для работы с элементами данных графика.

Ссылки на объекты:

- Элемент данных графика

Доступные методы:

- graphitem.get - получить элементы данных графика

### Объект элемента данных графика

Следующие объекты напрямую связаны с API `graphitem`.

Элемент графика

**Note:**
Элементы графика можно изменять только через API `graph`.

Объект элемента графика имеет следующие свойства.

| Property | Type | Description |
|---|---|---|
| gitemid | ID | ID элемента графика.<br><br>Поведение свойства:<br>- *только для чтения* |
| color | string | Цвет отрисовки элемента графика в виде шестнадцатеричного кода цвета.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| itemid | ID | ID элемента данных.<br><br>Поведение свойства:<br>- *обязательно* для операций создания |
| calc_fnc | integer | Значение элемента данных, которое будет отображаться.<br><br>Возможные значения:<br>1 - минимальное значение;<br>2 - *(по умолчанию)* среднее значение;<br>4 - максимальное значение;<br>7 - все значения;<br>9 - последнее значение, используется только для круговых и развернутых графиков. |
| drawtype | integer | Стиль отрисовки элемента графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* линия;<br>1 - заполненная область;<br>2 - жирная линия;<br>3 - точка;<br>4 - пунктирная линия;<br>5 - градиентная линия. |
| graphid | ID | ID графика, к которому принадлежит элемент графика. |
| sortorder | integer | Позиция элемента на графике.<br><br>По умолчанию: начинается с ”0” и увеличивается на единицу с каждой записью. |

| Property | Type | Description |
|---|---|---|
| type | integer | Тип элемента графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* простой;<br>2 - сумма графика, используется только для круговых и развернутых графиков. |
| yaxisside | integer | Сторона графика, на которой будет отображаться шкала Y элемента графика.<br><br>Возможные значения:<br>0 - *(по умолчанию)* левая сторона;<br>1 - правая сторона. |

**graphitem.get**

Описание

`integer/array graphitem.get(parameters)`

Этот метод позволяет получать элементы графика в соответствии с заданными параметрами.

**Note:**
Этот метод доступен пользователям любого типа. Разрешения на вызов метода можно отозвать в настройках ролей пользователя. Дополнительную информацию см. в разделе Роли пользователей.

Параметры

`(object)` Параметры, определяющие требуемый результат.

Метод поддерживает следующие параметры.

| Parameter | Type | Description |
|---|---|---|
| graphids | ID/array | Возвращать только элементы графиков, которые принадлежат указанным графикам. |
| itemids | ID/array | Возвращать только элементы графиков с указанными ID элементов данных. |
| type | integer | Возвращать только элементы графиков с указанным типом.<br><br>См. список поддерживаемых типов элементов графиков на странице объекта элемента графика. |
| selectGraphs | query | Возвращать свойство `graphs` с массивом графиков, к которым принадлежит элемент. |
| sortfield | string/array | Сортировать результат по указанным свойствам.<br><br>Возможные значения: `gitemid`. |
| countOutput | boolean | Эти параметры описаны в справочном комментарии. |
| editable | boolean | |
| limit | integer | |
| output | query | |
| preservekeys | boolean | |
| sortorder | string/array | |

Возвращаемые значения

`(object/array)` Возвращает либо:

- массив объектов;
- количество извлеченных объектов, если использовался параметр `countOutput`.

Примеры

Получение элементов графика из графика

Получите все элементы графика, используемые в графике, с дополнительной информацией об элементе данных и узле сети.

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "graphitem.get",
    "params": {
        "output": "extend",
        "graphids": "387"
    },
    "id": 1
}
```

Ответ:

```
{
    "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
}
```

Смотрите также

- График

Источник

CGraphItem::get() в *ui/include/classes/api/services/CGraphItem.php*.

**Обозначение** Типы данных

API Zabbix поддерживает следующие типы данных в качестве входных данных:

| Type | Description |
|---|---|
| ID | Уникальный идентификатор, используемый для ссылки на сущность. |
| boolean | Логическое значение (либо `true`, либо `false`). |
| flag | Значение, которое считается `true`, если оно передано и не равно `null`; в противном случае значение считается `false`. |
| integer | Целое число. |
| float | Число с плавающей точкой. |
| string | Текстовая строка. |
| text | Более длинная текстовая строка. |
| timestamp | Метка времени Unix. |
| array | Упорядоченная последовательность значений (обычный массив). |
| object | Ассоциативный массив. |
| query | Значение, определяющее, какие данные будут возвращены. Значение может быть задано как массив имен свойств (чтобы вернуть только определенные свойства) или как одно из предопределенных значений:<br>`extend` - возвращает все свойства объекта;<br>`count` - возвращает количество полученных записей, поддерживается только некоторыми подзапросами. |

**Attention:**
API Zabbix всегда возвращает значения только в виде строк или массивов.

Поведение свойства

Некоторые свойства объектов помечены короткими метками, описывающими их поведение. Используются следующие обозначения:

- ***только для чтения*** - значение свойства задаётся автоматически и не может быть определено или изменено пользователем, даже в особых случаях (например, *только для чтения* для наследуемых или обнаруженных объектов);
- ***только для записи*** - значение свойства можно установить, но нельзя прочитать после;
- ***постоянное*** -значение свойства можно задать при создании объекта, но нельзя изменить позже;
- ***поддерживаемое*** - значение свойства не обязательно к установке, но может быть задано при определённых условиях (например, *поддерживаемое*, если `type` установлен в ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” или ”HTTP agent”). Однако даже *поддерживаемое* свойства могут автоматически принимать значения по умолчанию независимо от условий;
- ***обязательное*** значение свойства обязательно для указания при всех операциях (кроме get) или при определённых условиях (например, *обязательное* при создании объекта; *обязательное*, если `operationtype` имеет значение ”global script”, а `opcommand_hst` не задан).

**Note:**
Для операций обновления свойство считается ”установленным”, если его значение задаётся в процессе выполнения операции обновления.

Свойства, не отмеченные метками, являются необязательными.

Поведение параметров

Некоторые параметры операции помечены короткими метками, описывающими их поведение для операции. Используются следующие метки:

- ***read-only*** - значение параметра задается автоматически и не может быть определено или изменено пользователем, даже в некоторых конкретных случаях (например, *read-only* для унаследованных объектов или обнаруженных объектов);
- ***write-only*** - значение параметра можно задать, но после этого к нему нельзя получить доступ;

- ***supported*** - значение параметра не требуется задавать, но его можно задать в некоторых конкретных случаях (например, *supported*, если `operating_mode` объекта Proxy установлен в "passive proxy"); однако следует отметить, что параметры *supported* все равно могут быть установлены в значения по умолчанию независимо от условий;
- ***required*** - значение параметра обязательно должно быть задано.

Параметры, не помеченные метками, являются необязательными.

**Зарезервированное значение ID равное "0"** Зарезервированное значение ID "0" можно использовать для фильтрации элементов и удаления связанных объектов. Например, для удаления ссылки на прокси с узла сети, proxytid необходимо задать значением 0 ("proxyid": "0") или для фильтрации узлов сети наблюдаемых сервером, опция proxyids должна быть задана значением 0 ("proxyids": "0").

**Общие параметры метода `get`** Следующие параметры поддерживаются всеми методами `get`:

| Parameter | Type | Description |
|---|---|---|
| countOutput | boolean | Возвращать количество записей в результате вместо самих данных. |
| editable | boolean | Если установлено в `true`, возвращать только объекты, для которых у пользователя есть права на запись.<br><br>По умолчанию: `false`. |
| excludeSearch | boolean | Возвращать результаты, которые не соответствуют критериям, заданным в параметре `search`. |
| filter | object | Возвращать только те результаты, которые точно соответствуют заданному фильтру.<br><br>Принимает объект, где ключи — это имена свойств (например, свойства объекта Host в `host.get`, свойства объекта Item в `item.get` и т. д.), а значения — либо одно значение, либо массив значений для сопоставления.<br><br>Не поддерживает свойства с типом данных `text`.<br><br>Обратите внимание, что некоторые методы имеют специальную функциональность для этого параметра, которая описана на странице метода (например, параметр `filter` в host.get также поддерживает свойства интерфейса узла сети). |
| limit | integer | Ограничить количество возвращаемых записей. |
| output | query | Свойства объекта, которые нужно вернуть.<br><br>Обратите внимание, что идентификатор объекта (то есть `hostid`, `itemid` и т. д.) всегда включается в ответ, даже если он не указан в параметре `output`.<br><br>По умолчанию: `extend`. |

| Parameter | Type | Description |
|---|---|---|
| preservekeys | boolean | Использовать идентификаторы в качестве ключей в результирующем массиве. |
| search | object | Возвращать результаты, которые соответствуют заданному шаблону (без учета регистра).<br><br>Принимает объект, где ключи — это имена свойств (например, свойства объекта Host в `host.get`, свойства объекта Item в `item.get` и т. д.), а значения — строки для поиска. Если дополнительные параметры не заданы, будет выполнен поиск `LIKE "%…%"`.<br><br>Поддерживает только свойства с типом данных `string` и `text`.<br><br>Обратите внимание, что некоторые методы имеют специальную функциональность для этого параметра, которая описана на странице метода (например, параметр `search` в host.get также поддерживает свойства интерфейса узла сети). |
| searchByAny | boolean | Если установлено в `true`, возвращать результаты, которые соответствуют любому из критериев, заданных в параметре `filter` или `search`, а не всем сразу.<br><br>По умолчанию: `false`. |
| searchWildcardsEnabled | boolean | Если установлено в `true`, позволяет использовать `*` в качестве символа подстановки в параметре `search`.<br><br>По умолчанию: `false`. |
| sortfield | string/array | Сортировать результат по заданным свойствам. Список свойств, которые можно использовать для сортировки, см. в описании конкретного метода API `get`. Макросы перед сортировкой не раскрываются.<br><br>Если значение не указано, данные будут возвращены без сортировки. |
| sortorder | string/array | Порядок сортировки. Если передан массив, каждое значение будет сопоставлено соответствующему свойству, указанному в параметре `sortfield`.<br><br>Возможные значения:<br>`ASC` - *(по умолчанию)* по возрастанию;<br>`DESC` - по убыванию. |

| Parameter | Type | Description |
|---|---|---|
| startSearch | boolean | Параметр `search` будет сравнивать начало полей, то есть вместо этого выполнять поиск `LIKE "…%"`.<br><br>Игнорируется, если `searchWildcardsEnabled` установлено в `true`. |

**Флаги происхождения сущности** Методы получения возвращают свойство `flags` для сущностей, связанных с низкоуровневым обнаружением (правило LLD/прототип правила LLD, элемент данных/прототип элемента данных и т. д.). Это свойство полезно для определения того, была ли сущность обнаружена, поскольку редактирование обнаруженных сущностей ограничено.

Свойство `flags` возвращает результат на основе комбинации (”+” операция) следующих значений:

| Value | Description |
|---|---|
| 0 | Базовая сущность (элемент данных, триггер, график, узел сети) |
| 1 | Правило низкоуровневого обнаружения |
| 2 | Любой прототип (прототип элемента данных, прототип триггера, прототип правила LLD и т. д.) |
| 4 | Обнаруженная сущность (обнаруженный элемент данных, триггер, график, узел сети, правило LLD) |

**Комбинированное** значение, возвращаемое свойством `flags`, может быть следующим:

| Value | Combination of | Description |
|---|---|---|
| **0** | 0 | Обычная сущность (элемент данных, триггер, график, узел сети). |
| **2** | 2 | Прототип сущности (прототип элемента данных, прототип триггера и т. д.). |
| **6** | 2+4 | Обнаруженный элемент данных, триггер, график, узел сети (преобразованный из прототипа). |
| **1** | 1 | Правило низкоуровневого обнаружения. |
| **3** | 1+2 | Прототип правила низкоуровневого обнаружения. |
| **5** | 1+4 | Обнаруженное правило низкоуровневого обнаружения (преобразованное из прототипа). |
| **7** | 1+2+4 | Обнаруженный прототип правила низкоуровневого обнаружения. |

**Примеры** Проверка прав пользователя

Имеет ли пользователь разрешение на запись на хосты, имена которых начинаются с ”MySQL” или ”Linux”?

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
        "search": {
            "host": ["MySQL", "Linux"]
        },
        "editable": true,
        "startSearch": true,
        "searchByAny": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": "0",
    "id": 1
}
```

**Note:**
Нулевой результат означает отсутствие хостов с разрешениями на чтение/запись.

Подсчет несоответствий

Подсчитать количество хостов, имена которых не содержат подстроку ”ubuntu”

Запрос:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
        "search": {
            "host": "ubuntu"
        },
        "excludeSearch": true
    },
    "id": 1
}
```

Ответ:

```
{
    "jsonrpc": "2.0",
    "result": "44",
    "id": 1
}
```

Поиск хостов по подстановочным знакам

Найти хосты, имя которых содержит слово «server» и имеют интерфейсные порты «10050» или «10071». Отсортировать результат по имени хоста в порядке убывания и ограничьте его до 5 хостов.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "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
}
```


Поиск хостов с использованием подстановочных знаков с «preservekeys»

Если к предыдущему запросу добавить параметр «preservekeys», то результат возвращается в виде ассоциативного массива, где ключами являются id объектов.

Запрос:

```
{
    "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
}
```

Ответ:

```
{
    "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
}
```

**Приложение 2. Изменения с 7.2 до 7.4**

**Обратно несовместимые изменения** map

ZBXNEXT-8784 Из объекта триггера ссылки карты удалены свойства `linktriggerid` и `linkid`.

медиатип

ZBX-26258 `mediatype.get`: Добавлена строгая проверка параметра `selectUsers`.

user

ZBX-26258 `user.get`: Добавлена строгая проверка параметров метода.<br> ZBX-26258 `user.get`: Свойство `userid` теперь возвращается только если оно явно запрошено или если `output` установлено в "extend".

usergroup

ZBX-26258 `usergroup.get`: Добавлена строгая проверка параметров `mfaids`, `selectUsers`.

**Другие изменения и исправления ошибок** панель мониторинга

ZBXNEXT-8919 Добавлено новое поле виджета панели мониторинга `ds.0.color_palette` для виджетов `svggraph` и `piechart`.<br> ZBXNEXT-9087 Добавлен новый тип виджета панели мониторинга `itemcard`.

прототип правила обнаружения

ZBXNEXT-1527 Добавлен новый API прототипа правила низкоуровневого обнаружения с методами `discoveryruleprototype.creat` `discoveryruleprototype.update`, `discoveryruleprototype.get`, `discoveryruleprototype.delete`.

graph

ZBXNEXT-1527 Добавлен новый параметр `selectDiscoveryData` в метод `graph.get`; параметр `selectGraphDiscovery` объявлен устаревшим.

graphprototype

ZBXNEXT-1527 Добавлены новые параметры `selectDiscoveryRulePrototype` и `selectDiscoveryData` в метод `graphprototype.get`.

узел сети

ZBXNEXT-1527 Добавлены новые параметры `selectDiscoveryData` и `selectDiscoveryRules` в метод `host.get`; параметры `selectHostDiscovery` и `selectDiscoveries` объявлены устаревшими.

группа узлов сети

ZBXNEXT-1527 Добавлен новый параметр `selectDiscoveryData` в метод `hostgroup.get`; параметр `selectGroupDiscoveries` объявлен устаревшим.

прототип узла сети

ZBXNEXT-1527 Добавлены новые параметры `selectDiscoveryRulePrototype` и `selectDiscoveryData` в метод `hostprototype.get`.

элемент данных

ZBXNEXT-1527 Добавлен новый параметр `selectDiscoveryData` в метод `item.get`; параметр `selectItemDiscovery` объявлен устаревшим.

itemprototype

ZBXNEXT-1527 Добавлены новые параметры `selectDiscoveryRulePrototype` и `selectDiscoveryData` в метод `itemprototype.get`.

map

ZBXNEXT-875 Добавлено свойство `zindex` к объекту элемент карты.<br> ZBXNEXT-2122 Добавлены свойства `edit_own_media` и `edit_user_media` к объекту action API ролей.<br> ZBXNEXT-8784 Добавлены свойства `background_scale`, `show_element_label`, `show_link_label` к объекту map.<br> ZBXNEXT-8784 Добавлено свойство `show_label` к объекту элемент карты.<br> ZBXNEXT-8784 Добавлены свойства `show_label`, `indicator_type`, `itemid`, `thresholds`, `highlights` к объекту map link.<br> ZBXNEXT-8784 Добавлен объект map link indicators.

mediatype

ZBXNEXT-9805 Добавлена опция 2 (”OAuth token”) в свойство `smtp_authentication` объекта mediatype.<br> ZBXNEXT-9805 Добавлены свойства `redirection_url`, `client_id`, `client_secret`, `authorization_url`, `token_url`, `tokens_status`, `access_token`, `access_token_updated`, `access_expires_in`, `refresh_token` в объект mediatype.

шаблон

ZBXNEXT-9872 Добавлены новые свойства `readme` и `wizard_ready`.<br> ZBXNEXT-1527 Добавлен новый параметр `selectDiscoveryRules` в метод `template.get`; параметр `selectDiscoveries` объявлен устаревшим.

шаблонная панель мониторинга

ZBXNEXT-9087 Добавлен новый тип виджета панели мониторинга шаблона `itemcard`.

триггер

ZBXNEXT-1527 Добавлен новый параметр `selectDiscoveryData` в метод `trigger.get`; параметр `selectTriggerDiscovery` объявлен устаревшим.

triggerprototype

ZBXNEXT-1527 Добавлены новые параметры `selectDiscoveryRulePrototype` и `selectDiscoveryData` в метод `triggerprototype.get`.

usermacro

ZBXNEXT-9872 Добавлено новое свойство `config` в объект макроса узла сети.

**Приложение 3. Изменения в 7.4**

**7.4.11** discoveryrule

ZBX-27780 Параметры `delay`, `history` и `trends` в объектах операции переопределения правила LLD снова поддерживают макросы LLD.

**7.4.3** панель мониторинга

ZBXNEXT-10143 Добавлено новое поле виджета панели мониторинга `highlight_row` в виджете problems.

задача

ZBX-26202 `task.create`: пользователи типов *Admin* и *User* теперь могут создавать задачи ”Выполнить сейчас”.

## 21. Расширения

**Обзор** Хотя Zabbix предлагает множество возможностей, всегда есть место для дополнительных функций. Расширения — это удобный способ изменения и увеличения возможностей мониторинга при помощи Zabbix без изменения исходного кода.

Вы можете расширить функциональность Zabbix как используя встроенные возможности расширения (траппер элементы данных, пользовательские параметры и т. д.), так и путём использования или создания пользовательских расширений (загружаемых модулей, плагинов и т. д.).

В этом разделе представлен обзор со ссылками на все варианты расширения Zabbix.

**Сбор данных с помощью пользовательских команд** Траппер элементы данных

Траппер элементы данных — это элементы данных, которые принимают входящие данные, а не запрашивают их. Траппер элементы данных полезны для отправки определённых данных на Zabbix сервер или прокси, например, периодические данные о доступности и производительности в случае долгоиграющих пользовательских скриптов. Отправка данных на Zabbix сервер или прокси возможна с помощью утилиты командной строки Zabbix sender или путем реализации коммуникационного протокола на основе JSON (аналогично тому, что используется в Zabbix sender). Отправка данных к серверу Zabbix также возможна при помощи метода API `history.push`.

Внешние проверки

Внешняя проверка — это элемент данных для выполнения проверок путём запуска исполняемого файла, например сценария оболочки или двоичного файла.

Внешние проверки выполняются Zabbix сервером или прокси (когда узел сети наблюдается через прокси) и не требуют наличия агента, работающего на отслеживаемом узле сети.

Пользовательские параметры

Пользовательский параметр — это заданная пользователем команда (связанная с задаваемым пользователем ключом), которая при выполнении может получить необходимые вам данные с узла сети, на котором работает Zabbix агент. Пользовательские параметры полезны для настройки элементов данных агента или агента 2, отсутствующих в Zabbix.

Элементы данных Zabbix агента `system.run[]`

Элемент данных Zabbix агента `system.run[]` — это элемент данных для определяемой пользователем команды (связанный с предопределённым ключом `system.run[]`, например `system.run[myscript.sh]`), который может быть выполнен на узле сети, где запущен агент Zabbix.

Примечание: элементы данных `system.run[]` отключены по умолчанию и, если они используются, должны быть включены (разрешены) и определены в конфигурационном файле Zabbix агента или Zabbix агента 2 (параметр конфигурации `AllowKey`).

> **Attention:**
> Пользовательские команды в элементах данных, таких как внешние проверки, пользовательские параметры и элементы данных Zabbix агента `system.run[]`, выполняются от того пользователя ОС, который используется для запуска компонентов Zabbix. Для выполнения этих команд данный пользователь должен обладать необходимыми правами.

Элементы данных HTTP агента

Элемент данных HTTP агента — это элемент данных для выполнения запросов данных через HTTP/HTTPS. Элементы данных HTTP агента полезны для отправки запросов к конечным точкам HTTP для получения данных от таких сервисов, как *Elasticsearch* и *OpenWeatherMap*, для проверки состояния Zabbix API или состояния веб-сервера Apache либо Nginx и т. д. Элементы данных HTTP (с включённой опцией разрешения получения трапов) так же могут функционировать как траппер элементы данных.

Скриптовые элементы данных

Скриптовый элемент данных — это элемент данных для выполнения пользовательского кода JavaScript, который получает данные через HTTP/HTTPS. Скриптовые элементы данных полезны, когда функциональности, предоставляемой элементами HTTP агента, недостаточно. Например, в сложных сценариях сбора данных, требующих нескольких шагов или сложной логики, скриптовый элемент данных можно настроить для выполнения HTTP-вызова, а затем обработки полученных данных и последующей передачи преобразованного значения второму HTTP-вызову.

> **Note:**
> Элементы данных HTTP агента и скриптовые элементы данных поддерживаются Zabbix сервером и прокси сервером, и не требуют наличия агента, работающего на отслеживаемом узле сети.

**Продвинутые расширения** Подгружаемые модули

Подгружаемые модули, написанные на языке C, представляют собой универсальный и ориентированный на производительность вариант расширения функциональности компонентов Zabbix (сервер, прокси, агент) на платформах UNIX. Подгружаемый модуль — это, по сути, общая библиотека, используемая демоном Zabbix и подгружаемая при запуске. Библиотека должна содержать определённые функции, чтобы процесс Zabbix мог обнаружить, что файл действительно является модулем, который он может загрузить и с которым он может работать.

Подгружаемые модули имеют набор преимуществ, включая возможность добавлять новые метрики или реализовывать любую другую логику (для примера, экспорта данных истории Zabbix), прекрасную производительность и возможность

разрабатывать, использовать и делиться функциональными возможностями, которые они предоставляют. Это способствует бесперебойному обслуживанию и помогает предоставлять новые функциональные возможности проще и независимо от основной кодой базы Zabbix.

Подгружаемые модули особенно полезны при сложной настройке мониторинга. При мониторинге встроенных систем наличие большого количества отслеживаемых параметров или тяжелые сценарии со сложной логикой или длительным временем запуска такие расширения, как пользовательские параметры, элементы Zabbix агента `system.run[]` и внешние проверки, будут влиять на производительность. Загружаемые модули предлагают способ расширения функциональности Zabbix без ущерба для производительности.

Плагины

Плагины расширяют возможности мониторинга Zabbix агента 2. Будучи написанными на языке программирования Go, они предоставляют альтернативу подгружаемым модулям (написанным на языке C).

Плагин — это пакет Go, который определяет структуру и реализует один или несколько интерфейсов плагина (*Exporter*, *Collector*, *Configurator*, *Runner*, *Watcher*). Поддерживаются два типа плагинов Zabbix агента 2:

- Встроенные плагины [en] (поддерживаются, начиная с Zabbix 4.4.0)
- Подгружаемые плагины [en] (поддерживаются, начиная с Zabbix 6.0.0)

Смотрите список плагинов, предоставляемых компанией Zabbix.

Инструкции и учебные пособия по созданию собственных плагинов смотрите в Центре разработчиков.

**Настройка оповещений** Вебхуки

Вебхук — это тип медиа Zabbix, который предоставляет возможность расширить функции оповещения Zabbix за счет внешнего программного обеспечения, такого как системы поддержки пользователей, чаты или мессенджеры. Подобно элементам данных script, вебхуки полезны для выполнения HTTP-вызовов с использованием пользовательского JavaScript-кода, например, для отправки уведомлений на различные платформы, такие как Microsoft Teams, Discord и Jira. Также можно возвращать некоторые данные (например, о созданных заявках в системе поддержки), которые затем отображаются в Zabbix.

Существующие вебхуки доступны в репозитории Git Zabbix. Для разработки пользовательских вебхуков см. Руководство по разработке вебхуков.

Скрипты оповещений

Скрипт оповещения — это способ оповещения Zabbix, который предоставляет возможность создать альтернативный способ (скрипт) для управления оповещениями Zabbix. Скрипты оповещений полезны, если вас не устраивают существующие способы оповещения для отсылки уведомлений в Zabbix.

**Кастомизация веб-интерфейса** Пользовательские темы

Можно изменить внешний вид веб-интерфейса Zabbix, используя пользовательские темы. Смотрите инструкции по созданию и применению своих собственных тем.

Модули веб-интерфейса

Модули веб-интерфейса предоставляют возможность расширить функциональность веб-интерфейса Zabbix путем добавления сторонних модулей или разработки своих собственных. С помощью модулей веб-интерфейса вы можете добавлять новые пункты меню, соответствующие им представления, действия и т.д.

**Глобальные скрипты** Глобальный скрипт — это определяемый пользователем набор команд, которые могут быть выполнены на объекте мониторинга (с помощью интерпретатора оболочки (/bin/sh)), в зависимости от настроенной области и прав пользователя. Глобальные скрипты можно настроить для следующих действий:

- Операция действия
- Действие вручную над узлом сети
- Действие вручную над событиями

Глобальные скрипты полезны во многих случаях. Например, если настроены операции действий или действия вручную над узлом сети, вы можете использовать глобальные скрипты для автоматического или ручного выполнения удалённых команд, например, перезапуск приложения (веб-сервера, промежуточного программного обеспечения, CRM и т. д.) или освобождение места на диске (удаление старых файлов, очистка `/tmp` и т. д.). Или другой пример: если настроено действие вручную над событиями, вы можете использовать глобальные скрипты для управления заявками о проблемах во внешних системах.

Глобальные скрипты могут выполняться Zabbix сервером, прокси или агентом.

**Attention:**
Пользовательские команды выполняются от пользователя ОС, который используется для запуска компонентов Zabbix. Для выполнения этих команд данный пользователь должен обладать необходимыми правами.

**Zabbix API** Zabbix API — это API на основе HTTP, который является частью веб-интерфейса Zabbix. С помощью Zabbix API вы можете выполнять любые из следующих операций:

- Программное получение и изменение конфигурации Zabbix.
- Импорт и экспорт конфигурации Zabbix.
- Доступ к данным истории и динамики изменений Zabbix.
- Настройка приложений для работы с Zabbix.
- Интеграция Zabbix со сторонним программным обеспечением.
- Автоматизация рутинных задач.

Zabbix API состоит из множества методов, которые номинально сгруппированы в отдельные API. Каждый метод выполняет определённую задачу. Доступные методы, а также обзор функций, предоставляемых Zabbix API, смотрите в Справочнике методов Zabbix API.

### 1. Подгружаемые модули

Обзор

Подгружаемые модули предлагают ориентированный на производительность вариант расширения Zabbix.

Функциональность Zabbix можно расширить многими способами, например, с помощью пользовательских параметров, внешних проверок, а также элементов данных Zabbix агента system.run. Они работают очень хорошо, но имеют один большой недостаток, называемый fork(). Zabbix для сбора пользовательских проверок каждый раз должен создавать новый порождённый процесс, что не очень хорошо сказывается на производительности. Обычно это не самая большая проблема, но тем не менее это может быть серьезной проблемой в случае мониторинга встроенных систем, имеющих большое количество наблюдаемых параметров или тяжёлых скриптов со сложной логикой или длительным временем запуска.

Поддержка подгружаемых модулей предлагает пути расширения Zabbix агента, сервера и прокси без ущерба для производительности.

Подгружаемый модуль — в своей основе разделяемая библиотека, которая используется Zabbix демоном и подгружается при старте демона. Библиотека должна содержать определённые функции, чтобы Zabbix процесс мог определить, что файл на самом деле является модулем, который можно загрузить и с ним работать.

Подгружаемые модули имеют много преимуществ. Отличная производительность и возможность реализовать любую логику очень важны, но, возможно, наиболее важное преимущество — возможность разработать, использовать и поделиться Zabbix модулем. Это способствует беспроблемному обслуживанию и помогает вносить новые функции легче и независимо от основной кодовой базы Zabbix.

Лицензирование и распространение модулей в бинарной форме регламентируется лицензией AGPL-3.0 (модули линкуются с Zabbix во время выполнения и используют заголовки Zabbix; в настоящее время весь код Zabbix, начиная с версии 7.0, лицензируется под лицензией AGPL-3.0). Бинарная совместимость не гарантируется Zabbix'ом.

Постоянство API модулей гарантируется в пределах одного цикла выпуска Zabbix LTS (Долгосрочная поддержка). Постоянство Zabbix API не гарантируется (технически имеется возможность вызова внутренних функций Zabbix из модуля, но нет гарантии, что такие модули будут работать).

API модулей

Для того чтобы разделяемая библиотека обрабатывалась как Zabbix модуль, она должна реализовывать и экспортировать несколько функций. На данный момент имеется шесть функций в API модулей Zabbix, только одна из которых обязательна, а остальные пять — не обязательны.

Обязательный интерфейс

Единственная обязательная функция — **zbx_module_api_version()**:

```
int zbx_module_api_version(void);
```

Эта функция должна возвращать версию API, реализованную в модуле, и, чтобы модуль загрузился, версия должна совпадать с версией модулей API, поддерживаемой Zabbix. Версией модулей API, поддерживаемой Zabbix, является ZBX_MODULE_API_VERSION. Таким образом, эта функция должна возвращать эту константу. Прежняя константа

ZBX_MODULE_API_VERSION_ONE, которая ранее использовалась для этих целей, теперь определена равной ZBX_MODULE_API_VERSION для сохранения совместимости исходного кода, но её использование не рекомендуется.

Опциональный интерфейс

Опциональными функциями являются следующие функции — **zbx_module_init()**, **zbx_module_item_list()**, **zbx_module_item_timeou** **zbx_module_history_write_cbs()** и **zbx_module_uninit()**:

```
int zbx_module_init(void);
```

Эта функция должна выполнять необходимую инициализацию для модуля (если таковая имеется). В случае успеха функция должна вернуть ZBX_MODULE_OK. В противном случае она должна вернуть ZBX_MODULE_FAIL. В последнем случае Zabbix не запустится.

```
ZBX_METRIC  *zbx_module_item_list(void);
```

Эта функция должна возвращать список элементов данных, поддерживаемых модулем. Каждый элемент данных указывается в структуре ZBX_METRIC, смотрите раздел ниже для подробностей. Список завершается структурой ZBX_METRIC с полем «key», равным NULL.

```
void    zbx_module_item_timeout(int timeout);
```

Если модуль экспортирует **zbx_module_item_list()**, тогда эта функция используется Zabbix, чтобы задать опцию времени ожидания в файле конфигурации Zabbix, которой проверки элементов данных, реализованных в модуле, должны подчиняться. Здесь параметр «timeout» (время ожидания) задаётся в секундах.

```
ZBX_HISTORY_WRITE_CBS   zbx_module_history_write_cbs(void);
```

Эта функция должна возвращать функции обратного вызова (callback), которые будут использоваться Zabbix сервером для экспорта истории различных типов данных. Функции обратного вызова представляют собой поля структуры ZBX_HISTORY_WRITE_CBS, поля могут быть NULL, если модуль не заинтересован в истории некоторого типа.

```
int zbx_module_uninit(void);
```

Эта функция должна выполнять необходимые деинициализации (если таковые имеются), такие как освобождение выделенных ресурсов, закрытие файловых дескрипторов и так далее.

Все функции вызываются один раз при запуске Zabbix, когда модуль загружается, за исключением zbx_module_uninit(), которая вызывается один раз при завершении работы Zabbix, когда модуль выгружается.

Определение элементов данных

Каждый элемент данных определяется в структуре ZBX_METRIC:

```
typedef struct
{
    char        *key;
    unsigned    flags;
    int     (*function)();
    char        *test_param;
}
ZBX_METRIC;
```

Здесь, **key** — ключ элемента данных (например, ”dummy.random”), **flags** — либо CF_HAVEPARAMS, либо 0 (в зависимости от того, принимает ли элемент данных параметры или нет), **function** — С функция, которая обрабатывает элемент данных (например, «zbx_module_dummy_random»), и **test_param** — список параметров, который используется, когда Zabbix агент запускается с флагом «-p» (например, ”1,1000”, может быть NULL). Пример определения может выглядеть наподобие этого:

```
static ZBX_METRIC keys[] =
{
    { "dummy.random", CF_HAVEPARAMS, zbx_module_dummy_random, "1,1000" },
    { NULL }
}
```

Каждая функция, которая обрабатывает элемент данных, должна принимать в качестве параметров два указателя, первый с типом AGENT_REQUEST и второй с типом 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;
}
```

Эти функции должны возвращать SYSINFO_RET_OK, если значение элемента данных получено успешно. В противном случае функции должны возвращать SYSINFO_RET_FAIL. Смотрите пример модуля «dummy» ниже для получения деталей, как получать информацию из AGENT_REQUEST и как указывать информацию в AGENT_RESULT.

Предоставление обратного вызова для экспорта истории

**Attention:**
Экспорт истории через модули более не поддерживается Zabbix прокси.

Модуль может задавать функции для экспорта данных истории по типам: Числовой (с плавающей точкой), Числовой (целое положительное), Символ, Текст и Журнал (лог):

```
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;
```

Каждый из них должен принимать массив «history» размером «history_num» элементов в виде аргументов. В зависимости от экспортируемого типа данных истории, «history» является массивом из следующих структур, соответственно:

```
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;
```

Обратные вызовы будут использоваться процессами синхронизации истории Zabbix сервера в конце процедуры синхронизации истории после того, как данные записаны в базу данных Zabbix и сохранены в кэш значений.

**Attention:**
В случае внутренней ошибки в модуле экспорта истории рекомендуется, чтобы модуль был написан таким образом, чтобы он не блокировал весь процесс мониторинга до восстановления, а вместо этого отбрасывал данные и позволял Zabbix серверу продолжать работу.

Сборка модулей

В настоящее время подразумевается, что модули должны быть собраны внутри дерева исходных кодов Zabbix, так как API модулей зависит от некоторых структур данных, которые определены в заголовках Zabbix.

Наиболее важный заголовок для подгружаемых модулей — **include/module.h**, который определяет эти структуры данных. Другие необходимые системные заголовки, которые помогают **include/module.h** работать должным образом, — это **stdlib.h** и **stdint.h**.

С учётом этой информации, все готово для сборки модуля. Модуль должен включать **stdlib.h**, **stdint.h** и **module.h**, а скрипт сборки должен понимать, что эти файлы указаны в путях include. Смотрите пример модуля «dummy» ниже для получения деталей.

Другим полезный заголовком является **include/zbxcommon.h**, который определяет функцию **zabbix_log()**, её можно использовать для журналирования и отладки.

Параметры конфигурации

Zabbix агент, сервер и прокси поддерживают два параметра для работы с модулями:

- LoadModulePath — полный путь до расположения подгружаемых модулей
- LoadModule — модуль(и) для загрузки при старте. Модули должны находиться в папке, указанной в параметре LoadModulePath, или же путь должен быть указан перед именем модуля. Если предшествующий путь абсолютный (начинается с «/»), то LoadModulePath игнорируется. Допускается добавлять несколько параметров LoadModule.

Например, для расширения возможностей Zabbix агента мы можем добавить следующие параметры:

```
LoadModulePath=/usr/local/lib/zabbix/agent/
LoadModule=mariadb.so
LoadModule=apache.so
LoadModule=kernel.so
LoadModule=/usr/local/lib/zabbix/dummy.so
```

При запуске агента модули mariadb.so, apache.so и kernel.so будут загружены из папки /usr/local/lib/zabbix/agent, тогда как dummy.so будет загружен из /usr/local/lib/zabbix. Агент не запустится, если модуль отсутствует, в случае некорректных прав доступа, или если разделяемая библиотека не является модулем Zabbix.

Настройка веб-интерфейса

Подгружаемые модули поддерживаются Zabbix агентом, сервером и прокси. Следовательно, тип элемента данных в веб-интерфейсе Zabbix зависит от того, где этот модуль загружен. Если модуль загружен на стороне агента, тогда тип элемента данных должен быть «Zabbix агент» или «Zabbix агент (активный)». Если модуль загружен на стороне сервера или прокси, тогда тип элемента данных должен быть «Простая проверка».

Экспорт истории через модули Zabbix не требует какой-либо настройки через веб-интерфейс. Если модуль успешно загружен сервером и предоставляет функцию **zbx_module_history_write_cbs()**, которая возвращает по крайней мере

одну не-NULL функцию обратного вызова, то экспорт истории будет включён автоматически.

Пустой модуль

Zabbix включает простой модуль, написанный на языке C. Модуль находится в 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
```

Модуль хорошо документирован, его можно использовать как шаблон для ваших собственных модулей.

После выполнения ./configure в корне дерева исходных кодов Zabbix, как описано ранее, просто выполните **make** для сборки **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"

/* the variable keeps timeout setting for item processing */
static int  item_timeout = 0;

/* module SHOULD define internal functions as static and use a naming pattern different from Zabbix intern
/* symbols (zbx_*) and loadable module API functions (zbx_module_*) to avoid conflicts
static int  dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result);
static int  dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result);
static int  dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result);

static ZBX_METRIC keys[] =
/*  KEY         FLAG        FUNCTION    TEST PARAMETERS */
{
    {"dummy.ping",      0,      dummy_ping, NULL},
    {"dummy.echo",      CF_HAVEPARAMS,  dummy_echo, "a message"},
    {"dummy.random",    CF_HAVEPARAMS,  dummy_random,   "1,1000"},
    {NULL}
};

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_api_version                                           *
 *                                                                            *
 * Purpose: returns version number of the module interface                    *
 *                                                                            *
```

```
 * Return value: ZBX_MODULE_API_VERSION - version of module.h module is       *
 *               compiled with, in order to load module successfully Zabbix   *
 *               MUST be compiled with the same version of this header file   *
 *                                                                            *
 ******************************************************************************/
int zbx_module_api_version(void)
{
    return ZBX_MODULE_API_VERSION;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_item_timeout                                          *
 *                                                                            *
 * Purpose: set timeout value for processing of items                         *
 *                                                                            *
 * Parameters: timeout - timeout in seconds, 0 - no timeout set               *
 *                                                                            *
 ******************************************************************************/
void    zbx_module_item_timeout(int timeout)
{
    item_timeout = timeout;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_item_list                                             *
 *                                                                            *
 * Purpose: returns list of item keys supported by the module                 *
 *                                                                            *
 * Return value: list of item keys                                            *
 *                                                                            *
 ******************************************************************************/
ZBX_METRIC  *zbx_module_item_list(void)
{
    return keys;
}

static int  dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    SET_UI64_RESULT(result, 1);

    return SYSINFO_RET_OK;
}

static int  dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char    *param;

    if (1 != request->nparam)
    {
        /* set optional error message */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
        return SYSINFO_RET_FAIL;
    }

    param = get_rparam(request, 0);

    SET_STR_RESULT(result, strdup(param));

    return SYSINFO_RET_OK;
}
```

```c
/******************************************************************************
 *                                                                            *
 * Function: dummy_random                                                     *
 *                                                                            *
 * Purpose: a main entry point for processing of an item                      *
 *                                                                            *
 * Parameters: request - structure that contains item key and parameters      *
 *              request->key - item key without parameters                    *
 *              request->nparam - number of parameters                        *
 *              request->params[N-1] - pointers to item key parameters        *
 *              request->types[N-1] - item key parameters types:              *
 *                  REQUEST_PARAMETER_TYPE_UNDEFINED (key parameter is empty) *
 *                  REQUEST_PARAMETER_TYPE_ARRAY (array)                      *
 *                  REQUEST_PARAMETER_TYPE_STRING (quoted or unquoted string) *
 *                                                                            *
 *             result - structure that will contain result                    *
 *                                                                            *
 * Return value: SYSINFO_RET_FAIL - function failed, item will be marked      *
 *                                  as not supported by zabbix                *
 *               SYSINFO_RET_OK - success                                     *
 *                                                                            *
 * Comment: get_rparam(request, N-1) can be used to get a pointer to the Nth  *
 *          parameter starting from 0 (first parameter). Make sure it exists  *
 *          by checking value of request->nparam.                            *
 *          In the same manner get_rparam_type(request, N-1) can be used to   *
 *          get a parameter type.                                             *
 *                                                                            *
 ******************************************************************************/
static int  dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char    *param1, *param2;
    int from, to;

    if (2 != request->nparam)
    {
        /* set optional error message */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
        return SYSINFO_RET_FAIL;
    }

    param1 = get_rparam(request, 0);
    param2 = get_rparam(request, 1);

    /* there is no strict validation of parameters and types for simplicity sake */
    from = atoi(param1);
    to = atoi(param2);

    if (from > to)
    {
        SET_MSG_RESULT(result, strdup("Invalid range specified."));
        return SYSINFO_RET_FAIL;
    }

    SET_UI64_RESULT(result, from + rand() % (to - from + 1));

    return SYSINFO_RET_OK;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_init                                                  *
 *                                                                            *
```

```
 * Purpose: the function is called on agent startup                           *
 *          It should be used to call any initialization routines             *
 *                                                                            *
 * Return value: ZBX_MODULE_OK - success                                      *
 *               ZBX_MODULE_FAIL - module initialization failed               *
 *                                                                            *
 * Comment: the module won't be loaded in case of ZBX_MODULE_FAIL             *
 *                                                                            *
 ******************************************************************************/
int zbx_module_init(void)
{
    /* initialization for dummy.random */
    srand(time(NULL));

    return ZBX_MODULE_OK;
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_uninit                                                *
 *                                                                            *
 * Purpose: the function is called on agent shutdown                          *
 *          It should be used to cleanup used resources if there are any      *
 *                                                                            *
 * Return value: ZBX_MODULE_OK - success                                      *
 *               ZBX_MODULE_FAIL - function failed                            *
 *                                                                            *
 ******************************************************************************/
int zbx_module_uninit(void)
{
    return ZBX_MODULE_OK;
}

/******************************************************************************
 *                                                                            *
 * Functions: dummy_history_float_cb                                          *
 *            dummy_history_integer_cb                                        *
 *            dummy_history_string_cb                                         *
 *            dummy_history_text_cb                                           *
 *            dummy_history_log_cb                                            *
 *                                                                            *
 * Purpose: callback functions for storing historical data of types float,    *
 *          integer, string, text and log respectively in external storage    *
 *                                                                            *
 * Parameters: history     - array of historical data                         *
 *             history_num - number of elements in history array              *
 *                                                                            *
 ******************************************************************************/
static void dummy_history_float_cb(const ZBX_HISTORY_FLOAT *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_integer_cb(const ZBX_HISTORY_INTEGER *history, int history_num)
{
    int i;
```

```
    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_string_cb(const ZBX_HISTORY_STRING *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_text_cb(const ZBX_HISTORY_TEXT *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_log_cb(const ZBX_HISTORY_LOG *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

/******************************************************************************
 *                                                                            *
 * Function: zbx_module_history_write_cbs                                     *
 *                                                                            *
 * Purpose: returns a set of module functions Zabbix will call to export      *
 *          different types of historical data                                *
 *                                                                            *
 * Return value: structure with callback function pointers (can be NULL if    *
 *               module is not interested in data of certain types)           *
 *                                                                            *
 ******************************************************************************/
ZBX_HISTORY_WRITE_CBS   zbx_module_history_write_cbs(void)
{
    static ZBX_HISTORY_WRITE_CBS    dummy_callbacks =
    {
        dummy_history_float_cb,
        dummy_history_integer_cb,
        dummy_history_string_cb,
        dummy_history_text_cb,
        dummy_history_log_cb,
    };

    return dummy_callbacks;
}
```

Модуль экпортирует 3 новых элемента данных:

- `dummy.ping` — всегда возвращает «1»
- `dummy.echo[param1]` — возвращает первый параметр как есть, например, `dummy.echo[ABC]` вернет ABC
- `dummy.random[param1, param2]` — возвращает случайное число из диапазона param1-param2, например, `dummy.random[1,1000000]`

Ограничения

Поддержка подгружаемых модулей реализована только на платформах Unix. Это означает, что подгружаемый модуль не работает на Zabbix агентах под Windows.

В некоторых случаях модулю может потребоваться прочитать параметры, имеющие отношение к модулю, из *zabbix_agentd.conf*. В настоящее время этот функционал не поддерживается. Если вам необходимо в вашем модуле использовать некоторые параметры конфигурации, то вам, вероятно, следует реализовать анализ отдельного файла конфигурации модуля.

**2. Плагины**

Обзор

Плагины расширяют возможности мониторинга агента 2 Zabbix. Написанные на языке программирования Go, они представляют собой альтернативу загружаемым модулям (написанным на C) и другим методам расширения функциональности Zabbix.

Эти возможности специфичны для агента 2 и его плагинов:

- Планируемые и гибкие интервалы как для пассивных, так и для активных проверок
- Управление очередью задач с учетом расписания и параллелизма задач
- Тайм-ауты на уровне плагина
- Проверка совместимости агента 2 Zabbix и его плагинов при запуске

Все метрики, поддерживаемые агентом 2 Zabbix, собираются плагинами. На этой странице описаны принципы настройки плагинов с точки зрения пользователя, а также перечислены встроенные и загружаемые плагины, предоставляемые Zabbix.

> **Note:**
> Начиная с Zabbix 6.0, плагины больше не нужно встраивать в агент 2. Их можно добавлять как загружаемые плагины, что упрощает создание плагинов для новых метрик. Инструкции по созданию собственных плагинов см. в Центре разработчика и Примере плагина для агента 2 Zabbix.

Настройка плагинов

В этом разделе представлены общие принципы и рекомендации по настройке плагинов.

Все плагины настраиваются с помощью параметра *Plugins.**, который может быть частью либо файла конфигурации Zabbix агента 2, либо собственного файла конфигурации плагина. Если плагин использует отдельный файл конфигурации, путь к этому файлу необходимо указать в параметре Include файла конфигурации Zabbix агента 2.

Типичный параметр плагина имеет следующую структуру:

*Plugins.<ИмяПлагина>.<Параметр>=<Значение>*

Кроме того, существуют две специальные группы параметров:

- *Plugins.<ИмяПлагина>.Default.<Параметр>=<Значение>* используется для определения значений параметров по умолчанию.
- *Plugins.<ИмяПлагина>.<ИмяСессии>.<Параметр>=<Значение>* используется для определения отдельных наборов параметров для различных целей мониторинга посредством именованных сессий.

Все имена параметров должны соответствовать следующим требованиям:

- рекомендуется писать названия плагинов с заглавной буквы
- параметр должен начинаться с заглавной буквы
- специальные символы не допускаются
- вложенность не ограничена максимальным уровнем
- количество параметров не ограничено

Например, для того, чтобы активные проверки, не имеющие интервала обновления *по расписанию*, выполнялись сразу же после перезапуска агента только для плагина Uptime, задайте в файле конфигурации `Plugins.Uptime.System.ForceActiveChecl`

Аналогично, чтобы задать пользовательский лимит на параллельные проверки для плагина CPU, задайте в файле конфигурации `Plugins.CPU.System.Capacity=N`.

Значения по умолчанию

В конфигурационном файле можно выставить значения по умолчанию для параметров, относящихся к соединению (URI, имя пользователя, пароль и т.д.), в следующем формате:

```
Plugins.<      >.Default.<      >=<      >
```

Например: `Plugins.Mysql.Default.Username=zabbix`, `Plugins.MongoDB.Default.Uri=tcp://127.0.0.1:27017` и т.д.

Если значение для такого параметра не задано в ключе элемента данных или в параметрах именованной сессии, плагин будет использовать это значение по умолчанию. Если значение параметра по умолчанию тоже не определено, то будет использоваться жёстко заданное в коде значение по умолчанию.

**Note:**
Если элемент данных не имеет никаких параметров, Zabbix агент 2 будет пытаться собрать метрику, используя значения, определённые в секции значений параметров по умолчанию.

Именованные сессии

Именованные сессии представляют дополнительный уровень параметров плагина и могут использоваться для определения отдельных наборов параметров аутентификации для каждого наблюдаемого экземпляра. Каждый параметр именованной сессии должен иметь следующую структуру:

```
Plugins.<      >.Sessions.<      >.<      >=<      >
```

Имя сессии может использоваться в качестве параметра строка_подключения в ключе элемента данных вместо указания URI, имени пользователя и/или пароля по отдельности.

В ключах элементов данных первым параметром может быть либо строка_подключения, либо URI. Если первый параметр ключа не совпадает ни с каким именем сессии, он будет рассматриваться как URI. Обратите внимание, что передача встроенных в URI учётных данных не поддерживается, вместо этого рекомендуется использовать параметры именованных сессий.

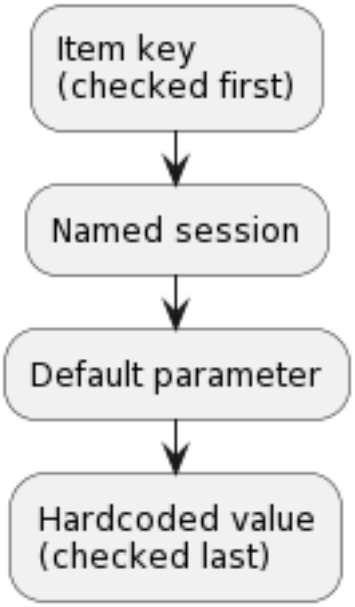
Список доступных параметров именованных сессий зависит от плагина.

Можно переопределить параметры сессии, указывая новые значения в параметрах ключа элемента данных (смотри пример).

Если параметр не определён для именованной сессии, Zabbix агент 2 будет использовать значение, определённое в параметрах плагина по умолчанию.

Приоритет параметров

Плагины Zabbix агента 2 ищут значения относящихся к соединению параметров в следующем порядке:



1. Первый параметр ключа элемента данных сравнивается с именами сессий. Если совпадений не найдено, он рассматривается как реальное значение, в этом случае шаг 3 пропускается. Если же совпадение найдено, то значение этого параметра (как правило, URI) должно быть определено в параметрах сессии.
2. Остальные параметры будут браться из ключа элемента данных, если они там есть.
3. Если параметр ключа элемента данных (например, пароль) не указан, плагин будет искать соответствующий параметр именованной сессии.
4. Если параметр сессии также не указан, будет использоваться значение, определённое в соответствующих параметрах по умолчанию.
5. Если всё равно значение определить не удалось, то плагин будет использовать жёстко заданное в коде значение по умолчанию.

Пример 1

Мониторинг двух экземпляров «MySQL1» и «MySQL2».

Параметры конфигурации:

```
Plugins.Mysql.Sessions.MySQL1.Uri=tcp://127.0.0.1:3306
Plugins.Mysql.Sessions.MySQL1.User=mysql1_user
Plugins.Mysql.Sessions.MySQL1.Password=      _
Plugins.Mysql.Sessions.MySQL2.Uri=tcp://192.0.2.0:3306
Plugins.Mysql.Sessions.MySQL2.User=mysql2_user
Plugins.Mysql.Sessions.MySQL2.Password=    _
```

В результате таких настроек, каждое имя сессии может использоваться в качестве параметра _ в ключах элементов данных, например: `mysql.ping[MySQL1]`, `mysql.ping[MySQL2]`.

Пример 2

Предоставление некоторых параметров в ключе элемента данных.

Параметры конфигурации:

```
Plugins.Postgres.Sessions.Session1.Uri=tcp://192.0.2.234:5432
Plugins.Postgres.Sessions.Session1.User=old_username
Plugins.Postgres.Sessions.Session1.Password=session_password
```

Ключ элемента данных: `pgsql.ping[session1,new_username,,postgres]`

В результате таких настроек агент будет подключаться к PostgreSQL, используя следующие параметры:

- URI из параметров сессии: `192.0.2.234:5432`
- Имя пользователя из ключа элемента данных: `new_username`
- Пароль из параметров сессии (так как он опущен в ключе): `session_password`
- Имя базы данных из ключа элемента данных: `postgres`

Пример 3

Сбор метрик, используя параметры конфигурации по умолчанию.

Параметры конфигурации:

```
Plugins.PostgreSQL.Default.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Default.User=zabbix
Plugins.PostgreSQL.Default.Password=password
```

Ключ элемента данных: `pgsql.ping[,,,postgres]`

В результате таких настроек, агент будет подключаться к PostgreSQL, используя следующие параметры:

- URI по умолчанию: `192.0.2.234:5432`
- Имя пользователя по умолчанию: `zabbix`
- Пароль по умолчанию: `password`
- Имя базы данных из ключа элемента данных: `postgres`

Соединения

Некоторые плагины поддерживают сбор метрик с нескольких экземпляров одновременно. Можно наблюдать как локальные, так и удаленные экземпляры. Поддерживаются TCP и Unix-сокет соединения.

Рекомендуется настроить плагины так, чтобы они соединения с экземплярами поддерживали в открытом состоянии. Преимущества — снижение перегрузки сети, задержек и использования CPU и памяти из-за меньшего количества соединений. Об этом позаботится клиентская библиотека.

**Note:**
Период времени, в течение которого неиспользуемые соединения должны оставаться открытыми, можно задать параметром *Plugins.<ИмяПлагина>.KeepAlive*.
Пример: *Plugins.Memcached.KeepAlive*

Встроенные плагины

Встроенные плагины Zabbix agent 2 доступны сразу после установки. Щелкните имя плагина, чтобы перейти в репозиторий плагина с дополнительной информацией.

| Имя плагина | Описание | Поддерживаемые ключи элементов данных | Комментарии |
|---|---|---|---|
| Агент | Метрики используемого агента Zabbix. | agent.hostname, agent.ping, agent.version | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Ceph | Мониторинг Ceph. | ceph.df.details, ceph.osd.stats, ceph.osd.discovery, ceph.osd.dump, ceph.ping, ceph.pool.discovery, ceph.status | |
| CPU | Мониторинг CPU системы (количество CPU/ядер CPU, обнаруженные CPU, процент загрузки). | system.cpu.discovery, system.cpu.num, system.cpu.util | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Docker | Мониторинг контейнеров Docker. | docker.container_info, docker.container_stats, docker.containers, docker.containers.discovery, docker.data_usage, docker.images, docker.images.discovery, docker.info, docker.ping | См. также: Параметры конфигурации |
| File | Сбор метрик файлов. | vfs.file.cksum, vfs.file.contents, vfs.file.exists, vfs.file.md5sum, vfs.file.regexp, vfs.file.regmatch, vfs.file.size, vfs.file.time | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Kernel | Мониторинг ядра. | kernel.maxfiles, kernel.maxproc | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Log | Мониторинг файлов журналов. | log, log.count, logrt, logrt.count | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys.<br><br>См. также:<br>Параметры конфигурации плагина (Unix/Windows) |
| Memcached | Мониторинг сервера Memcached. | memcached.ping, memcached.stats | |
| Modbus | Чтение данных Modbus. | modbus.get | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |

| Имя плагина | Описание | Поддерживаемые ключи элементов данных | Комментарии |
|---|---|---|---|
| MQTT | Получает опубликованные значения тем MQTT. | mqtt.get | Чтобы настроить зашифрованное соединение с брокером MQTT, укажите параметры TLS в файле конфигурации агента как параметры именованной сессии или значения по умолчанию. В настоящее время параметры TLS нельзя передавать как параметры ключа элемента данных. |
| MySQL | Мониторинг MySQL и его форков. | 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 | Чтобы настроить зашифрованное соединение с базой данных, укажите параметры TLS в файле конфигурации агента как параметры именованной сессии или значения по умолчанию. В настоящее время параметры TLS нельзя передавать как параметры ключа элемента данных. |
| NetIf | Мониторинг сетевых интерфейсов. | net.if.collisions, net.if.discovery, net.if.in, net.if.out, net.if.total | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Oracle | Мониторинг 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 | Перед использованием плагина установите Oracle Instant Client. |
| Proc | Процент загрузки CPU процесса. | proc.cpu.util | Поддерживаемый ключ имеет те же параметры, что и ключ агента Zabbix key. |
| Redis | Мониторинг сервера Redis. | redis.config, redis.info, redis.ping, redis.slowlog.count | Чтобы настроить зашифрованное соединение с Redis, укажите параметры TLS в файле конфигурации агента как параметры именованной сессии или значения по умолчанию. Параметры TLS нельзя передавать как параметры ключа элемента данных. Обратите внимание, что некорректная или иным образом недопустимая конфигурация TLS может помешать запуску Zabbix agent 2, поэтому перед включением TLS проверьте файлы сертификатов, права доступа и пути. |

| Имя плагина | Описание | Поддерживаемые ключи элементов данных | Комментарии |
|---|---|---|---|
| Smart | Мониторинг S.M.A.R.T. | smart.attribute.discovery, smart.disk.discovery, smart.disk.get | Минимально требуемая версия smartctl — 7.1.<br>Для пользователя, запускающего Zabbix agent 2, требуются права sudo/root для smartctl. Плагин использует только следующие команды:<br>`/usr/sbin/smartctl -a *`<br>`/usr/sbin/smartctl --scan *`<br>`/usr/sbin/smartctl -j -V`<br><br>Поддерживаемые keys можно использовать с Zabbix agent 2 только в Linux/Windows, как для пассивной, так и для активной проверки.<br>См. также:<br>Параметры конфигурации |
| SW | Список установленных пакетов. | system.sw.packages, system.sw.packages.get | Поддерживаемые ключи имеют те же параметры, что и ключ агента Zabbix key. |
| Swap | Размер пространства swap в байтах/процентах. | system.swap.size | Поддерживаемый ключ имеет те же параметры, что и ключ агента Zabbix key. |
| SystemRun | Выполняет указанный команду. | system.run | Поддерживаемый ключ имеет те же параметры, что и ключ агента Zabbix key.<br><br>См. также:<br>Параметры конфигурации плагина (Unix/Windows) |
| Systemd | Мониторинг служб systemd. | systemd.unit.discovery, systemd.unit.get, systemd.unit.info | |
| TCP | Проверка доступности TCP-соединения. | net.tcp.port | Поддерживаемый ключ имеет те же параметры, что и ключ агента Zabbix key. |
| UDP | Мониторинг доступности и производительности служб UDP. | net.udp.service, net.udp.service.perf | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Uname | Получение информации о системе. | system.hostname, system.sw.arch, system.uname | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| Uptime | Сбор метрик времени работы системы. | system.uptime | Поддерживаемый ключ имеет те же параметры, что и ключ агента Zabbix key. |
| VFSDev | Сбор метрик VFS. | vfs.dev.discovery, vfs.dev.read, vfs.dev.write | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| WebCertificate | Мониторинг сертификатов TLS/SSL веб-сайтов. | web.certificate.get | |
| WebPage | Мониторинг веб-страниц. | web.page.get, web.page.perf, web.page.regexp | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |

| Имя плагина | Описание | Поддерживаемые ключи элементов данных | Комментарии |
|---|---|---|---|
| ZabbixAsync | Асинхронный сбор метрик. | 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 | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| ZabbixStats | Внутренние метрики сервера Zabbix/прокси или количество задержанных элементов данных в очереди. | zabbix.stats | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |
| ZabbixSync | Синхронный сбор метрик. | 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. | Поддерживаемые ключи имеют те же параметры, что и ключи агента Zabbix keys. |

Загружаемые плагины

Загружаемые плагины Zabbix agent 2 недоступны из коробки и должны устанавливаться отдельно:

- В Linux можно использовать пакеты (например, установить Ember+ в Ubuntu с помощью `apt install zabbix-agent2-plugi` или собрать плагины
- В Windows можно установить из MSI или собрать плагины

**Attention:**
Перед установкой плагина, пожалуйста, проверьте его файл README. Он может содержать особые требования и инструкции по установке.

Нажмите на имя плагина, чтобы перейти в репозиторий плагина, где находится файл README с дополнительной информацией.

| Plugin name | Description | Supported item keys | Comments |
|---|---|---|---|
| Ember+ | Мониторинг Ember+. | ember.get | См. также параметры конфигурации плагина Ember+. |

| Plugin name | Description | Supported item keys | Comments |
|---|---|---|---|
| MongoDB | Мониторинг серверов и кластеров MongoDB (документо-ориентированная распределенная база данных). | 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 | Для настройки зашифрованных подключений к базе данных укажите параметры TLS в файле конфигурации агента как параметры именованной сессии.<br>В настоящее время параметры TLS нельзя передавать как параметры ключа элемента данных.<br><br>См. также параметры конфигурации плагина MongoDB. |
| MSSQL | Мониторинг базы данных 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 | Для настройки зашифрованного подключения к базе данных укажите параметры TLS в файле конфигурации агента как параметры именованной сессии или значения по умолчанию. В настоящее время параметры TLS нельзя передавать как параметры ключа элемента данных.<br><br>См. также параметры конфигурации плагина MSSQL. |
| NVIDIA GPU | Мониторинг NVIDIA GPU. | 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 | См. также параметры конфигурации плагина NVIDIA GPU. |

| Plugin name | Description | Supported item keys | Comments |
|---|---|---|---|
| PostgreSQL | Мониторинг PostgreSQL и его форков. | pgsql.autovacuum.count,<br>pgsql.archive,<br>pgsql.bgwriter,<br>pgsql.cache.hit,<br>pgsql.connections,<br>pgsql.custom.query,<br>pgsql.dbstat,<br>pgsql.dbstat.sum,<br>pgsql.db.age,<br>pgsql.db.bloating_tables,<br>pgsql.db.discovery,<br>pgsql.db.size,<br>pgsql.locks,<br>pgsql.oldest.xid,<br>pgsql.ping,<br>pgsql.queries,<br>pgsql.replication.count,<br>pgsql.replication.process,<br>pgsql.replication.process.discovery,<br>pgsql.replication.recovery_role,<br>pgsql.replication.status,<br>pgsql.replication_lag.b,<br>pgsql.replication_lag.sec,<br>pgsql.uptime,<br>pgsql.version,<br>pgsql.wal.stat | Для настройки зашифрованных подключений к базе данных укажите параметры TLS в файле конфигурации агента как параметры именованной сессии или значения по умолчанию.<br>В настоящее время параметры TLS нельзя передавать как параметры ключа элемента данных.<br><br>См. также параметры конфигурации плагина PostgreSQL. |

**Note:**
Загружаемые плагины при запуске с параметрами:<br> - `-V --version` - выводят версию плагина и информацию о лицензии;<br> - `-h --help` - выводят справочную информацию.

### 3. Модули веб-интерфейса

Обзор

Можно улучшить функциональность веб-интерфейса Zabbix, добавив сторонние модули или разработав свои собственные модули без необходимости изменения исходного кода Zabbix.

Обратите внимание, что код модуля будет работать с теми же привилегиями, что и исходный код Zabbix. Это означает, что:

- сторонние модули могут быть вредоносными. Вы должны доверять модулям, которые вы устанавливаете;
- ошибки в коде стороннего модуля могут привести к сбою веб-интерфейса. Если это произойдёт, просто удалите код модуля из веб-интерфейса. Как только вы перезагрузите веб-интерфейс Zabbix, вы увидите уведомление о том, что некоторые модули отсутствуют. Перейдите к Администрированию модулей (в разделе *Администрирование* → *Общие* → *Модули* (*Administration* → *General* → *Modules*)) и нажмите *Сканировать директорию (Scan directory)*, чтобы удалить несуществующие модули из базы данных.

Установка

Всегда читайте руководство по установке для конкретного модуля. Рекомендуется устанавливать новые модули по одному, чтобы легче выявлять сбои.

Непосредственно перед установкой модуля:

- Убедитесь, что вы загрузили модуль из надежного источника. Установка вредоносного кода может привести к последствиям, таким как потеря данных
- Разные версии одного и того же модуля (с одним и тем же ID) можно устанавливать параллельно, но одновременно можно включить только одну версию

Шаги по установке модуля:

- Распакуйте ваш модуль в отдельную папку внутри папки `modules` веб-интерфейса Zabbix
- Убедитесь, что папка модуля содержит как минимум файл manifest.json
- Перейдите в Управление модулями и нажмите кнопку *Сканировать каталог*
- Новый модуль появится в списке вместе со своей версией, автором, описанием и статусом
- Включите модуль, нажав на его статус

Устранение неполадок:

| Проблема | Решение |
|---|---|
| *Модуль не появился в списке* | Если ваш модуль не появился в списке, убедитесь, что manifest.json существует в modules/your-module/. Если его нет, возможно, вы распаковали модуль не в тот каталог. Если файл есть, модуль может быть несовместим с вашей версией Zabbix. Также проверьте, что пользователь веб-сервера имеет как минимум права на чтение и поиск (`r-x`) для папки модуля и всех подкаталогов, а также права на чтение (`r--`) для всех файлов внутри. |
| *Веб-интерфейс аварийно завершил работу* | Код модуля несовместим с текущей версией Zabbix или конфигурацией сервера. Удалите файлы модуля и перезагрузите веб-интерфейс. Вы увидите уведомление о том, что некоторые модули отсутствуют. Перейдите в Управление модулями и снова нажмите *Сканировать каталог*, чтобы удалить несуществующие модули из базы данных. |
| *Появляется сообщение об ошибке с одинаковым пространством имен, ID или действиями* | Новый модуль попытался зарегистрировать пространство имен, ID или действия, которые уже зарегистрированы другими включенными модулями. Перед включением нового модуля отключите конфликтующий модуль (указанный в сообщении об ошибке). |
| *Появляются технические сообщения об ошибках* | Сообщите об ошибках разработчику модуля. |

Разработка модулей

Информацию о разработке пользовательских модулей смотрите в Центре разработчиков [en].

## 22. Приложения

**Обзор** В этом разделе содержатся справочные материалы, параметры конфигурации, технические спецификации и руководства по процедурам, дополняющие основное руководство.

**Основной справочник**

- Установка и настройка — создание базы данных, установка и настройка сервера и агента.
- Конфигурация демонов — параметры конфигурации для Zabbix сервера, прокси, агента и так далее.
- Элементы данных — справочник по ключам элементов данных и поддерживаемым типам данных.
- Макросы — синтаксис и применение встроенных и пользовательских макросов.

### 1. Установка и настройка

В этом разделе рассматриваются создание базы данных, настройка безопасности, процедуры настройки для конкретной платформы и конфигурация компонентов.

Пожалуйста, воспользуйтесь боковым меню, чтобы найти процедуру, относящуюся к вашей среде.

### 1. Создание базы данных

Обзор

База данных Zabbix должна быть создана во время установки сервера или прокси Zabbix.

В этом разделе приведены инструкции по созданию базы данных Zabbix. Для каждой поддерживаемой базы данных доступен отдельный набор инструкций.

**Note:**
Чтобы повысить безопасность базы данных путем создания ролей/пользователей базы данных с минимальными привилегиями, см. рекомендации по созданию базы данных для каждой поддерживаемой базы данных: <br><br>
- MySQL/MariaDB
- PostgreSQL/TimescaleDB

Для настройки защищенных TLS-соединений см. Безопасное подключение к базе данных.

UTF-8 - единственная кодировка, поддерживаемая Zabbix. Известно, что она работает без каких-либо проблем с безопасностью. Пользователям следует учитывать, что при использовании некоторых других кодировок известны проблемы с безопасностью. Для перехода на UTF-8 см. Восстановление набора символов и сопоставления базы данных Zabbix. См. также Ограничения фильтрации с сопоставлениями utf8mb4.

**Note:**
Если установка выполняется из репозитория Zabbix Git, перед переходом к следующим шагам необходимо выполнить следующую команду: <br><br> `make dbschema`

MySQL/MariaDB

Поддерживаются наборы символов utf8 (также известный как utf8mb3) и utf8mb4 (с сопоставлениями utf8_bin и utf8mb4_bin соответственно) для корректной работы сервера/прокси Zabbix с базой данных MySQL. Для новых установок рекомендуется использовать utf8mb4.

Детерминированные триггеры необходимо создавать во время импорта схемы. В MySQL и MariaDB для этого требуется установить `GLOBAL log_bin_trust_function_creators = 1`, если включено бинарное журналирование, нет привилегий суперпользователя и `log_bin_trust_function_creators = 1` не задан в файле конфигурации MySQL.

Если вы устанавливаете Zabbix из пакетов или контейнеров, см. соответствующие инструкции.

Если вы устанавливаете Zabbix из исходных кодов:

- Создайте и настройте базу данных и пользователя.

```
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;
```

- Импортируйте данные в базу данных и установите набор символов utf8mb4 по умолчанию (предполагается, что вы находитесь в корневом каталоге исходных кодов Zabbix). Для базы данных прокси Zabbix следует импортировать только `schema.sql` (без `images.sql` и `data.sql`).

```
cd database/mysql
mysql -uzabbix -p<password> zabbix < schema.sql
#### stop here if you are creating database for Zabbix proxy
mysql -uzabbix -p<password> zabbix < images.sql
mysql -uzabbix -p<password> --default-character-set=utf8mb4 zabbix < data.sql
```

`log_bin_trust_function_creators` можно отключить после успешного импорта схемы:

```
mysql -uroot -p<password>

mysql> SET GLOBAL log_bin_trust_function_creators = 0;
mysql> quit;
```

PostgreSQL

У вас должен быть пользователь базы данных с правами на создание объектов базы данных.

Если вы устанавливаете Zabbix из пакетов или контейнеров, см. соответствующие инструкции.

Если вы устанавливаете Zabbix из исходных кодов:

- Создайте пользователя базы данных.

Следующая команда shell создаст пользователя zabbix. Укажите пароль при запросе и повторите пароль (обратите внимание, сначала у вас может быть запрошен пароль sudo):

```
sudo -u postgres createuser --pwprompt zabbix
```

- Создайте базу данных.

Следующая команда shell создаст базу данных zabbix (последний параметр) с ранее созданным пользователем в качестве владельца (-O zabbix).

```
sudo -u postgres createdb -O zabbix -E Unicode -T template0 zabbix
```

- Импортируйте начальную схему и данные (предполагается, что вы находитесь в корневом каталоге исходных кодов Zabbix). Для базы данных Zabbix proxy следует импортировать только schema.sql (без images.sql и data.sql).

```
cd database/postgresql
cat schema.sql | sudo -u zabbix psql zabbix
#### stop here if you are creating database for Zabbix proxy
cat images.sql | sudo -u zabbix psql zabbix
cat data.sql | sudo -u zabbix psql zabbix
```

**Attention:**
Приведенные выше команды даны в качестве примера, который будет работать в большинстве установок GNU/Linux. Вы можете использовать другие команды в зависимости от того, как настроена ваша система/база данных, например: <br><br> psql -U <username> <br><br> Если у вас возникнут трудности при настройке базы данных, обратитесь к администратору базы данных.

TimescaleDB

Инструкции по созданию и настройке TimescaleDB приведены в отдельном разделе.

SQLite

Использование SQLite поддерживается только для **прокси Zabbix**!

База данных будет создана автоматически, если она не существует.

Вернитесь к разделу установки.

**2. Восстановление набора символов и сопоставления базы данных Zabbix**

MySQL/MariaDB

Исторически, MySQL и производные от неё использовали «utf8» как синоним для utf8mb3 — собственная 3-байтовая реализация командой разработчиков MySQL стандарта UTF8, который является 4-байтовым. Начиная с версии MySQL 8.0.28 и MariaDB 10.6.1, набор символов «utf8mb3» является устаревшим, и в какой-то момент его поддержка будет прекращена, в то время как «utf8» станет ссылкой на «utf8mb4». Начиная с версии Zabbix 6.0, поддерживается «utf8mb4». Во избежание проблем в будущем, строго рекомендуется использовать «utf8mb4». Другим преимуществом переключения на «utf8mb4» является поддержка дополнительных символов Юникода.

**Warning:**
Поскольку версии Zabbix до 6.0 не знают об utf8mb4, обеспечьте сначала обновление сервера Zabbix и схемы базы данных до версии 6.0.x или более новой, прежде чем выполнять переход на utf8mb4.

**1.** Проверьте набор символов базы данных и сопоставление.

Например:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| latin2                   | latin2 _general_ci   |
+--------------------------+----------------------+
```

Или:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8                     | utf8_bin             |
+--------------------------+----------------------+
```

Как мы видим, набор символов в выводе — не «utf8mb4», а сопоставление — не «utf8mb4_bin», поэтому нам нужно их исправить.

**2.** Остановите Zabbix.

**3.** Создайте резервную копию базы данных!

**4.** Исправьте набор символов и сопоставление на уровне базы данных:

```
alter database <your DB name> character set utf8mb4 collate utf8mb4_bin;
```

Исправленные значения:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8mb4                  | utf8mb4_bin          |
+--------------------------+----------------------+
```

**5.** Загрузите скрипт, чтобы исправить набор символов и сопоставление на уровне таблицы и столбца:

```
mysql <your DB name> < utf8mb4_convert.sql
```

**6.** Выполните скрипт:

```
               SET @ZABBIX_DATABASE = '<            >';
If MariaDB →   set innodb_strict_mode = OFF;
               CALL zbx_convert_utf8();
If MariaDB →   set innodb_strict_mode = ON;
               drop procedure zbx_convert_utf8;
```

Обратите внимание, что кодировка «utf8mb4» может занимать чуть больше места на диске, чем раньше.

**7.** Если ошибок нет — вы можете захотеть создать резервную копию базы данных с исправленной базой.

**8.** Запустите Zabbix.

#### 3. Обновление базы данных для использования первичных ключей

Обзор

В этом разделе приведены инструкции по ручному обновлению таблиц в существующих установках до первичных ключей.

Переход на первичные ключи оптимизирует индексирование и доступ к данным, что может ускорить выполнение запросов и сэкономить место. Это также улучшает управление данными и синхронизацию в кластерных конфигурациях, помогая масштабированию и обеспечивая надежную работу системы даже при отказе некоторых серверов.

> **Attention:**
> Инструкции, приведенные на этой странице, предназначены для опытных пользователей и могут потребовать адаптации под вашу конкретную конфигурацию. Переход на первичные ключи может быть длительным и ресурсоемким. Убедитесь, что доступно достаточно свободного места на диске; в зависимости от размера базы данных и хранимых данных процесс может потребовать до 2,5 раза больше места, чем сейчас занимают таблицы истории.

Первичные ключи используются для всех таблиц в новых установках начиная с Zabbix 6.0.

Автоматического обновления базы данных до первичных ключей нет; однако существующие установки можно обновить вручную **после** обновления Zabbix server до 6.0 или более новой версии.

**Attention:**
Начиная с Zabbix 7.0, при обновлении таблиц до первичных ключей таблицы также переводятся на использование типов данных double precision. <br><br> Если у вас Zabbix 7.0 (или новее), таблицы уже используют double precision. Однако инструкции на этой странице по-прежнему можно использовать для обновления таблиц до первичных ключей без влияния на таблицы, которые уже используют double precision. <br><br> Если у вас Zabbix 6.4 (или более ранняя версия), рекомендуется сначала обновить таблицы до double precision. Дополнительную информацию см. в разделе Обновление до числовых значений расширенного диапазона в документации Zabbix 7.0.

Доступны инструкции для:

- MySQL
- PostgreSQL
- TimescaleDB

Важные примечания

Чтобы выполнить обновление базы данных:

1. Остановите сервер Zabbix.

Настоятельно рекомендуется останавливать сервер Zabbix на время обновления. Однако, если это абсолютно необходимо, вы можете выполнить обновление, пока сервер работает (только для MySQL, MariaDB и PostgreSQL без TimescaleDB).

2. Создайте резервную копию базы данных.
3. Установите последнюю версию пакета zabbix-sql-scripts, совместимую с вашей версией Zabbix (например, для RHEL: `dnf install zabbix-sql-scripts`).
4. Запустите скрипты для вашей базы данных.
5. Запустите сервер Zabbix.

**Warning:**
Запускайте скрипты только для базы данных сервера. Прокси не получит преимуществ от этого обновления.

Если база данных использует разделы, обратитесь за помощью к администратору БД или в службу поддержки Zabbix.

CSV-файлы можно удалить после успешного обновления до первичных ключей.

При необходимости веб-интерфейс Zabbix можно перевести в режим обслуживания.

MySQL

Экспорт и импорт необходимо выполнять в tmux/screen, чтобы гарантировать, что сеанс не будет прерван.

Смотрите также: Важные примечания

MySQL 8.0+ с mysqlsh

Этот метод можно использовать при работающем сервере Zabbix, но рекомендуется остановить сервер на время обновления. MySQL Shell (*mysqlsh*) должен быть установлен и иметь возможность подключаться к БД.

- Войдите в консоль MySQL как root (рекомендуется) или как любой пользователь с привилегиями FILE.
- Запустите MySQL с включенной переменной local_infile.
- Переименуйте старые таблицы и создайте новые, выполнив `history_upgrade_prepare.sql`.

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

- Экспортируйте и импортируйте данные.

Подключитесь через mysqlsh. Если используется подключение через сокет, может потребоваться указать путь.

```
sudo mysqlsh -uroot -S /run/mysqld/mysqld.sock --no-password -Dzabbix
```

Переключитесь в режим JavaScript с помощью:

```
\js
```

Затем выполните приведенный ниже код (CSVPATH можно изменить при необходимости):

```
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" });
```

Если вы получите сообщение "JavaScript is not supported", значит в вашей установке MySQL Shell отсутствует поддержка JS. В этом случае установите официальный пакет MySQL Shell от Oracle (или соберите его из исходного кода), чтобы был включен режим JavaScript.

- Следуйте инструкциям после миграции, чтобы удалить старые таблицы.

MariaDB/MySQL 8.0+ без mysqlsh

Этот способ обновления занимает больше времени и должен использоваться только в том случае, если обновление с помощью *mysqlsh* невозможно.

Обновление таблиц

- Войдите в консоль MySQL как root (рекомендуется) или любой пользователь с привилегиями FILE.
- Если выполняется миграция при работающем сервере Zabbix, запустите MySQL с включенной переменной local_infile.
- Переименуйте старые таблицы и создайте новые, выполнив `history_upgrade_prepare.sql`:

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

Миграция при остановленном сервере

`max_execution_time` (в MySQL) или `max_statement_time` (в MariaDB) необходимо отключить перед миграцией данных, чтобы избежать тайм-аута во время миграции.

Для MySQL:

```
SET @@max_execution_time=0;
```

Для 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;
```

Следуйте инструкциям после миграции, чтобы удалить старые таблицы.

Миграция при работающем сервере

Проверьте, для каких путей включен импорт/экспорт:

```
mysql> SELECT @@secure_file_priv;
+-----------------------+
| @@secure_file_priv    |
+-----------------------+
| /var/lib/mysql-files/ |
+-----------------------+
```

Если значение *secure_file_priv* является путем к каталогу, экспорт/импорт будет выполняться для файлов в этом каталоге. В этом случае соответствующим образом измените пути к файлам в запросах или установите значение *secure_file_priv* в пустую строку на время обновления.

Если значение *secure_file_priv* пустое, экспорт/импорт можно выполнять из любого расположения.

Если значение *secure_file_priv* равно NULL, задайте путь к каталогу, который содержит экспортированные данные таблиц (в примере выше это '/var/lib/mysql-files/').

Дополнительную информацию см. в документации MySQL или документации MariaDB.

`max_execution_time` (в MySQL) или `max_statement_time` (в MariaDB) необходимо отключить перед экспортом данных, чтобы избежать тайм-аута во время экспорта.

Для MySQL:

```
SET @@max_execution_time=0;
```

Для 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
```

Следуйте инструкциям после миграции, чтобы удалить старые таблицы.

PostgreSQL

Экспорт и импорт необходимо выполнять в tmux/screen, чтобы сеанс не был разорван. Для установок с TimescaleDB пропустите этот раздел и перейдите к PostgreSQL + TimescaleDB.

См. также: Важные примечания

Обновление таблиц

- Переименуйте таблицы с помощью `history_upgrade_prepare.sql`:

```
sudo -u zabbix psql zabbix < /usr/share/zabbix/sql-scripts/postgresql/option-patches/history_upgrade_prepa
```

Миграция с остановленным сервером

- Экспортируйте текущую историю, импортируйте ее во временную таблицу, затем вставьте данные в новые таблицы, игнорируя дубликаты:

```
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;
```

См. советы по повышению производительности INSERT: PostgreSQL: Bulk Loading Huge Amounts of Data, Checkpoint Distance and Amount of WAL.

- Следуйте инструкциям после миграции, чтобы удалить старые таблицы.

Миграция при работающем сервере

- Экспортируйте текущую историю, импортируйте ее во временную таблицу, затем вставьте данные в новые таблицы, игнорируя дубликаты:

```
\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;
```

- Следуйте инструкциям после миграции, чтобы удалить старые таблицы.

PostgreSQL + TimescaleDB

Экспорт и импорт необходимо выполнять в tmux/screen, чтобы сеанс не был разорван. Во время обновления сервер Zabbix должен быть остановлен.

См. также: Important notes

- Переименуйте таблицы с помощью `history_upgrade_prepare.sql`.
  - Если сжатие включено (в стандартной установке), выполните скрипт из `/usr/share/zabbix/sql-scripts/postgresql`

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
```

  - Если сжатие отключено, выполните скрипт из `/usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-`

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
```

- Запустите скрипты миграции hypertable TimescaleDB в зависимости от настроек сжатия:
  - Если сжатие включено (в стандартной установке), выполните скрипты из `/usr/share/zabbix/sql-scripts/postgresc`

```
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
```

  - Если сжатие отключено, выполните скрипты из `/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
```

См. также: Tips для повышения производительности INSERT.

- Следуйте инструкциям после миграции, чтобы удалить старые таблицы.

Пост-миграция

Для всех баз данных после завершения миграции сделайте следующее:

- Проверьте, что всё работает, как ожидается.
- Удалите старые таблицы:

```
DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;
```

- Для TimescaleDB также удалите следующую старую таблицу:

```
DROP TABLE trends_old;
```

Смотрите также

- Подготовка таблицы журнала аудита к партиционированию

#### 4. Подготовка таблицы журнала аудита для партиционирования

Обзор

Некоторые базы данных (например, MySQL) требуют, чтобы столбец партиционирования был частью уникального ограничения таблицы (table's unique constraint). Поэтому для партиционирования таблицы `auditlog` по времени первичный ключ необходимо изменить с `auditid` на составной ключ `auditid` + `clock`.

В этом разделе приведены инструкции по изменению первичного ключа таблицы `auditlog`.

**Attention:**
Инструкции, представленные на этой странице, предназначены для опытных пользователей. Обратите внимание, что эти инструкции, возможно, придется скорректировать для вашей конкретной конфигурации. Изменение первичного ключа также может быть несовместимо с будущими обновлениями, поэтому может потребоваться ручная обработка будущих обновлений. <br><br> Изменение первичного ключа может быть ресурсоёмкой операцией, которая занимает много времени в зависимости от размера таблицы `auditlog`. Рекомендуется остановить сервер Zabbix и переключить веб-интерфейс Zabbix в режим обслуживания на время изменения. Однако, если это абсолютно необходимо, есть способ изменить первичный ключ без простоя (смотрите ниже).

Партиционирование таблицы auditlog может улучшить, например, процедуру очистки истории в больших инсталляциях.

Хотя очистка истории Zabbix в настоящее время не может использовать преимущества партиционированных таблиц (за исключением TimescaleDB), вы можете отключить очистку истории Zabbix и удалять партиции с помощью скриптов.

Начиная с версии Zabbix 7.0, таблица auditlog для TimescaleDB была преобразована в гипертаблицу, что позволяет администратору удалять данные по частям. Чтобы обновить существующую таблицу auditlog до гипертаблицы, смотрите Обновление схемы TimescaleDB.

MySQL

Важные замечания по перестроению индексов

MySQL автоматически перестраивает индексы для первичного ключа во время операции ALTER TABLE. Однако, чтобы обеспечить оптимальную производительность базы данных, настоятельно рекомендуется также вручную перестроить индексы с помощью оператора OPTIMIZE TABLE.

Перестроение индексов может временно потребовать столько же дополнительного дискового пространства, сколько использует сама таблица. Чтобы получить текущий размер данных и индексов, можно выполнить следующие операторы:

```
ANALYZE TABLE auditlog;
SHOW TABLE STATUS LIKE 'auditlog';
```

Если доступное дисковое пространство вызывает беспокойство, следуйте инструкциям Изменение первичного ключа без простоя. Также доступны другие варианты:

- Увеличение параметра MySQL sort_buffer_size [en] может помочь сократить использование дискового пространства при ручном перестроении индексов. Однако изменение этой переменной может повлиять на общее использование памяти базы данных.
- Рассмотрите возможность освобождения места путем удаления потенциально ненужных данных.
- Рассмотрите возможность уменьшения параметра очистки истории *Период хранения данных (Data storage period)* перед выполнением очистки истории.

Изменение первичного ключа с простоем

1. Удалите текущий первичный ключ таблицы auditlog и добавьте новый первичный ключ.

```
ALTER TABLE auditlog DROP PRIMARY KEY, ADD PRIMARY KEY (auditid, clock);
```

2. Перестройте индексы (необязательно, но настоятельно рекомендуется, смотрите Важные примечания по перестроению индексов).

```
OPTIMIZE TABLE auditlog;
```

Изменение первичного ключа без простоя

Здесь описан ручной метод изменения первичного ключа. В качестве альтернативы вы можете использовать набор инструментов pt-online-schema-change [en] от Percona. Этот набор инструментов автоматически выполняет следующие действия, а также минимизирует пространство, используемое для изменения таблицы auditlog.

1. Создайте новую таблицу с новым первичным ключом и создайте индексы.

```
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. Поменяйте таблицы местами.

```
RENAME TABLE auditlog TO auditlog_old, auditlog_new TO auditlog;
```

3. Скопируйте данные из старой таблицы в новую.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Это можно сделать частями (несколько операторов INSERT INTO с условиями WHERE clock по мере необходимости), чтобы избежать чрезмерного использования ресурсов.

4. Удалите старую таблицу.

```
DROP TABLE auditlog_old;
```

PostgreSQL

Важные замечания по перестроению индексов

PostgreSQL автоматически перестраивает индексы для первичного ключа во время операции ALTER TABLE. Однако настоятельно рекомендуется также вручную перестроить индексы с помощью оператора REINDEX TABLE CONCURRENTLY, чтобы обеспечить оптимальную производительность базы данных.

Перестроение индексов может временно потребовать до трех раз больше дискового пространства, чем сейчас занимают индексы. Чтобы узнать текущий размер индексов, можно выполнить следующий запрос:

```
SELECT pg_size_pretty(pg_indexes_size('auditlog'));
```

Если доступное дисковое пространство вызывает беспокойство, следуйте инструкциям в разделе Изменение первичного ключа без простоя. Также доступны и другие варианты:

- Увеличение параметра PostgreSQL maintenance_work_mem может помочь снизить использование дискового пространства при ручном перестроении индексов. Однако изменение этой переменной может повлиять на общее использование памяти базой данных.
- Если у вас есть другой диск или табличное пространство с большим объемом свободного места, можно изменить расположение временного хранилища для перестроения индекса. Вы можете задать параметр PostgreSQL temp_tablespaces, чтобы указать другое табличное пространство для временных объектов.
- Рассмотрите возможность освобождения места путем удаления потенциально ненужных данных.
- Рассмотрите возможность уменьшения параметра *Период хранения данных* housekeeper перед запуском housekeeper.

Изменение первичного ключа с простоем

1. Удалите текущий первичный ключ таблицы auditlog и добавьте новый первичный ключ.

```
ALTER TABLE auditlog DROP CONSTRAINT auditlog_pkey;
ALTER TABLE auditlog ADD PRIMARY KEY (auditid,clock);
```

2. Перестройте индексы (необязательно, но настоятельно рекомендуется; см. Важные замечания по перестройке индексов).

```
REINDEX TABLE CONCURRENTLY auditlog;
```

Изменение первичного ключа без простоя

Здесь описан ручной метод изменения первичного ключа. В качестве альтернативы можно рассмотреть расширение pg_repack для создания новой таблицы, копирования данных и замены таблиц.

1. Создайте новую таблицу с новым первичным ключом и создайте индексы.

```
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. Поменяйте таблицы местами.

```
ALTER TABLE auditlog RENAME TO auditlog_old;
ALTER TABLE auditlog_new RENAME TO auditlog;
```

3. Скопируйте данные из старой таблицы в новую.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Это можно сделать частями (несколько операторов `INSERT INTO` с условиями `WHERE clock` по мере необходимости), чтобы избежать чрезмерного использования ресурсов.

4. Удалите старую таблицу.

```
DROP TABLE auditlog_old;
```

Смотрите также

- Обновление базы данных для использования первичных ключей

#### 5. Безопасные соединения с базой данных

Обзор

Этот раздел описывает шаги по настройке Zabbix и примеры конфигурации для безопасных соединений TLS между:

| База данных | Компоненты Zabbix |
|---|---|
| MySQL | Веб-интерфейс Zabbix, Zabbix сервер, Zabbix прокси |
| PostgreSQL | Веб-интерфейс Zabbix, Zabbix сервер, Zabbix прокси |

За подробностями настроек шифрования соединений в СУБД, обратитесь к официальной документации вендора ([en]):

- MySQL: соединения между исходным сервером базы данных и репликой.
- MySQL: групповая репликация серверов баз данных и т.п.
- PostgreSQL: опции шифрования.

Все примеры базируются на выпусках GA release MySQL CE (8.0) и PgSQL (13), доступных через официальные репозитории, используя CentOS 8.

Требования

Для настройки шифрования требуется следующее:

- Поддерживаемая разработчиками операционная система с OpenSSL >=1.1.X или альтернатива.

> **Note:**
> Рекомендуется избегать ОС в статусе end-of-life, особенно в случае новых инсталляций

- Механизм базы данных (РСУБД), установленный и поддерживаемый из официального репозитория, предоставленного разработчиком. Операционные системы нередко поставляются с устаревшими версиями программного обеспечения баз данных, в которых поддержка шифрования не реализована; например, системы на базе RHEL 7 и PostgreSQL 9.2, MariaDB 5.5 без поддержки шифрования.

Терминология

Выставление этой опции принуждает использовать TLS соединения к базе данных от Zabbix сервера/прокси и от веб-интерфейса к базе данных:

- required — подключаться, используя TLS как транспортный режим без проверок идентичности;
- verify_ca — подключаться, используя TLS и проверять сертификат;
- verify_full — подключаться, используя TLS, проверять сертификат и проверять, что идентифкация базы данных (CN), указанная в DBHost, соответствует своему сертификату;

Настройка Zabbix

Веб-интерфейс к базе данных

Безопасные подключения к базе данных могут быть настроены во время установки веб-интерфейса:

- Отметьте флажок *TLS шифрование базы данных* на шаге Настройка соединения с базой данных, чтобы включить шифрование на транспортном уровне.
- Отметьте флажок *Удостоверять сертификат базы данных* (который появляется при выборе опции *TLS шифрование базы данных*), чтобы разрешить шифрование с сертификатом.

**Note:**
Для MySQL флажок *TLS шифрование базы данных* отключён, если *Хост базы данных* выставлен в localhost, так как соединение, использующее сокет-файл (на Unix) или разделяемую память (на Windows), не может быть шифрованным.<br> Для PostgreSQL флажок *TLS шифрование базы данных* отключён, если значение поля *Хост базы данных* начинается с косой черты или же пустое.

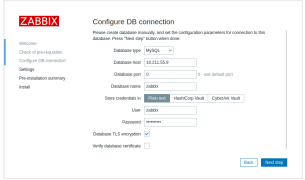
В режиме TLS шифрования с сертификатом (если оба флажка отмечены) становятся доступными следующие параметры:

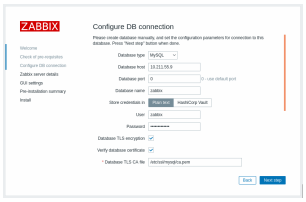
| Параметр | Описание |
|---|---|
| *Файл TLS CA базы данных* | Укажите полный путь к корректному файлу удостоверяющего центра (certificate authority, CA) TLS. |
| *Файл TLS ключа базы данных* | Укажите полный путь к корректному файлу ключа TLS. |
| *Файл TLS сертификата базы данных* | Укажите полный путь к корректному файлу сертификата TLS. |
| *Сверка хоста базы данных* | Отметьте этот флажок, чтобы активировать сверку хоста.<br>Отключено для MYSQL, поскольку библиотека PHP MySQL не разрешает пропускать шаг валидации сертификата узла. |
| *Список шифров TLS базы данных* | Укажите пользовательский список допустимых шифров. Формат списка шифров должен отвечать стандарту OpenSSL.<br>Доступно только для MySQL. |

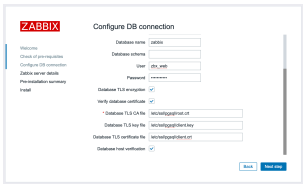
**Attention:**
Параметры TLS должны указывать на корректные файлы. Если они указывают на несуществующие или некорректные файлы, это приведёт к ошибке авторизации.<br> Если файлы сертификатов доступны на запись, веб-интерфейс генерирует предупреждение в отчёте Информация о системе о том, что "TLS certificate files must be read-only" (отображается только если пользователь PHP является владельцем сертификата).
Защищённые паролем сертификаты не поддерживаются.

Сценарии использования

Веб-интерфейс Zabbix использует графический интерфейс, чтобы определить возможные опции: `required`,`verify_ca`, `verify_full`. Указывайте нужные опции на шаге *Настройка соединения с базой данных* помощника по установке. Эти опции соответствуют файлу конфигурации (zabbix.conf.php) следующим образом:

| Параметр интерфейса | Файл конфигурации | Описание | Результат |
|---|---|---|---|
|  | ...<br>// Used for TLS connection.<br>$DB['ENCRYPTION'] = true;<br>$DB['KEY_FILE'] = '';<br>$DB['CERT_FILE'] = '';<br>$DB['CA_FILE'] = '';<br>$DB['VERIFY_HOST'] = false;<br>$DB['CIPHER_LIST'] = '';<br>... | Отметьте *TLS шифрование базы данных*<br>Оставьте *Удостоверять сертификат базы данных* неотмеченным | Включает режим «required». |

| Параметр интерфейса | Файл конфигурации | Описание | Результат |
|---|---|---|---|
|  | ...<br>$DB['ENCRYPTION'] = true;\\ $DB['KEY_FILE'] = '';<br>$DB['CERT_FILE'] = '';<br>$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';<br>$DB['VERIFY_HOST'] = false;<br>$DB['CIPHER_LIST'] = '';<br>... | 1. Отметьте *TLS шифрование базы данных* и *Удостоверять сертификат базы данных*<br>2. Укажите путь к *Файл TLS CA базы данных* | Включает режим «verify_ca». |
| | ...<br>// Used for TLS connection with strictly defined Cipher list.<br>$DB['ENCRYPTION'] = true;<br>$DB['KEY_FILE'] = '<key_file_path>';<br>$DB['CERT_FILE'] = '<key_file_path>';<br>$DB['CA_FILE'] = '<key_file_path>';<br>$DB['VERIFY_HOST'] = true;<br>$DB['CIPHER_LIST'] = '<cipher_list>';<br>...<br><br>Или:<br><br>...<br>// Used for TLS connection without Cipher list defined - selected by MySQL server<br>$DB['ENCRYPTION'] = true;<br>$DB['KEY_FILE'] = '<key_file_path>';<br>$DB['CERT_FILE'] = '<key_file_path>';<br>$DB['CA_FILE'] = '<key_file_path>';<br>$DB['VERIFY_HOST'] = true;<br>$DB['CIPHER_LIST'] = '';<br>... | 1. Отметьте *TLS шифрование базы данных* и *Удостоверять сертификат базы данных*<br>2. Укажите путь к *Файл TLS ключа базы данных*<br>3. Укажите путь к *Файл TLS CA базы данных*<br>4. Укажите путь к *Файл TLS сертификата базы данных*<br>5. Укажите список шифров TLS (опционально) | Включает режим «verify_full» для MySQL. |

| Параметр интерфейса | Файл конфигурации | Описание | Результат |
|---|---|---|---|
|  | ...<br>$DB['ENCRYPTION'] = true;<br>$DB['KEY_FILE'] = '<key_file_path>';<br>$DB['CERT_FILE'] = '<key_file_path>';<br>$DB['CA_FILE'] = '<key_file_path>';<br>$DB['VERIFY_HOST'] = true;<br>$DB['CIPHER_LIST'] = ' ';<br>... | 1. Отметьте *TLS шифрование базы данных* и *Удостоверять сертификат базы данных*<br>2. Укажите путь к *Файл TLS ключа базы данных*<br>3. Укажите путь к *Файл TLS CA базы данных*<br>4. Укажите путь к *Файл TLS сертификата базы данных*<br>5. Отметьте *Сверка хоста базы данных* | Включает режим «verify_full» для PostgreSQL. |

**Смотрите также:** Примеры настроек шифрования для MySQL, Примеры настроек шифрования для PostgreSQL.

Настройка Zabbix сервера/прокси

Безопасные подключения к базе данных могут быть настроены с помощью соответствующих параметров в файле конфигурации Zabbix сервера и/или прокси.

| Настройка | Результат |
|---|---|
| Отсутствует | Подключение к базе данных без шифрования. |
| 1. Установите DBTLSConnect=required | Сервер/прокси делает TLS подключение к базе данных. Нешифрованные подключения не разрешены. |
| 1. Установите DBTLSConnect=verify_ca<br>2. Установите DBTLSCAFile — укажите файл удостоверяющего центра (certificate authority, CA) TLS | Сервер/прокси делает TLS подключение к базе данных после проверки сертификата базы данных. |
| 1. Установите DBTLSConnect=verify_full<br>2. Установите DBTLSCAFile — укажите файл удостоверяющего центра (certificate authority, CA) TLS | Сервер/прокси делает TLS подключение к базе данных после проверки сертификата базы данных и идентичности хоста базы данных. |
| 1. Установите DBTLSCAFile — укажите файл удостоверяющего центра (certificate authority, CA)<br>2. Установите DBTLSCertFile — укажите файл сертификата открытого ключа (public key) клиента<br>3. Установите DBTLSKeyFile — укажите файл закрытого ключа (private key) клиента | Сервер/прокси при подключении к базе данных предъявляет сертификат клиента. |
| 1. Установите DBTLSCipher — список шифров (алгоритмов шифрования), которые клиент разрешает при подключении с использованием протокола TLS до версии TLS 1.2<br><br>или DBTLSCipher13 — список шифров, которые клиент разрешает при подключении с использованием протокола TLS 1.3 | (MySQL) TLS подключение выполняется, используя шифр из представленного списка.<br>(PostgreSQL) Установка этой опции рассматривается как ошибка. |

### 1. Настройка шифрования для MySQL

Обзор

В этом разделе приведены несколько примеров настройки шифрования для CentOS 8.2 и MySQL 8.0.30, которые можно использовать как краткое руководство по шифрованию соединения с базой данных.

**Attention:**
Если для узла MySQL задано значение localhost, параметры шифрования будут недоступны. В этом случае соединение между веб-интерфейсом Zabbix и базой данных использует файл сокета (в Unix) или разделяемую память (в Windows) и не может быть зашифровано.

**Note:**
Список комбинаций шифрования не ограничивается приведенными на этой странице. Доступно гораздо больше комбинаций.

Предварительные требования

Установите базу данных MySQL из официального репозитория.

Подробности о том, как использовать репозиторий MySQL, см. в документации MySQL.

Сервер MySQL готов принимать защищенные подключения с использованием самоподписанного сертификата.

Чтобы увидеть, какие пользователи используют зашифрованное подключение, выполните следующий запрос (Performance Schema должна быть включена):

```
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;
```

Режим Required

Конфигурация MySQL

Современные версии базы данных готовы к работе "из коробки" для режима шифрования `required`. После первоначальной настройки и запуска будет создан серверный сертификат.

Создайте пользователей и роли для основных компонентов:

**Attention:**
Для версий MySQL 8.4+ вместо `mysql_native_password` следует использовать `caching_sha2_password`.

```
mysql> CREATE USER
 'zbx_srv'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>',
 'zbx_web'@'%' IDENTIFIED WITH mysql_native_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'@'%';
```

Обратите внимание, что протокол X.509 не используется для проверки идентичности, но пользователь настроен на использование только зашифрованных соединений. Дополнительные сведения о настройке пользователей см. в документации MySQL.

Выполните проверку соединения (для тестирования защищенных соединений нельзя использовать соединение через сокет):

```
mysql -u zbx_srv -p -h 10.211.55.9 --ssl-mode=REQUIRED
```

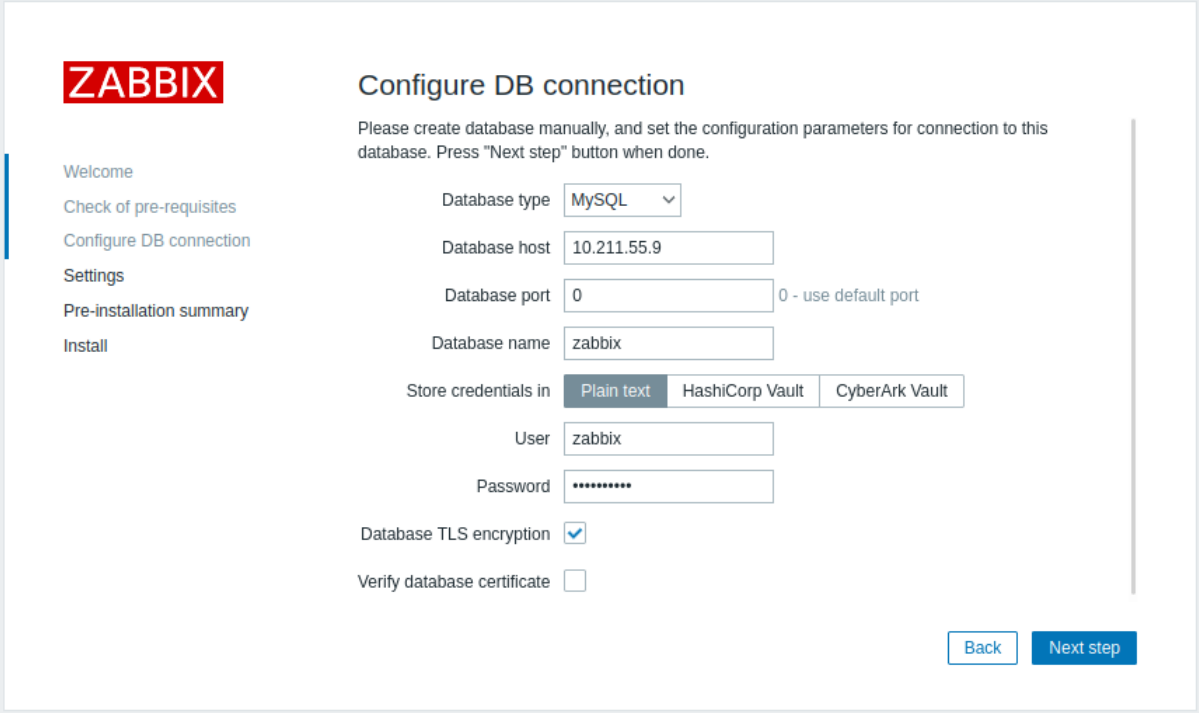
Проверьте текущий статус и доступные наборы шифров:

```
mysql> status
--------------
mysql Ver 8.0.21 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
```

Веб-интерфейс

Чтобы включить шифрование только для передачи данных для соединений между веб-интерфейсом Zabbix и базой данных:

- Установите флажок *Database TLS encryption*
- Оставьте *Verify database certificate* неотмеченным




Сервер

Чтобы включить шифрование только для передачи данных для соединений между сервером и базой данных, настройте */etc/zabbix/zabbix_server.conf*:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Шифрование с проверкой центра сертификации

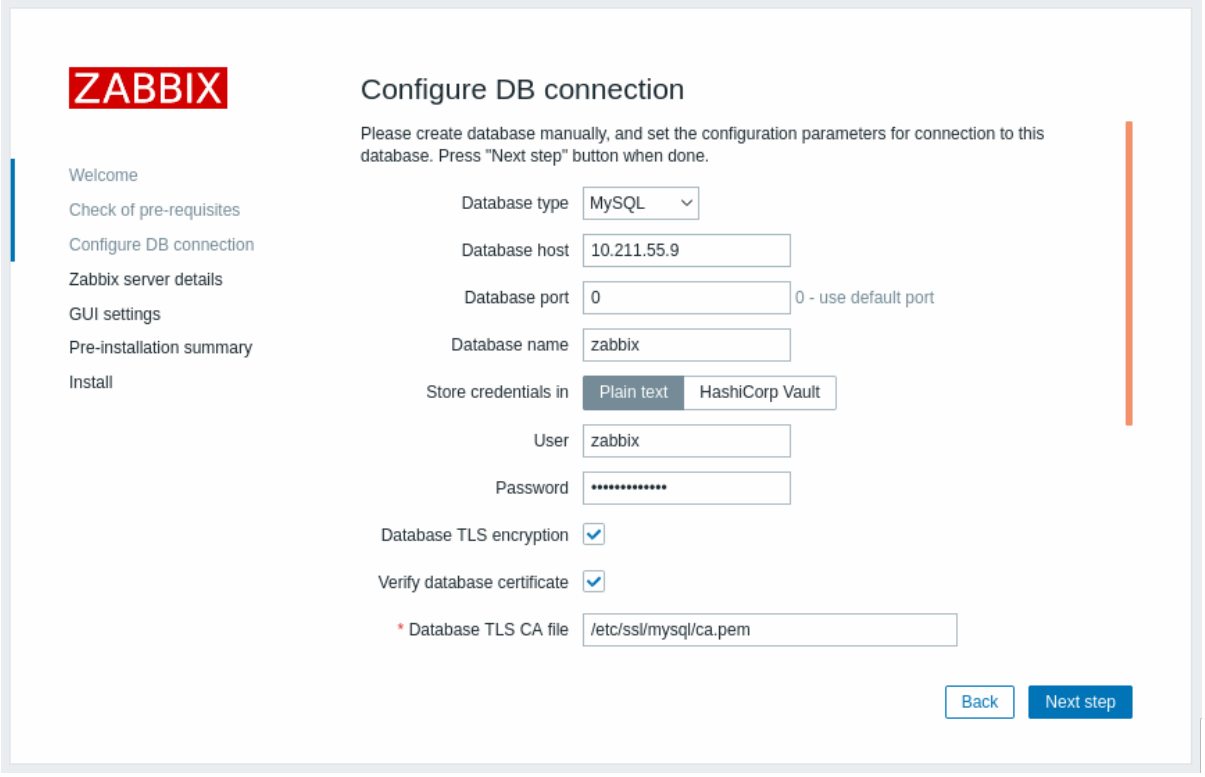
Скопируйте необходимый MySQL CA на сервер веб-интерфейса Zabbix и назначьте соответствующие разрешения, чтобы веб-сервер мог читать этот файл.

**Note:**
Этот режим не работает в RHEL 7 из-за более старых библиотек MySQL.

Веб-интерфейс

Чтобы включить шифрование с проверкой сертификата для соединений между веб-интерфейсом Zabbix и базой данных:

- Отметьте *Database TLS encryption* и *Verify database certificate*
- Укажите путь к файлу Database TLS CA




В качестве альтернативы это можно задать в */etc/zabbix/web/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'] = '';
...
```

Для устранения неполадок используйте командную строку, чтобы проверить, возможно ли подключение для нужного пользователя:

```
    mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=REQUIRED --ssl-ca=/var/lib/mysql/ca.pem
```

Сервер

Чтобы включить шифрование с проверкой сертификата для соединений между сервером Zabbix и базой данных, настройте */etc/zabbix/zabbix_server.conf*:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/mysql/ca.pem
...
```

Режим Verify Full

Конфигурация MySQL

Установите параметр конфигурации сервера MySQL CE (*/etc/my.cnf.d/server-tls.cnf*) в значение:

```
[mysqld]
...
#####                           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
...
```

Ключи для сервера MySQL CE и клиента (веб-интерфейс Zabbix) следует создать вручную в соответствии с документацией MySQL CE: Creating SSL and RSA certificates and keys using MySQL или Creating SSL certificates and keys using openssl

**Attention:**
Сертификат сервера MySQL должен содержать поле Common Name, установленное в значение FQDN, так как веб-интерфейс Zabbix будет использовать DNS-имя для связи с базой данных или IP-адрес узла сети базы данных.

Создайте пользователя MySQL:

**Attention:**
Для версий MySQL 8.4+ вместо `mysql_native_password` следует использовать `caching_sha2_password`.

```
mysql> CREATE USER
  'zbx_srv'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>',
  'zbx_web'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>'
  REQUIRE X509
  PASSWORD HISTORY 5;
```

Проверьте, можно ли войти с этим пользователем:

```
mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=VERIFY_IDENTITY --ssl-ca=/var/lib/mysql/ca.pem --ssl-cert=/v
```

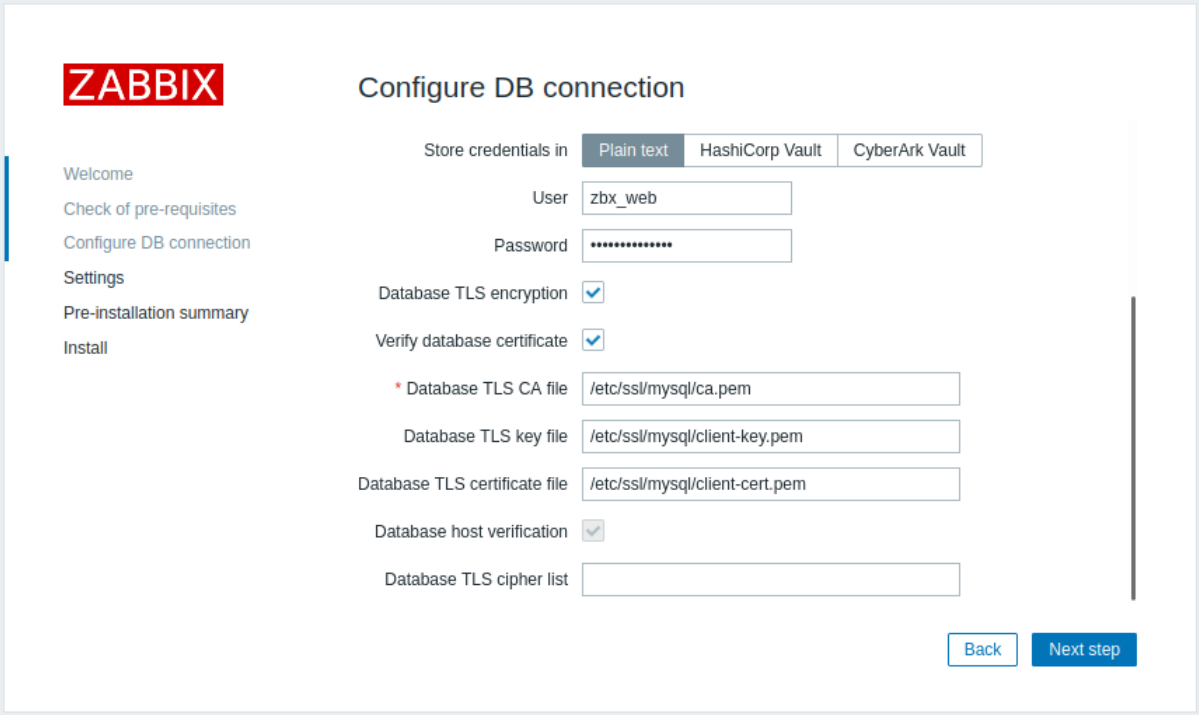
Веб-интерфейс

Чтобы включить шифрование с полной проверкой для соединений между веб-интерфейсом Zabbix и базой данных:

- Установите флажки *Database TLS encryption* и *Verify database certificate*
- Укажите путь к *Database TLS key file*
- Укажите путь к *Database TLS CA file*
- Укажите путь к *Database TLS certificate file*

Обратите внимание, что *Database host verification* отмечен и недоступен для изменения - этот шаг нельзя пропустить для MySQL.

**Warning:**
Если поле *Database TLS cipher list* оставить пустым, будут включены общие наборы шифров, разрешенные и веб-интерфейсом (клиентом), и сервером. В качестве альтернативы наборы шифров можно задать явно в соответствии с требованиями к настройке наборов шифров.




Кроме того, это можно задать в */etc/zabbix/web/zabbix.conf.php*:

```
...
// Used for TLS connection with strictly defined Cipher list.
$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
...
// or
...
// Used for TLS connection without Cipher list defined - selected by MySQL server
$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'] = '';
...
```

Сервер

Чтобы включить шифрование с полной проверкой для соединений между сервером Zabbix и базой данных, настройте */etc/zabbix/zabbix_server.conf*:

```
...
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
...
```

**2. Настройка шифрования для PostgreSQL**

Обзор

В этом разделе предоставлено несколько примеров настроек шифрования для CentOS 8.2 и PostgreSQL 13.

**Note:**
Если значение поля *Хост базы данных* начинается с косой черты или это поле пустое, то соединения между веб-интерфейсом Zabbix и PostgreSQL шифрованными быть не могут (параметры в GUI выключаются).

Предварительные требования

Установите базу данных PostgreSQL, используя официальный репозиторий.

«Из коробки» PostgreSQL не настроен на приём TLS подключений. Пожалуйста, следуйте инструкциям из документации PostgreSQL для подготовки сертификата и postgresql.conf [en], а также для контроля доступа пользователей[en] посредством файла ph_hba.conf.

По умолчанию, сокет PostgreSQL привязан к localhost, для удалённых сетевых подключений разрешите прослушивание на реальном сетевом интерфейсе.

Настройки PostgreSQL для всех режимов могут выглядеть следующим образом:

**/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'
...
```

Для контроля доступа подправьте */var/lib/pgsql/13/data/pg_hba.conf*:

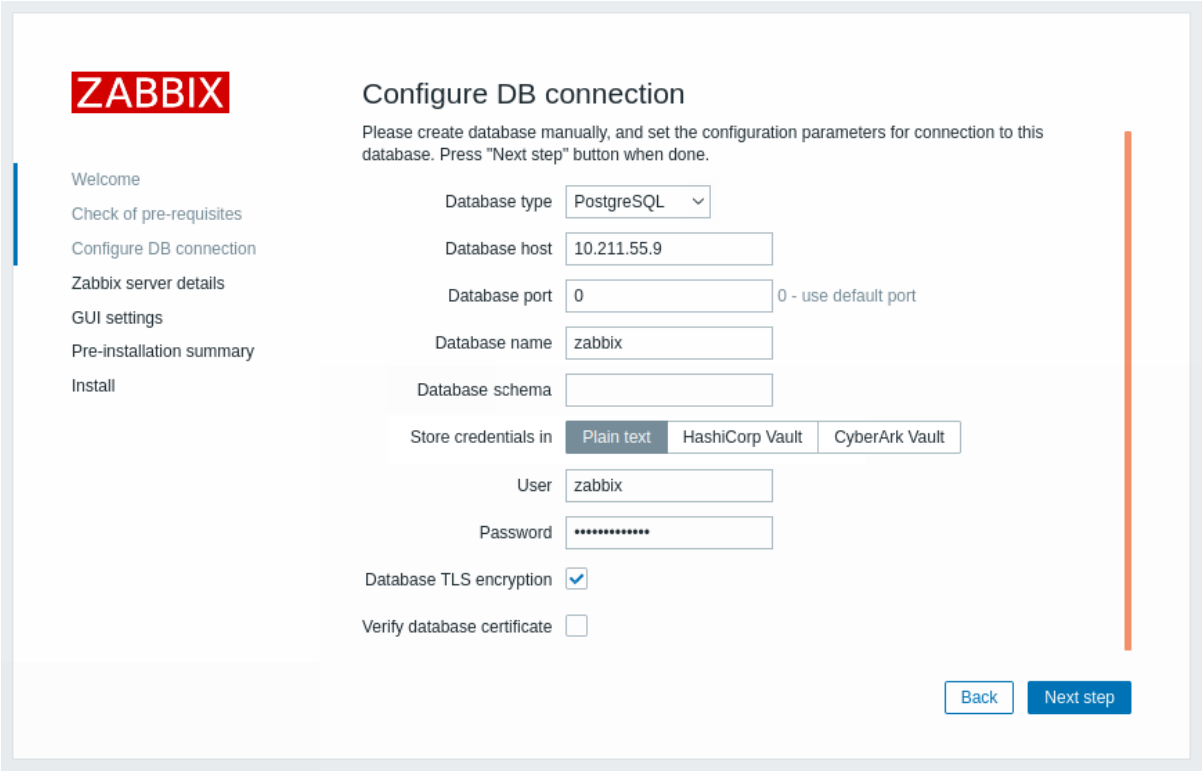
```
...
### require
hostssl all all 0.0.0.0/0 md5

### 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
...
```

Режим Required

Веб-интерфейс

Чтобы включить только шифрование на транспортном уровне для соединений между веб-интерфейсом Zabbix и базой данных:

- Отметьте *TLS шифрование базы данных*
- Оставьте *Удостоверять сертификат базы данных* неотмеченным




Сервер

Чтобы включить только шифрование на транспортном уровне для соединений между сервером Zabbix и базой данных, настройте */etc/zabbix/zabbix_server.conf*:

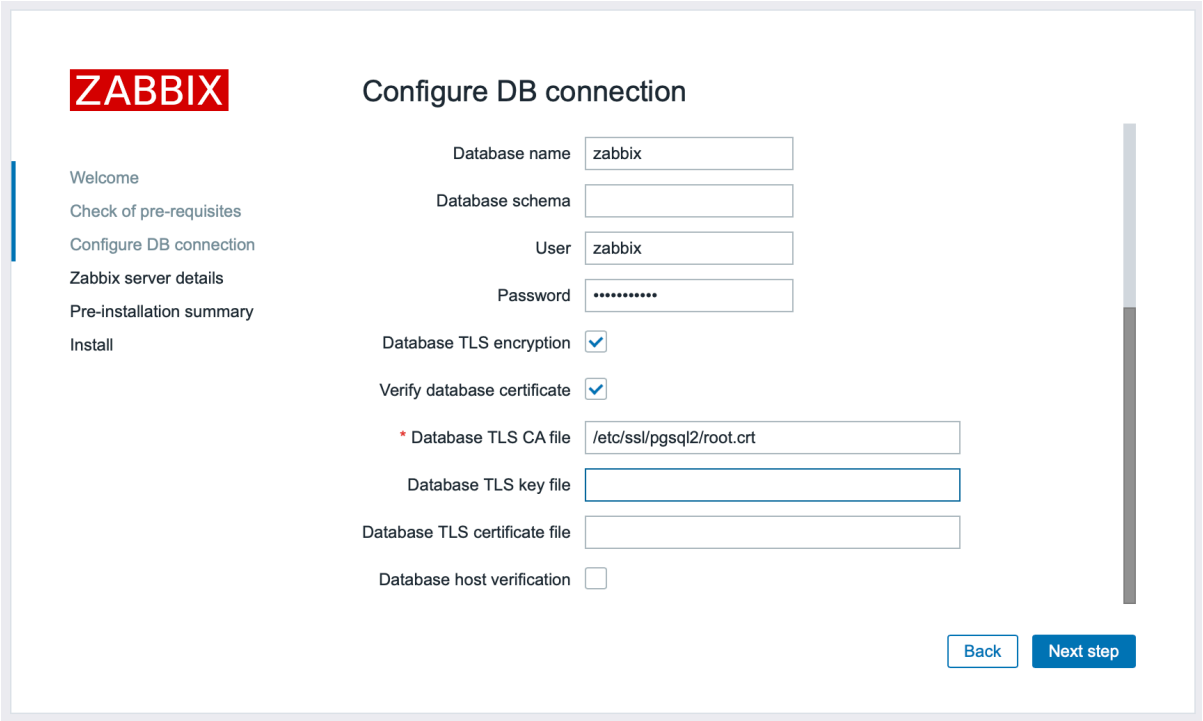
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Режим Verify CA

Веб-интерфейс

Чтобы включить шифрование с проверкой сертификата для соединений между веб-интерфейсом Zabbix и базой данных:

- Отметьте *TLS шифрование базы данных* и *Удостоверять сертификат базы данных*
- Укажите путь в поле *Файл TLS CA базы данных*




В качестве альтернативы, можно выставить в */etc/zabbix/web/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'] = '';
...
```

Сервер

Чтобы включить шифрование с проверкой сертификата для соединений между сервером Zabbix и базой данных, настройте */etc/zabbix/zabbix_server.conf:*

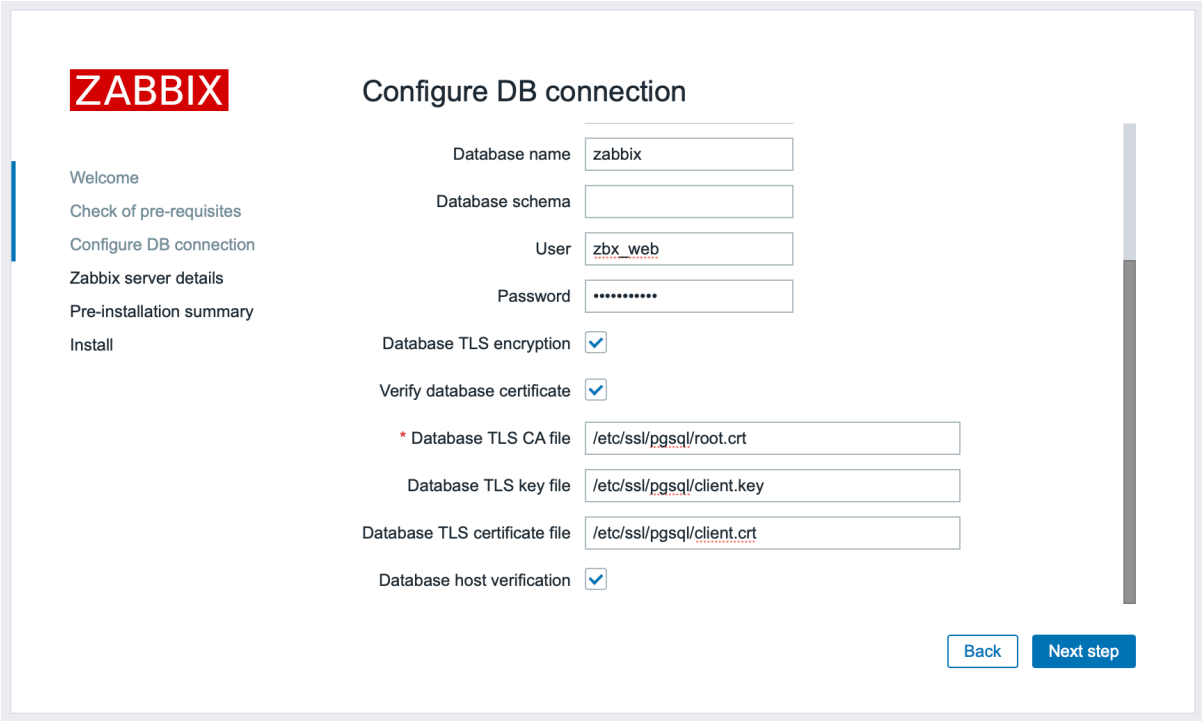
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/pgsql/root.crt
...
```

Режим Verify full

Веб-интерфейс

Чтобы включить шифрование с сертификатом и проверкой идентичности узла базы данных для соединений между веб-интерфейсом Zabbix и базой данных:

- Отметьте *TLS шифрование базы данных* и *Удостоверять сертификат базы данных*
- Укажите путь в поле *Файл TLS ключа базы данных*
- Укажите путь в поле *Файл TLS CA базы данных*
- Укажите путь в поле *Файл TLS сертификата базы данных*
- Отметьте *Сверка хоста базы данных*




Как альтернатива, можно задать в */etc/zabbix/web/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'] = '';
...
```

Сервер

Чтобы включить шифрование с сертификатом и проверкой идентичности узла базы данных для соединений между сервером Zabbix и базой данных, настройте */etc/zabbix/zabbix_server.conf*:

```
...
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
...
```

**6 Безопасное подключение к веб-интерфейсу**

Обзор

В этом разделе приведены шаги по настройке Zabbix и примеры конфигурации для безопасных TLS-соединений между веб-интерфейсом Zabbix и сервером Zabbix.

Конфигурация

По умолчанию связь между веб-интерфейсом Zabbix и сервером Zabbix не зашифрована. Для повышения безопасности включите TLS на обеих сторонах. Ниже приведен пример самого простого способа сделать это.

1. Сгенерируйте сертификаты и ключи.

Создайте рабочий каталог:

```
sudo mkdir -p /etc/zabbix/ssl && cd /etc/zabbix/ssl
```

Создайте сертификат CA (измените значение MyZabbixCA, чтобы оно соответствовало фактическому common name):

```
sudo openssl genrsa -out ca.key 4096
sudo openssl req -new -x509 -days 3650 -key ca.key -out ca.crt -subj "/CN=MyZabbixCA/"
```

Сгенерируйте закрытый ключ и сертификат для сервера Zabbix (измените значение zabbix-server.example.com, чтобы оно соответствовало фактическому common name):

```
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
```

Сгенерируйте закрытый ключ и сертификат для веб-интерфейса Zabbix (измените значение zabbix-frontend-node, чтобы оно соответствовало фактическому common name):

```
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. Установите правильные права доступа.

Для сервера Zabbix (при необходимости измените владельца/группу в соответствии с пользователем демона сервера Zabbix в вашем дистрибутиве):

```
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
```

Для веб-интерфейса (при необходимости измените владельца/группу в соответствии с пользователем веб-сервера в вашем дистрибутиве):

```
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. Настройте сервер Zabbix.

В zabbix_server.conf добавьте:

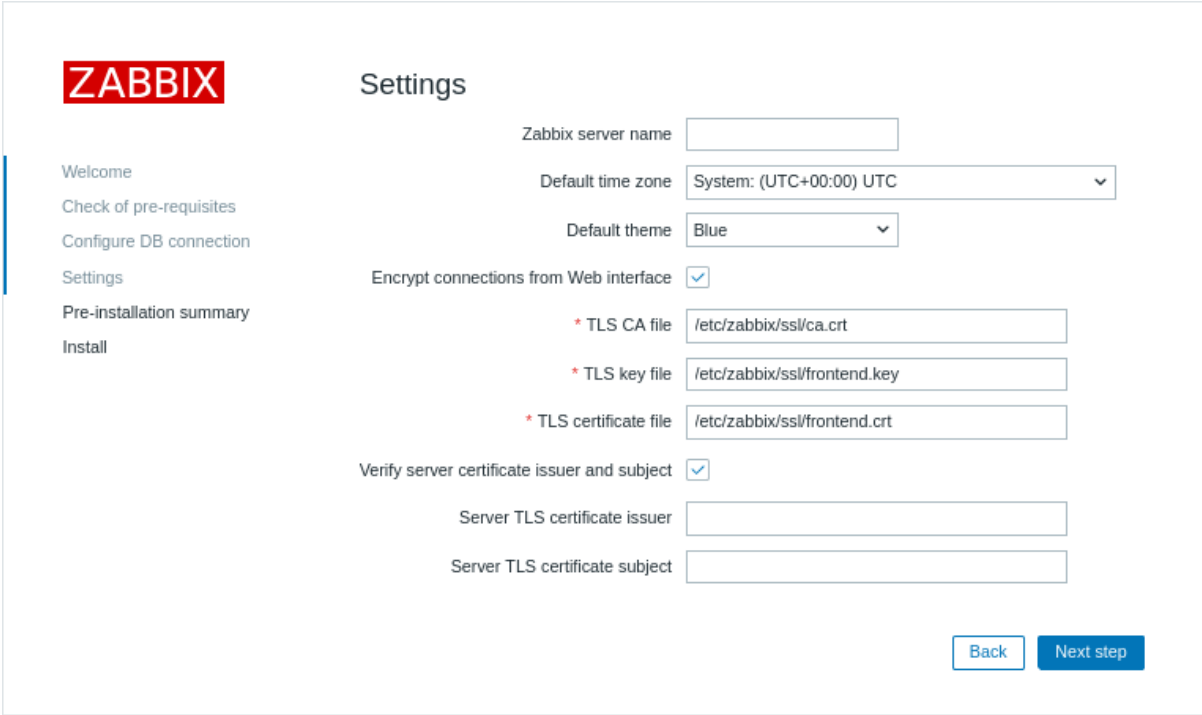
```
TLSFrontendAccept=cert
TLSCertFile=/etc/zabbix/ssl/server.crt
TLSKeyFile=/etc/zabbix/ssl/server.key
TLSCAFile=/etc/zabbix/ssl/ca.crt
####          :
#### TLSFrontendCertIssuer=/CN=MyZabbixCA/
#### TLSFrontendCertSubject=/CN=zabbix-frontend-node/
```

Затем перезапустите сервер:

```
sudo systemctl restart zabbix-server
```

4. Настройте веб-интерфейс Zabbix.

Во время установки веб-интерфейса включите параметр *Encrypt connections from Web interface* (а также *Verify server certificate issuer and subject*, если необходимо) и заполните поля *TLS CA file*, *TLS key file*, *TLS certificate file* (а также поля *Server TLS certificate issuer* и *Server TLS certificate subject*, если необходимо):




| Parameter | Description |
|---|---|
| *TLS CA file* | Укажите полный путь к файлу сертификата центра сертификации (CA), используемому для проверки сертификата сервера. |
| *TLS key file* | Укажите полный путь к файлу закрытого ключа клиента, соответствующего сертификату клиента. |
| *TLS certificate file* | Укажите полный путь к файлу сертификата клиента, если требуется взаимная аутентификация TLS. |
| *Server TLS certificate issuer* | Укажите Distinguished Name (DN) издателя, который должен совпадать с сертификатом сервера. |
| *Server TLS certificate subject* | Укажите Distinguished Name (DN) субъекта, который должен совпадать с сертификатом сервера. |

В существующих установках отредактируйте следующие поля в `zabbix.conf.php`:

```
$ZBX_SERVER_TLS['ACTIVE'] = '1';
$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';
//          :
// $ZBX_SERVER_TLS['CERTIFICATE_ISSUER']  = '/CN=MyZabbixCA/';
// $ZBX_SERVER_TLS['CERTIFICATE_SUBJECT'] = '/CN=zabbix-server.example.com/';
```

5. Проверьте шифрование, убедившись, что в журнале веб-интерфейса Zabbix или в файле журнала сервера Zabbix нет сообщений об ошибках:

```
tail -f /var/log/zabbix/zabbix_server.log
```

#### 7 Настройка TimescaleDB

Обзор

Для обеспечения большего уровня производительности при масштабировании, Zabbix поддерживает TimescaleDB — решение на основе базы данных PostgreSQL для автоматического партиционирования данных на фрагменты (chunks) на основе времени.

**Warning:**
В настоящее время TimescaleDB не поддерживается Zabbix прокси-сервером.

Инструкции на этой странице могут быть использованы для следующих сценариев:

- Создание базы данных TimescaleDB или для миграции с существующих таблиц PostgreSQL на TimescaleDB (смотрите Настройка).
- Обновление существующей схемы базы данных TimescaleDB при обновлении Zabbix (смотрите Обновление схемы TimescaleDB).

Конфигурация

**Предварительные требования**: расширение TimescaleDB поддерживаемой версии, установленное на сервере базы данных поддерживаемой версии. Инструкции по установке см. в документации TimescaleDB.

> **Warning:**
> Перед установкой TimescaleDB установите поддерживаемый выпуск PostgreSQL из официального репозитория PostgreSQL.

Включите расширение TimescaleDB для конкретной базы данных, выполнив:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb CASCADE;" | sudo -u postgres psql zabbix
```

Для выполнения этой команды требуются права администратора базы данных.

> **Note:**
> Если вы используете схему базы данных, отличную от 'public', необходимо добавить в приведенную выше команду предложение SCHEMA. Например:<br> `echo "CREATE EXTENSION IF NOT EXISTS timescaledb SCHEMA yourschema CASCADE;" | sudo -u postgres psql zabbix`

Затем выполните скрипт `postgresql/timescaledb/schema.sql`. Для новых установок скрипт необходимо запускать после создания обычной базы данных PostgreSQL с начальной схемой/данными (см. создание базы данных).

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/schema.sql | sudo -u zabbix psql zabbix
```

> **Attention:**
> Пожалуйста, игнорируйте предупреждающие сообщения о том, что при выполнении скрипта `schema.sql` в TimescaleDB версии 2.9.0 и выше не соблюдаются рекомендации по лучшим практикам. Несмотря на это предупреждение, настройка будет успешно завершена.

Миграция существующих данных истории, трендов и журнала аудита может занять много времени. На время миграции сервер и веб-интерфейс Zabbix должны быть остановлены.

Скрипт `schema.sql` задает следующие параметры обслуживания:

- Переопределить период хранения истории элементов данных
- Переопределить период хранения трендов элементов данных

Чтобы использовать разбиение обслуживания на разделы для истории и трендов, необходимо включить оба этих параметра. Также можно включить переопределение отдельно только для истории или только для трендов.

Для PostgreSQL и TimescaleDB скрипт `postgresql/timescaledb/schema.sql` задает два дополнительных параметра:

- Включить сжатие
- Сжимать записи старше 7 дней

Для успешного удаления сжатых данных процессом очистки истории, обе опции *Переопределить период хранения истории элементов данных (Override item history period)* и *Переопределить период хранения динамики изменения элементов данных (Override item trend period)* должны быть активированы. Если переопределение отключено, а таблицы имеют сжатые фрагменты, процесс очистки истории не удалит данные из этих таблиц, а в разделах *Очистка истории (Housekeeping)* и *Информация о системе (System information)* будет отображено предупреждение о некорректной конфигурации.

Все эти параметры можно изменить в разделе *Администрирование* → *Очистка истории* (*Administration* → *Housekeeping*) после установки.

> **Note:**
> Возможно, вы захотите запустить инструмент timescaledb-tune, предоставляемый TimescaleDB, чтобы оптимизировать параметры конфигурации PostgreSQL в вашем файле `postgresql.conf`.

Обновление схемы TimescaleDB

При обновлении Zabbix до версии, содержащей новые гипертаблицы TimescaleDB, сервер Zabbix не настраивает эти гипертаблицы автоматически (например, при обновлении с Zabbix 6.4 до 7.4, так как в версиях 7.0.0 и 7.0.2 были добавлены новые гипертаблицы).

Чтобы настроить новые гипертаблицы TimescaleDB, выполните следующие шаги:

1. Запустите сервер Zabbix; это обновит существующую базу данных.
2. Проверьте в файле журнала сервера, что обновление базы данных завершено; после завершения остановите сервер Zabbix. Обратите внимание, что сервер записывает предупреждение, если пытается включить сжатие для таблицы, которая не является гипертаблицей.
3. Выполните скрипт `postgresql/timescaledb/schema.sql`; он настроит новые гипертаблицы TimescaleDB. Обратите внимание, что начиная с Zabbix 7.0.0 расположение и имя скрипта изменились с `postgresql/timescaledb.sql` на `postgresql/timescaledb/schema.sql`.

**Attention:**
Пожалуйста, игнорируйте предупреждающие сообщения о том, что при выполнении скрипта `schema.sql` в TimescaleDB версии 2.9.0 и выше не соблюдаются рекомендации по лучшим практикам. Несмотря на это предупреждение, настройка будет успешно завершена.

Сжатие TimescaleDB

Собственное сжатие TimescaleDB поддерживается для всех таблиц Zabbix, которые являются гипертаблицами TimescaleDB. Во время обновления или миграции на TimescaleDB первоначальное сжатие больших таблиц может занять много времени.

Обратите внимание, что сжатие поддерживается под лицензией «timescale» Timescale Community license и не поддерживается под лицензией «apache» Apache 2.0 license. Если сжатие не поддерживается, в журнал сервера Zabbix выводится предупреждающее сообщение, а пользователи не могут активировать сжатие в веб-интерфейсе.

**Note:**
Перед использованием сжатия пользователям рекомендуется ознакомиться со сжатием в документации TimescaleDB [en].

Обратите внимание, что существуют определённые ограничения, налагаемые сжатием, а именно:

- Изменения сжатых фрагментов (вставки, удаления, обновления) не допускаются.
- Изменения схемы для сжатых таблиц не допускаются.

Настройки сжатия можно изменить в блоке *Сжатие истории, тенденций и журнала аудита* в разделе *Администрирование* → *Очистка истории* веб-интерфейса Zabbix.

| Параметр | По умолчанию | Комментарии |
|---|---|---|
| *Включить сжатие (Enable compression)* | Включено | Установка или снятие флажка не активирует/деактивирует сжатие немедленно. Так как сжатием занимается процесс очистки истории (housekeeper), изменения вступят в силу в течение максимум двух периодов времени, заданных в параметре `HousekeepingFrequency` (в файле конфигурации сервера zabbix_server.conf).<br><br>После отключения сжатия новые фрагменты, попавшие в период сжатия, не будут сжиматься. Однако все ранее сжатые данные останутся сжатыми. Чтобы распаковать ранее сжатые фрагменты, следуйте инструкциям в документации TimescaleDB [en].<br><br>При обновлении с более старых версий Zabbix с поддержкой TimescaleDB сжатие не будет включено по умолчанию. |
| *Compress records older than (Сжимать записи старше, чем)* | 7d | Этот параметр не может быть меньше 7 дней.<br><br>Из-за неизменности сжатых фрагментов все запоздавшие данные (например, данные, задержанные прокси), которые старше этого значения, будут отброшены. |

Для повышения производительности обновления динамики изменений вы можете уменьшить параметр «chunk_time_interval»

для таблиц `trends` и `trends_uint` с 30 дней до 7 дней или меньше в зависимости от того, сколько элементов данных используют динамику изменений. Цель этой настройки — придерживаться лучших практик TimescaleDB и гарантировать, что размер фрагмента останется в пределах доступных системных ресурсов.

#### 8 Настройка Elasticsearch

> **Attention:**
> Поддержка Elasticsearch является экспериментальной!

Zabbix поддерживает хранение исторических данных с помощью Elasticsearch вместо базы данных. Пользователи могут выбирать место хранения исторических данных между совместимой базой данных и Elasticsearch. Процедура настройки, описанная в этом разделе, применима к Elasticsearch версии 7.X. Если используется более ранняя или более поздняя версия Elasticsearch, некоторые функции могут работать не так, как ожидается.

> **Warning:**
> 1.Если все исторические данные хранятся в Elasticsearch, тренды **не** вычисляются и не сохраняются в базе данных. Если тренды не вычисляются и не сохраняются, может потребоваться увеличить период хранения истории.<br>
> 2. При использовании Elasticsearch диапазонные запросы, извлекающие значения из базы данных, ограничены временной меткой периода хранения данных.

Настройка

Для обеспечения надлежащей связи между всеми задействованными элементами убедитесь, что параметры файла конфигурации сервера и файла конфигурации веб-интерфейса настроены должным образом.

Zabbix сервер и веб-интерфейс

Выдержка из файла конфигурации Zabbix сервера с параметрами, которые необходимо обновить:

```
### Option: HistoryStorageURL
# History storage HTTP[S] URL.
#
# Mandatory: no
# Default:
# HistoryStorageURL=
### Option: HistoryStorageTypes
# Comma separated list of value types to be sent to the history storage.
#
# Mandatory: no
# Default:
# HistoryStorageTypes=uint,dbl,str,log,text
```

Пример значений параметров, которые требуется указать в файле конфигурации Zabbix сервера:

```
HistoryStorageURL=http://test.elasticsearch.lan:9200
HistoryStorageTypes=str,log,text
```

Подобная конфигурация заставит Zabbix сервер хранить значения истории числовых типов в соответствующей базе данных, а текстовых данных истории в Elasticsearch.

Elasticsearch поддерживает следующие типы элементов данных:

`uint,dbl,str,log,text`

Разъяснение поддерживаемых типов элементов данных:

| Тип значения элемента данных | Таблица в базе данных | Тип в Elasticsearch |
|---|---|---|
| Числовой (целое положительное) | history_uint | uint |
| Числовой (с плавающей точкой) | history | dbl |
| Символ | history_str | str |
| Журнал (лог) | history_log | log |
| Текст | history_text | text |

Выдержка из файла конфигурации веб-интерфейса Zabbix (`conf/zabbix.conf.php`) с параметрами, которые необходимо обновить:

```
// Elasticsearch url (can be string if same url is used for all types).
$HISTORY['url']   = [
      'uint' => 'http://localhost:9200',
      'text' => 'http://localhost:9200'
];
// Value types stored in Elasticsearch.
$HISTORY['types'] = ['uint', 'text'];
```

Пример значений параметров, которые требуется указать в файле конфигурации Zabbix веб-интерфейса:

```
$HISTORY['url']   = 'http://test.elasticsearch.lan:9200';
$HISTORY['types'] = ['str', 'text', 'log'];
```

Подобная конфигурация заставит хранить значения истории , и ( ) в Elasticsearch.

Также требуется сделать $HISTORY глобальной переменной в `conf/zabbix.conf.php`, чтобы убедиться, что всё работает должным образом (смотрите файл `conf/zabbix.conf.php.example`, как это сделать):

```
// Zabbix GUI configuration file.
global $DB, $HISTORY;
```

Установка Elasticsearch и создание сопоставления

Последними двумя шагами, чтобы все заработало, являются установка самого Elasticsearch и создание процесса сопоставления.

Для установки Elasticsearch, пожалуйста, обратитесь к инструкции по установке Elasticsearch [en].

**Note:**
Сопоставление является структурой данных в Elasticsearch (аналогично таблице в базе данных). Сопоставления по всем типам данных истории доступны здесь: `database/elasticsearch/elasticsearch.map`.

**Warning:**
Создание сопоставления обязательно. Некоторый функционал будет поломан, если сопоставление не создано согласно этой инструкции.

Для создания сопоставления для типа `text` отправьте следующий запрос в Elasticsearch:

```
curl -X PUT \
 http://your-elasticsearch.here: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"
         }
      }
   }
}'
```

Похожий запрос требуется выполнить для создания сопоставлений по значениям истории и ( ) с соответствующим исправлением типа.

> **Note:**
> Чтобы работать с Elasticsearch, пожалуйста, обратитесь к странице требований для получения более подробной информации.

> **Note:**
> Очистка истории не удаляет никакие данные из Elasticsearch.

Запись данных истории в несколько индексов на основе даты

В этом разделе описываются дополнительные шаги, которые необходимы для работы с конвейерами и узлами ingest.

Для начала вам нужно создать шаблоны для индексов.

Следующий пример показывает запрос на создание шаблона uint:

```
curl -X PUT \
 http://your-elasticsearch.here:9200/_template/uint_template \
 -H 'content-type:application/json' \
 -d '{
   "index_patterns": [
      "uint*"
   ],
   "settings": {
      "index": {
         "number_of_replicas": 1,
         "number_of_shards": 5
      }
   },
   "mappings": {
      "properties": {
         "itemid": {
            "type": "long"
         },
         "clock": {
            "format": "epoch_second",
            "type": "date"
         },
         "value": {
            "type": "long"
         }
      }
   }
}'
```

Для создания остальных шаблонов пользователю нужно изменить URL (последняя часть является именем шаблона), изменить поле "index_patterns", чтобы совпадало с именем индекса, и указать корректное сопоставление, которое можно взять из database/elasticsearch/elasticsearch.map.

Например, можно использовать следующую команду, чтобы создать шаблон для текстового индекса:

```
curl -X PUT \
 http://your-elasticsearch.here: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"
         }
      }
   }
}'
```

Этот шаг необходим, чтобы позволить Elasticsearch задавать корректные сопоставления для индексов, которые создаются автоматически. Затем это требуется для создания определения конвейера. Конвейер является своего рода предварительной обработкой данных перед тем как поместить их в индексы. Можно использовать следующую команду, чтобы создать конвейер для индекса uint:

```
curl -X PUT \
 http://your-elasticsearch.here:9200/_ingest/pipeline/uint-pipeline \
 -H 'content-type:application/json' \
 -d '{
   "description": "daily uint index naming",
   "processors": [
      {
         "date_index_name": {
            "field": "clock",
            "date_formats": [
               "UNIX"
            ],
            "index_name_prefix": "uint-",
            "date_rounding": "d"
         }
      }
   ]
}'
```

Пользователь может изменить параметр округления (”date_rounding”), чтобы задать спефицичный период ротации индекса. Для создания остальных конвейеров пользователю необходимо изменить URL (последняя часть является именем конвейера) и изменить поле ”index_name_prefix”, чтобы оно соответствовало имени индекса.

Смотрите также документацию Elasticsearch [en].

Кроме того, запись данных истории в несколько индексов на основе даты необходимо также включить в новом параметре в конфигурации Zabbix сервера:

```
### Option: HistoryStorageDateIndex
```

```
# Enable preprocessing of history values in history storage to store values in different indices based on
# 0 - disable
# 1 - enable
#
# Mandatory: no
# Default:
# HistoryStorageDateIndex=0
```

Решение проблем

Следующие шаги могут помочь вам в решении проблем с настройкой Elasticsearch:

1. Проверьте, что сопоставления корректны (запрос GET к URL нужного индекса, наподобие: `http://localhost:9200/uint`).
2. Проверьте, что шарды не находятся в состоянии ошибки (может помочь перезапуск Elasticsearch).
3. Проверьте настройки Elasticsearch. Конфигурация должна разрешать доступ с хоста веб-интерфейса Zabbix и хоста сервера Zabbix.
4. Проверьте журналы Elasticsearch.
5. Для проверки медленных запросов в базе данных Elasticsearch можно использовать `LogSlowQueries`.

Если у вас по-прежнему возникают проблемы с вашей инсталляцией, пожалуйста, создайте отчёт об ошибке со всей информацией из этого списка (сопоставления, журналы ошибок, настройка, версия и так далее).

#### 9 Примечания по настройке Nginx для Zabbix в зависимости от дистрибутива

RHEL

В системах RHEL 8, 9 и 10 Nginx включён в AppStream — дополнительных репозиториев не требуется. Просто подключите и установите командами:

```
dnf module list nginx
dnf -y install nginx
```

В RHEL 7 Nginx может быть установлен из репозитория CentOS 7 Extras (если используется CentOS) командой:

```
yum -y install nginx
```

Если вы предпочитаете самые последние сборки, настройте официальный репозиторий NGINX в файле `/etc/yum.repos.d/nginx.r`

```
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

В SUSE Linux Enterprise Server 15 необходимо настроить `php-fpm` (путь к файлу конфигурации может немного отличаться в зависимости от сервис-пака):

```
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
```

#### 10 Запуск агента от имени root

Начиная с Zabbix **5.0.0**, файл службы systemd для агента Zabbix в официальных пакетах явно содержит директивы `User` и `Group`. Обе установлены в `zabbix`.

Теперь невозможно настроить, от имени какого пользователя запускается агент Zabbix, через файл `zabbix_agentd.conf`, поскольку агент будет игнорировать эту настройку и запускаться от имени пользователя, указанного в файле службы systemd. Чтобы запускать агент Zabbix от имени root, необходимо внести изменения, описанные ниже.

Zabbix агент

Чтобы переопределить пользователя и группу по умолчанию для Zabbix агента, выполните:

```
systemctl edit zabbix-agent
```

Затем добавьте следующее содержимое:

```
[Service]
User=root
Group=root
```

Перезагрузите конфигурацию демонов и перезапустите службу zabbix-agent:

```
systemctl daemon-reload
systemctl restart zabbix-agent
```

Для **Zabbix агента** это возвращает функционал, позволяющий настраивать пользователя в файле `zabbix_agentd.conf`. Теперь Вам нужно выставить параметры конфигурации `User=root` и `AllowRoot=1` в файле конфигурации агента.

Zabbix агент 2

Чтобы переопределить пользователя и группу по умолчанию для Zabbix агента 2, выполните:

```
systemctl edit zabbix-agent2
```

Затем добавьте следующее содержимое:

```
[Service]
User=root
Group=root
```

Перезагрузите конфигурацию демонов и перезапустите службу zabbix-agent2:

```
systemctl daemon-reload
systemctl restart zabbix-agent2
```

Для **Zabbix агента 2** это полностью определяет пользователя, от имени которого он работает. Никаких дополнительных модификаций не требуется.

### 11 Агент Zabbix в Microsoft Windows

Настройка агента

Оба поколения агентов Zabbix работают в качестве службы Windows (Windows service). Для Zabbix агента 2, замените *agentd* на *agent2* в инструкциях ниже.

Вы можете запустить один экземпляр Zabbix агента или несколько экземпляров агента на хосте Microsoft Windows. Одиночный экземпляр может использовать одно из:

- файл конфигурации по умолчанию, расположенный в той же папке, что и бинарный файл агента;
- файл конфигурации, указанный в командной строке.

В случае нескольких экземпляров каждый экземпляр агента должен иметь свой собственный файл конфигурации (один из экземпляров может использовать файл конфигурации по умолчанию).

Пример файла конфигурации доступен в архиве с исходными кодами Zabbix в виде файла:

- `conf/zabbix_agentd.win.conf` для Zabbix агента;
- `conf/zabbix_agent2.conf` для Zabbix агента2.

Если вы хотите установить Zabbix агента/агента 2 для Windows в качестве службы из архива без явного указания файла конфигурации, то перед установкой агента:

- `conf/zabbix_agentd.conf` следует скопировать вручную в папку, куда будет установлен zabbix_agentd.exe;
- `conf/zabbix_agent2.conf` и папку `conf/zabbix_agent2.d` следует скопировать вручную в папку, куда будет установлен zabbix_agent2.exe.

Смотрите опции файла конфигурации для получения подробных сведений по настройке Zabbix Windows агента.

Параметр Hostname

Для выполнения активных проверок на узле сети Zabbix агенту требуется, чтобы было задано имя хоста. Кроме того, указанное значение имени хоста на стороне агента должно в точности совпадать с «именем узла сети», указанном для этого узла сети в веб-интерфейсе.

Значение имени хоста на стороне агента можно задать параметрами **Hostname** или **HostnameItem** в файле конфигурации агента — или будут использованы значения по умолчанию, если ни один из этих параметров не указан.

Значением по умолчанию для параметра **HostnameItem** является значение, возвращаемое ключом агента «system.hostname». Для платформы Windows он вернёт результат функции *gethostname()*, которая опрашивает провайдеров пространств имён, чтобы определить имя локального хоста. Если никакой из провайдеров пространств имён не отвечает, возвращается NetBIOS имя хоста.

Значением по умолчанию для параметра **Hostname** является значение, возвращаемое параметром **HostnameItem**. Таким образом, фактически, если оба этих параметра не заданы, то реальным именем узла сети будет NetBIOS имя хоста; Zabbix агент будет использовать NetBIOS имя хоста для получения списка активных проверок от Zabbix сервера и отправки ему результатов.

Ключ «system.hostname» поддерживает два необязательных параметра — *тип* и *преобразование*.

Параметр *тип* определяет тип имени, которое должен вернуть элемент данных:

- *netbios* (по умолчанию) — возвращает NetBIOS имя хоста, ограниченное до 15 символов и только В ВЕРХНЕМ РЕГИСТРЕ;
- *host* — чувствительное к регистру, возвращает полное, реальное имя хоста Windows (без домена);
- *shorthost* — возвращает часть имени хоста до первой точки. Возвращается полная строка, если имя не содержит точку.
- *fqdn* — возвращает полное доменное имя (FQDN, без конечной точки).

Параметр *преобразование* позволяет указать правило дополнительного преобразования для имени хоста:

- *none* (по умолчанию) — использовать оригинальный регистр букв;
- *lower* — приводить текст к нижнему регистру.

Таким образом, чтобы упростить настройку файла zabbix_agentd.conf и сделать его унифицированным, можно использовать два различных подхода:

1. Оставить параметры **Hostname** или **HostnameItem** не заданными, и тогда Zabbix агент будет использовать NetBIOS имя хоста как имя узла сети;
2. Оставить параметр **Hostname** не заданным и задать **HostnameItem** таким образом:<br> **HostnameItem=system.hostname[** — тогда Zabbix агент будет использовать полное, реальное (регистрозависимое) имя хоста Windows как имя узла сети<br> **HostnameItem=system.hostname[shorthost,lower]** — тогда Zabbix агент будет использовать только часть имени хоста до первой точки, приведённую к нижнему регистру.<br> **HostnameItem=system.hostname[fqdn]** — тогда Zabbix агент будет использовать полное доменное имя в качестве имени узла сети.

Также при установке, запуске, остановке и удалении службы Windows имя хоста используется как часть имени службы Windows. Например, если в файле конфигурации Zabbix агента указано `Hostname=Windows_db_server`, тогда агент будет установлен как служба Windows «`Zabbix Agent [Windows_db_server]`». Поэтому, чтобы у каждого экземпляра Zabbix агента имелись разные имена служб Windows, каждый экземпляр должен использовать отличное от другого экземпляра имя узла сети.

Установка агента как службы Windows

Перед установкой агента вручную скопируйте `conf/zabbix_agentd.conf` в папку, куда будет установлен zabbix_agentd.exe.

Для установки одиночного экземпляра Zabbix агента с файлом конфигурации по умолчанию:

```
zabbix_agentd.exe --install
```

**Attention:**
На 64-битных системах требуется 64-битная версия Zabbix агента, чтобы все проверки, связанные с запущенными 64-битными процессами, работали корректно.

Если вы хотите использовать файл конфигурации, отличный от файла по умолчанию, вы должны использовать следующую команду для установки службы:

```
zabbix_agentd.exe --config <  _  _      > --install
```

Необходимо указать полный путь к файлу конфигурации.

Можно установить несколько экземпляров Zabbix агента как службы, примерно следующим образом:

```
zabbix_agentd.exe --config <  _      _  _     _1> --install --multiple-agents
zabbix_agentd.exe --config <  _      _  _     _2> --install --multiple-agents
...
zabbix_agentd.exe --config <  _      _  _     _N> --install --multiple-agents
```

Установленная служба теперь будет видна в Панели Управления.

Запуск агента

Для запуска службы агента, вы можете использовать Панель Управления или сделать это из командной строки.

Для запуска одиночного экземпляра Zabbix агента с файлом конфигурации по умолчанию:

```
zabbix_agentd.exe --start
```

Для запуска одиночного экземпляра Zabbix агента с отличным от умолчания файлом конфигурации:

```
zabbix_agentd.exe --config <  _  _       > --start
```

Для запуска одного из нескольких экземпляров Zabbix агента:

```
zabbix_agentd.exe --config <  _       _  _     > --start --multiple-agents
```

Остановка агента

Для остановки службы агента, вы можете использовать Панель Управления или сделать это из командной строки.

Для остановки одиночного экземпляра Zabbix агента, который запущен с файлом конфигурации по умолчанию:

```
zabbix_agentd.exe --stop
```

Для остановки одиночного экземпляра Zabbix агента, который запущен с другим файлом конфигурации:

```
zabbix_agentd.exe --config <  _  _       > --stop
```

Для остановки одного из нескольких экземпляров Zabbix агента:

```
zabbix_agentd.exe --config <  _       _  _     > --stop --multiple-agents
```

Удаление службы Windows агента

Для удаления одиночного экземпляра Zabbix агента, который использует файл конфигурации по умолчанию:

```
   zabbix_agentd.exe --uninstall
```

Для удаления одиночного экземпляра Zabbix агента, который использует отличный от умолчания файл конфигурации:

```
   zabbix_agentd.exe --config <  _  _       > --uninstall
```

Для удаления нескольких экземпляров Zabbix агента из служб Windows:

```
  zabbix_agentd.exe --config <  _       _     _1> --uninstall --multiple-agents
  zabbix_agentd.exe --config <  _       _     _2> --uninstall --multiple-agents
  ...
  zabbix_agentd.exe --config <  _       _     _N> --uninstall --multiple-agents
```

Ограничения

Zabbix агент для Windows не поддерживает нестандартные конфигурации Windows, где CPU распределены неравномерно по узлам NUMA. Если логические CPU распределены неравномерно, то метрики производительности CPU могут быть недоступными для некоторых CPU. Например, если имеется 72 логических CPU с 2 нодами NUMA, обе ноды должны иметь по 36 CPU каждая.

**12 Настройка SAML с Microsoft Entra ID**

Обзор

В этом разделе приведены рекомендации по настройке единого входа и инициализации пользователей в Zabbix из Microsoft Entra ID (ранее Microsoft Azure Active Directory) с использованием аутентификации SAML 2.0.

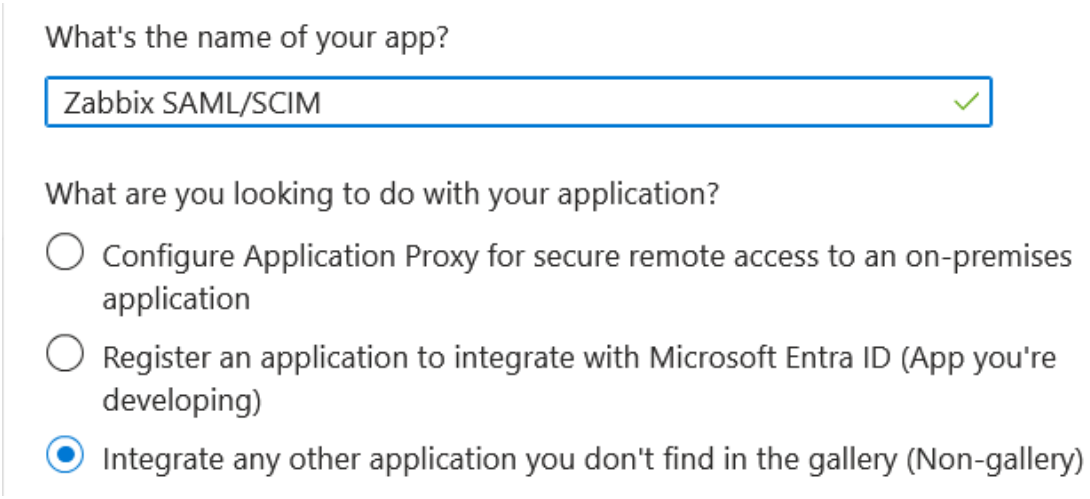
Настройка Microsoft Entra ID

Создание приложения

1. Войдите в центр администрирования Microsoft Entra на странице Microsoft Entra ID. Для тестирования вы можете создать бесплатную пробную учетную запись в Microsoft Entra ID.

2. В центре администрирования Microsoft Entra выберите *Applications -> Enterprise applications -> New application -> Create your own application*.

3. Укажите имя вашего приложения и выберите параметр *Integrate any other application...*. После этого нажмите *Create*.

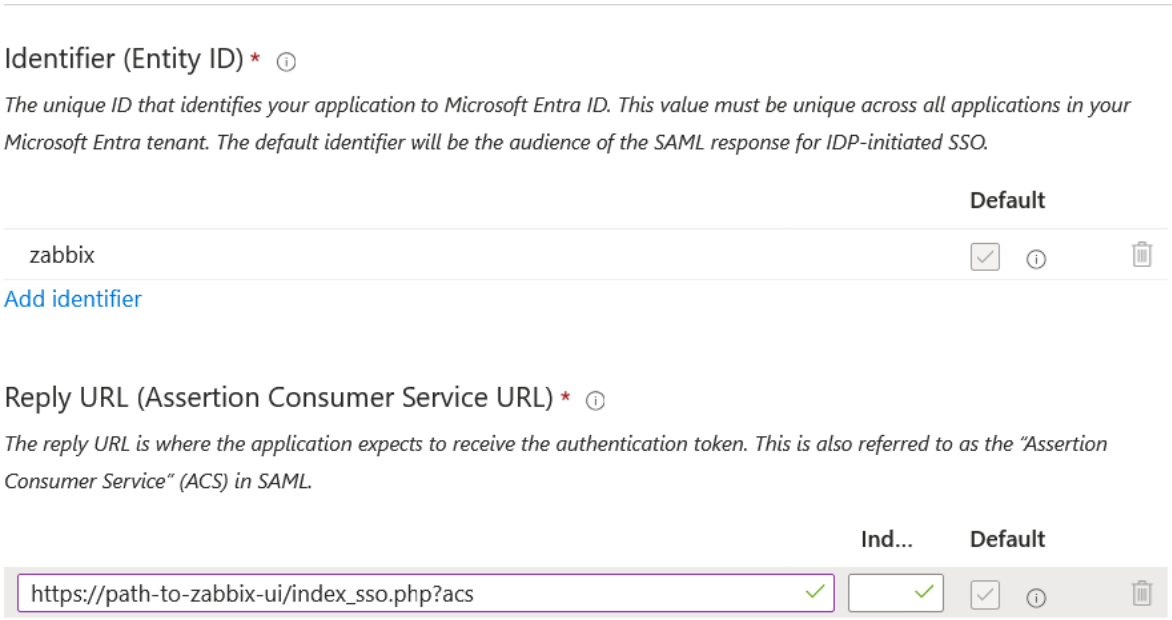



Настройка единого входа

1. На странице вашего приложения перейдите в раздел *Set up single sign on* и нажмите *Get started*. Затем выберите *SAML*.

2. Отредактируйте *Basic SAML Configuration*:

   - В *Identifier (Entity ID)* задайте уникальное имя, чтобы идентифицировать ваше приложение в Microsoft Entra ID, например, `zabbix`;
   - В *Reply URL (Assertion Consumer Service URL)* укажите конечную точку единого входа Zabbix: `https://<path-to-zabbix-ui>/`



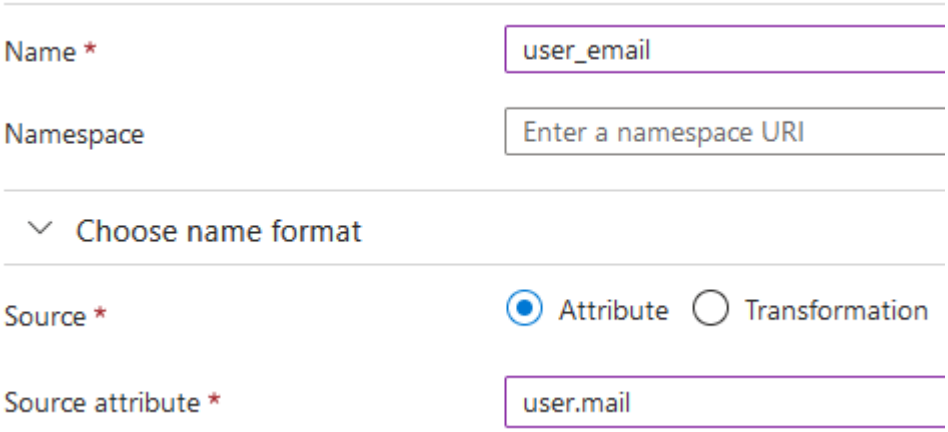

Обратите внимание, что требуется "https". Чтобы это работало с Zabbix, необходимо добавить в `conf/zabbix.conf.php` следующую строку:

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

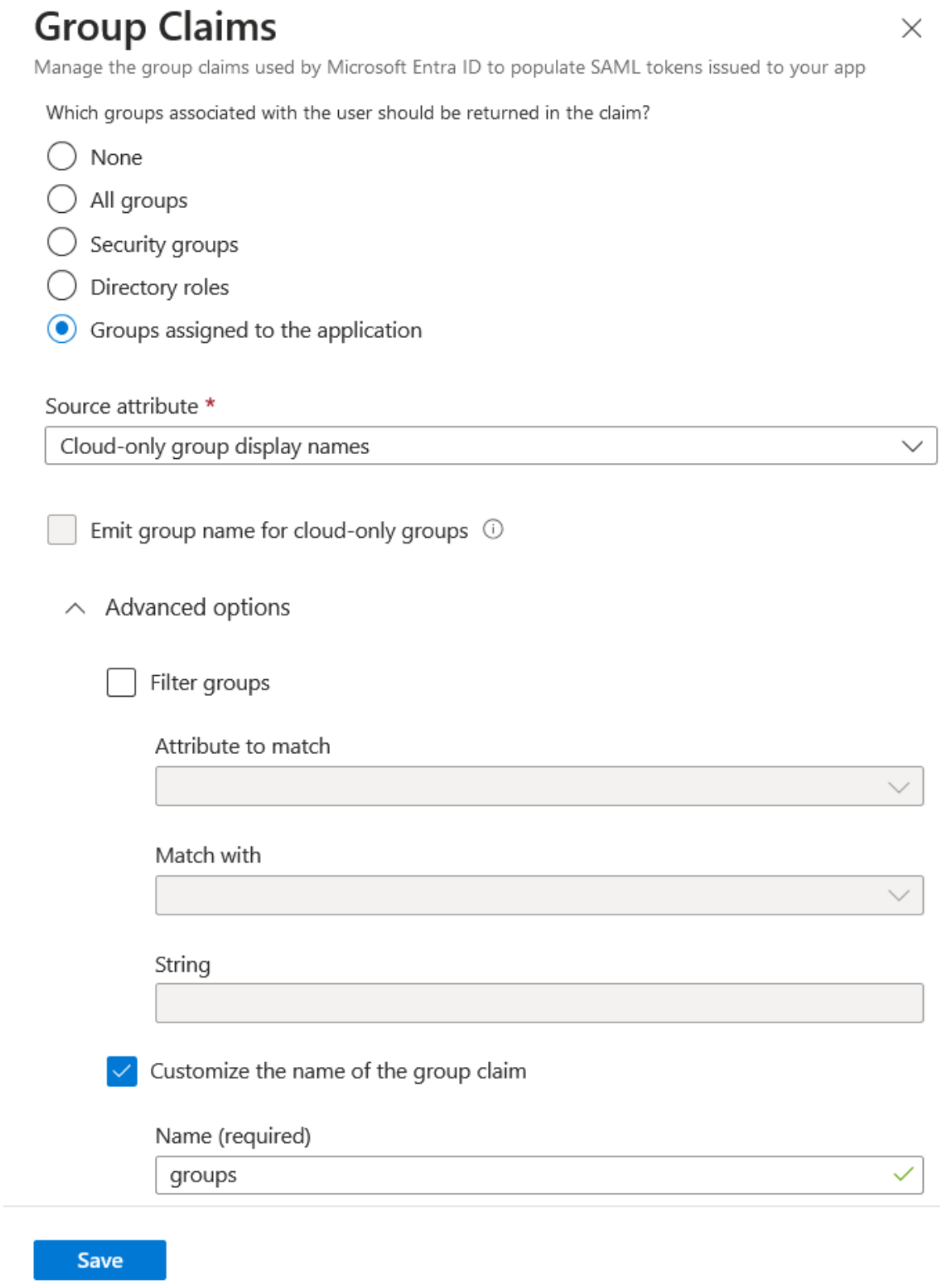
3. Отредактируйте *Attributes & Claims*. Необходимо добавить все атрибуты, которые вы хотите передать в Zabbix (user_name, user_lastname, user_email, user_mobile, groups).

Имена атрибутов задаются произвольно. Можно использовать разные имена атрибутов, однако требуется, чтобы они совпадали со значением соответствующего поля в настройках SAML Zabbix.

- Нажмите *Add new claim*, чтобы добавить атрибут:




- Нажмите *Add a group claim*, чтобы добавить атрибут для передачи групп в Zabbix:




Важно, чтобы в этом claim в Zabbix передавались имена групп, а не идентификаторы групп, через выбранный *Source attribute*. В противном случае JIT-подготовка пользователей не будет работать корректно.

4. В *SAML Certificates* скачайте Base64-сертификат, предоставленный Entra ID, и поместите его в `conf/certs` установки веб-интерфейса Zabbix.

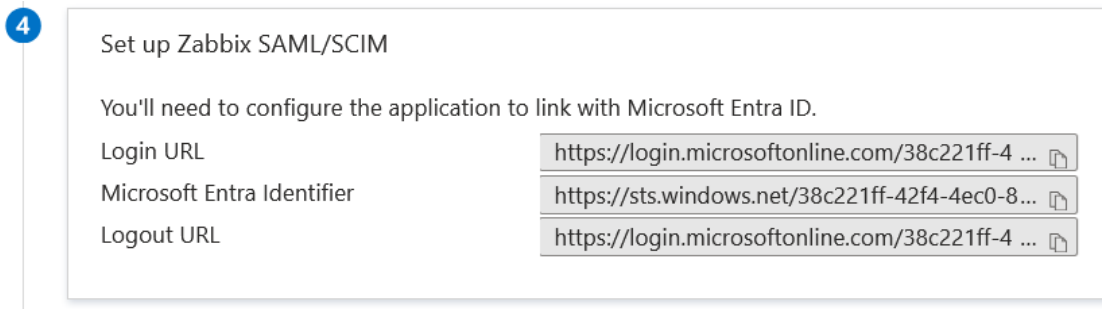
Установите для него права 644, выполнив:

```
chmod 644 entra.cer
```

Убедитесь, что `conf/zabbix.conf.php` содержит строку:

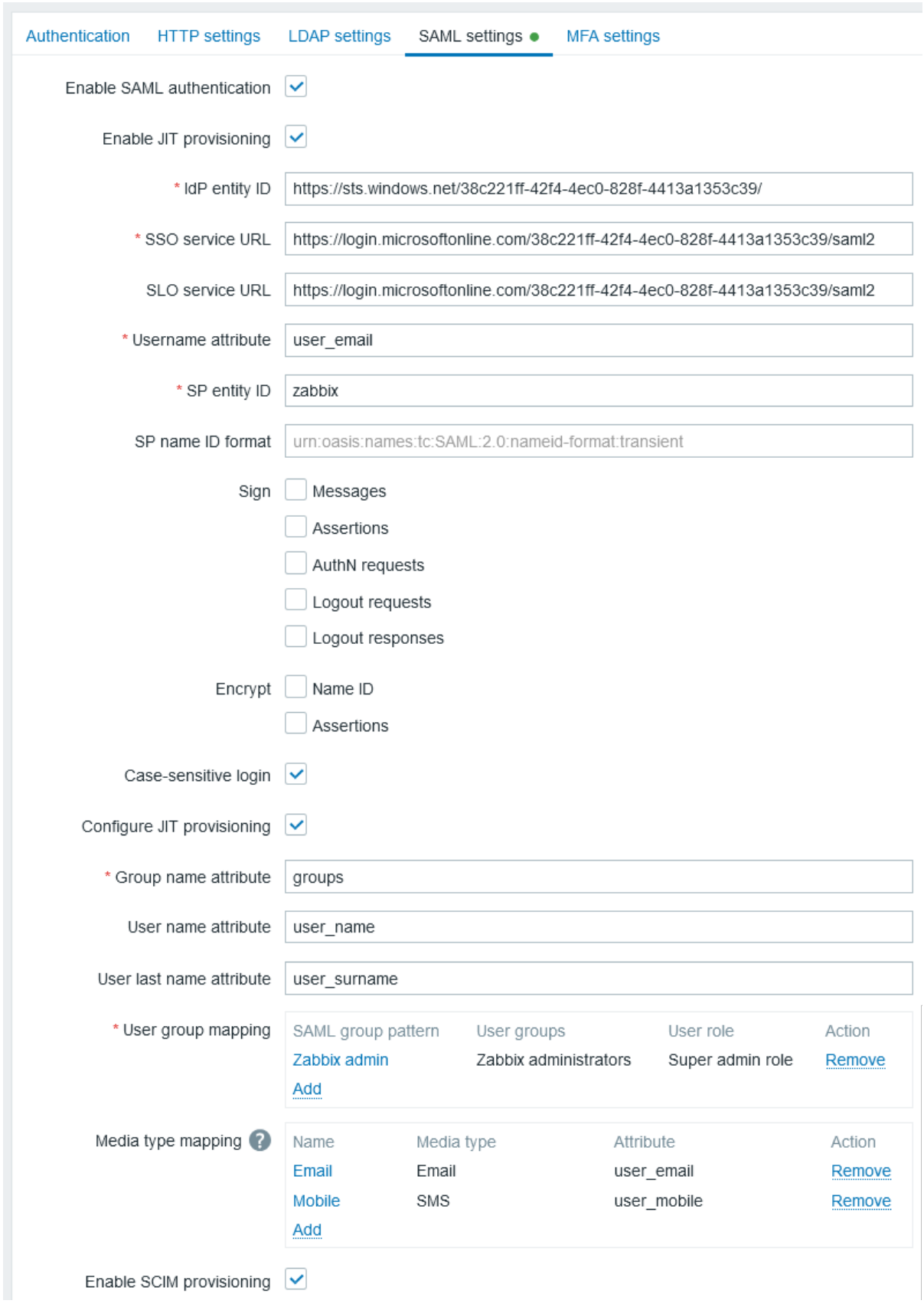
```
$SSO['IDP_CERT'] = 'conf/certs/entra.cer';
```

5. Используйте значения из *Set up <your app name>* в Entra ID для настройки аутентификации Zabbix SAML (см. следующий раздел):




Конфигурация Zabbix

1. В Zabbix перейдите в настройки SAML и заполните параметры конфигурации на основе конфигурации Entra ID:




| Поле Zabbix | Поле настройки в Entra ID | Пример значения |
|---|---|---|
| *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` |

| Поле Zabbix | Поле настройки в Entra ID | Пример значения |
|---|---|---|
| *User last name attribute* | Custom attribute (claim) | `user_lastname` |

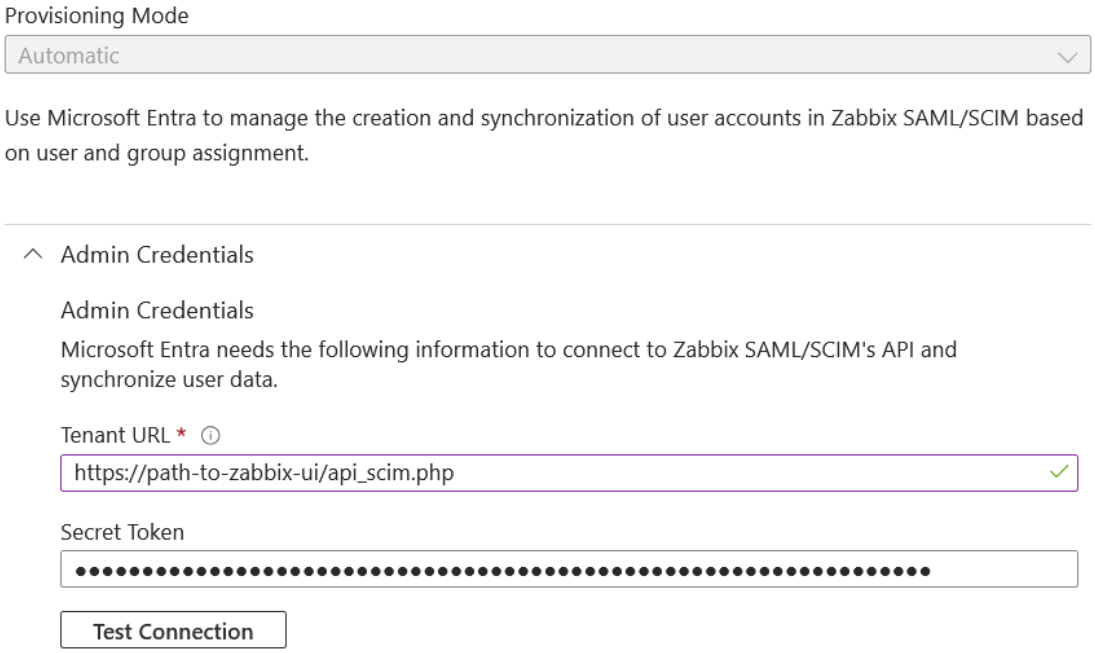
Также требуется настроить сопоставление групп пользователей. Сопоставление медиа является необязательным.

Нажмите *Update*, чтобы сохранить эти настройки.

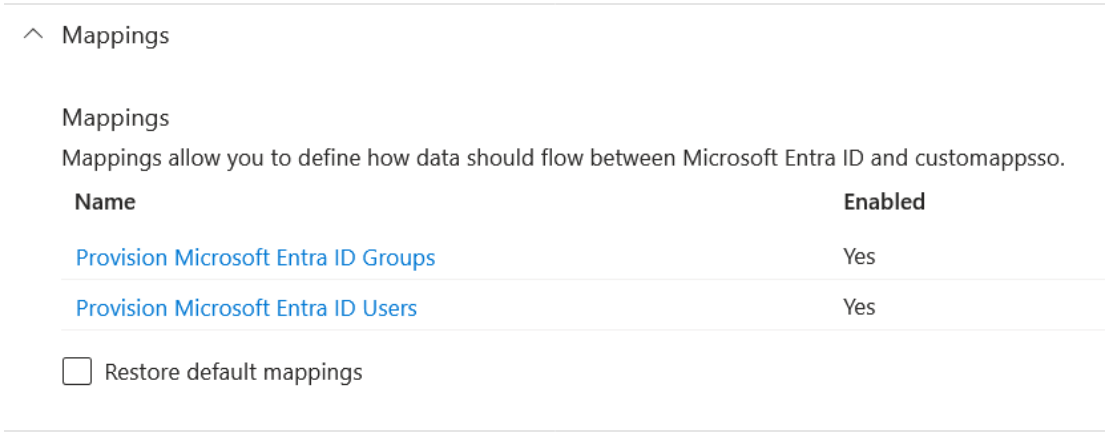
SCIM-подготовка пользователей

1. На странице приложения Entra ID в главном меню откройте страницу Provisioning. Нажмите *Get started* и затем выберите режим Automatic provisioning:

- В *Tenant URL* укажите следующее значение: `https://<path-to-zabbix-ui>/api_scim.php`
- В *Secret token* введите токен API Zabbix с правами Super admin.
- Нажмите *Test connection*, чтобы проверить, установлено ли соединение.

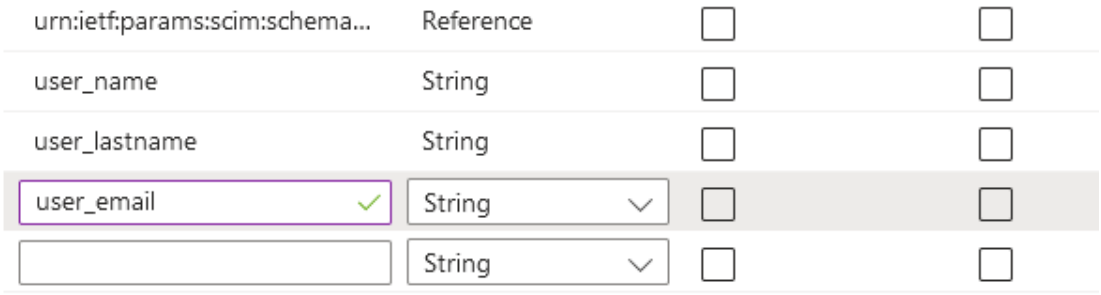



2. Теперь вы можете добавить все атрибуты, которые будут передаваться через SCIM в Zabbix. Для этого нажмите *Mappings*, а затем *Provision Microsoft Entra ID Users*.




Внизу списка Attribute Mapping включите *Show advanced options*, а затем нажмите *Edit attribute list for customappsso*.

Внизу списка атрибутов добавьте собственные атрибуты с типом 'String':




Сохраните список.

3. Теперь вы можете добавить сопоставления для добавленных атрибутов. Внизу списка Attribute Mapping нажмите *Add New Mapping* и создайте сопоставления, как показано ниже:

После добавления всех сопоставлений сохраните список сопоставлений.

4. В качестве предварительного условия для подготовки пользователей в Zabbix у вас должны быть настроены пользователи и группы в Entra ID.

Для этого перейдите в *Microsoft Entra admin center*, а затем добавьте пользователей/группы на соответствующих страницах Users и Groups.

5. После того как пользователи и группы будут созданы в Entra ID, вы можете перейти в меню *Users and groups* вашего приложения и добавить их в приложение.

6. Перейдите в меню *Provisioning* вашего приложения и нажмите *Start provisioning*, чтобы выполнить подготовку пользователей в Zabbix.

Обратите внимание, что запрос PATCH Users в Entra ID не поддерживает изменения в media.

Подписание запросов на аутентификацию

Entra ID можно настроить для проверки подписи подписанных запросов на аутентификацию.

Чтобы это заработало, создайте открытый и закрытый ключи:

```
openssl req -x509 -newkey rsa:4096 -keyout /usr/share/zabbix/conf/certs/request-sign.key -out /usr/share/z
```

Назначьте права доступа:

```
chown apache /usr/share/zabbix/conf/certs/request-sign.key
chmod 400 /usr/share/zabbix/conf/certs/request-sign.key
```

Обновите конфигурацию веб-интерфейса Zabbix, добавив:

```
$SSO['SP_KEY'] = 'conf/certs/request-sign.key';
$SSO['SP_CERT'] = 'conf/certs/request-sign.crt';
```

Устранение неполадок

Проблемы с аутентификацией могут возникать в браузерах MicroSoft Edge, когда пользователь, пытающийся войти в Zabbix через SAML, уже вошёл в систему с профилем MicroSoft Edge. Признаком такой проблемы может быть возможность входа в Zabbix через MicroSoft Edge в приватном режиме.

Чтобы избежать проблем с аутентификацией в этом случае, может потребоваться установить `requestedAuthnContext` в значение «false» в файле конфигурации веб-интерфейса Zabbix (*zabbix.conf.php*).

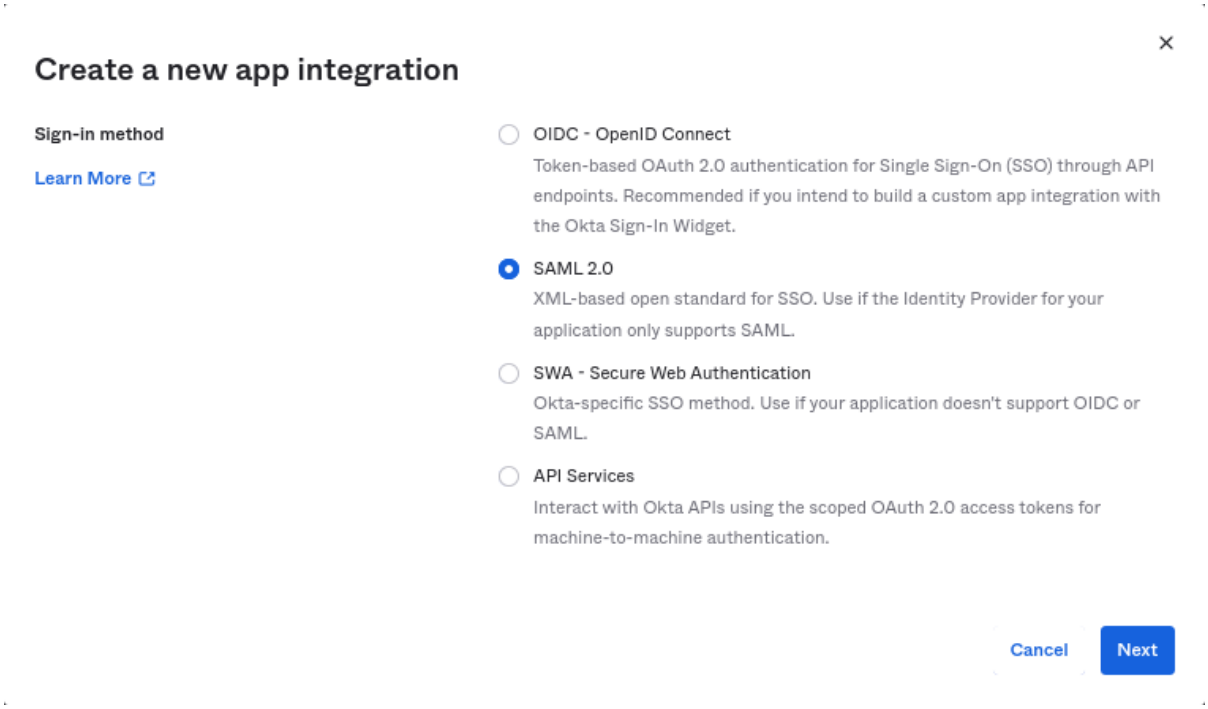
```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => false
    ]
];
```

#### 13 Настройка SAML с Okta

В этом разделе приведены рекомендации по настройке Okta для включения аутентификации SAML 2.0 и подготовки пользователей для Zabbix.

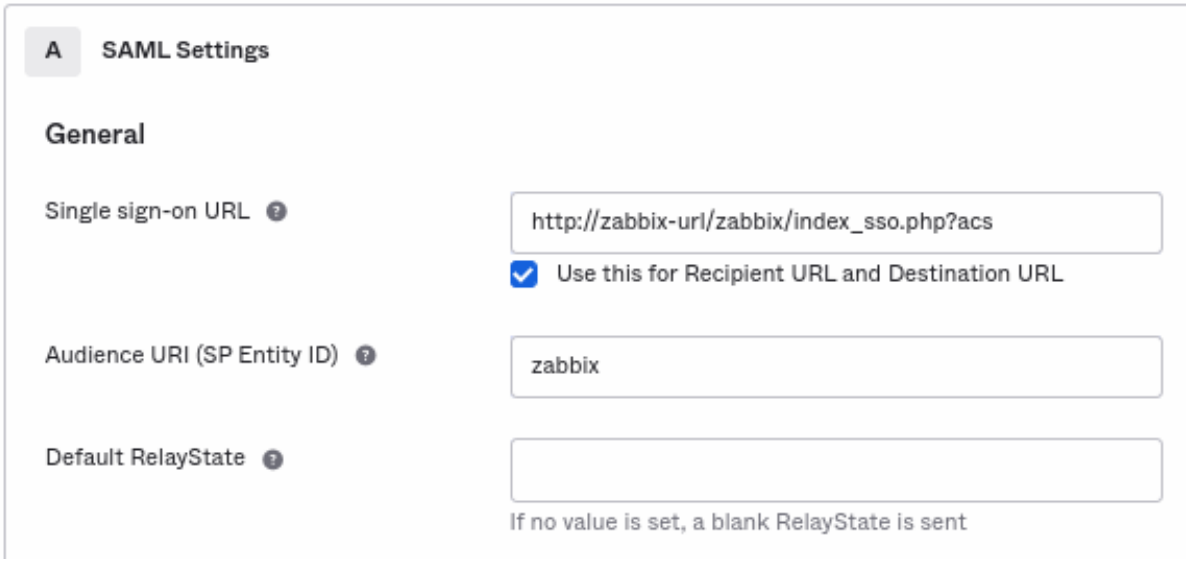
Настройка Okta

1. Перейдите по ссылке https://developer.okta.com/signup/ и зарегистрируйтесь/войдите в свою учетную запись.
2. В веб-интерфейсе Okta перейдите в *Applications* → *Applications*.
3. Нажмите *Create App Integration*.

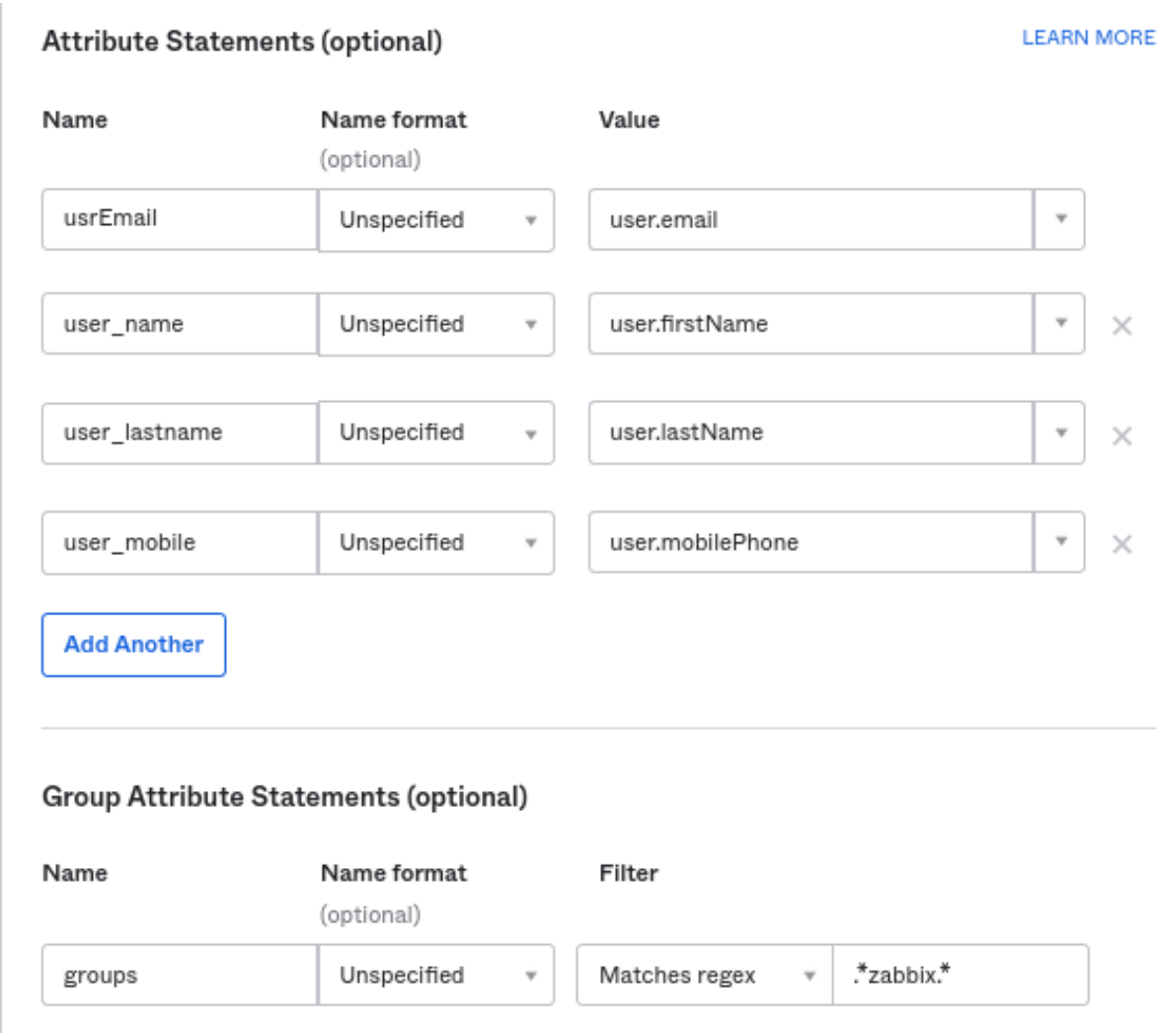



Выберите ”SAML 2.0” в качестве метода входа и нажмите *Next*.

4. В общих настройках укажите имя приложения и нажмите *Next*.
5. В настройках SAML введите значения, указанные ниже, затем нажмите *Next*.




- В разделе **General** добавьте:
  - *Single sign-on URL*: `http://<your-zabbix-url>/zabbix/index_sso.php?acs<br>` Обратите внимание на использование "http", а не "https", чтобы параметр `acs` не был обрезан в запросе. Также должен быть отмечен флажок *Use this for Recipient URL and Destination URL*.
  - *Audience URI (SP Entity ID)*: `zabbix<br>` Обратите внимание, что это значение будет использоваться в SAML assertion как уникальный идентификатор поставщика услуг (если оно не совпадает, операция будет отклонена). В этом поле можно указать URL или любую строку данных.
  - *Default RelayState*:<br> Оставьте это поле пустым; если требуется пользовательское перенаправление, его можно добавить в Zabbix в настройках *Users* → *Users*.
  - Заполните остальные поля по своему усмотрению.
- В разделе **Attribute Statements/Group Attribute Statements** добавьте:



Эти утверждения атрибутов вставляются в SAML assertion, которыми обмениваются с Zabbix.

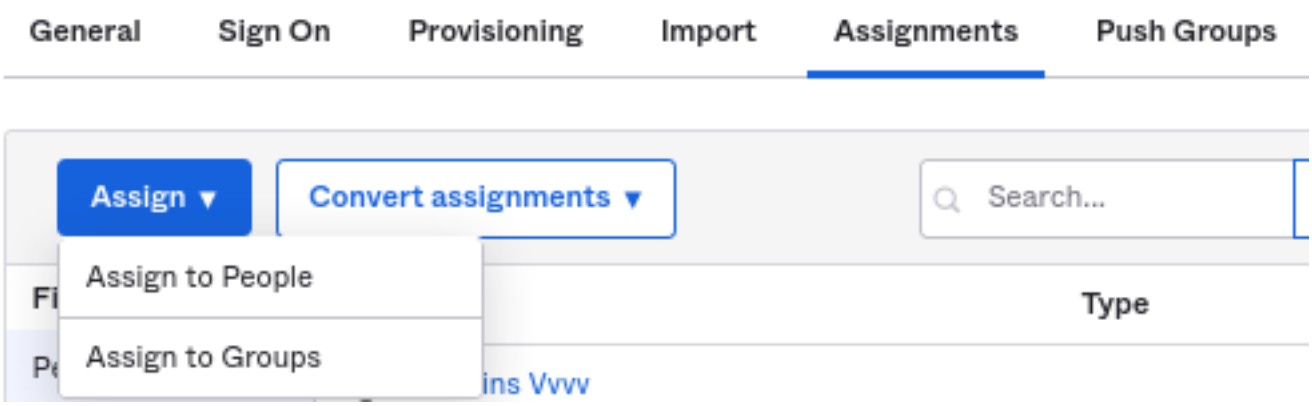
Имена атрибутов, используемые здесь, приведены в качестве произвольных примеров. Вы можете использовать другие имена атрибутов, однако они должны соответствовать значению соответствующего поля в настройках Zabbix SAML.

Если вы хотите настроить вход в Zabbix по SAML *без* JIT-подготовки пользователей, тогда требуется только атрибут email.

**Note:**
Если планируется использовать зашифрованное соединение, сгенерируйте закрытый и открытый сертификаты шифрования, затем загрузите открытый сертификат в Okta. Форма загрузки сертификата появляется, когда для *Assertion Encryption* установлено значение "Encrypted" (нажмите *Show Advanced Settings*, чтобы найти этот параметр).

6. На следующей вкладке выберите "I'm a software vendor. I'd like to integrate my app with Okta" и нажмите *Finish*.

7. Перейдите на вкладку "Assignments" только что созданного приложения и нажмите кнопку *Assign*, затем выберите *Assign to People* в раскрывающемся списке.



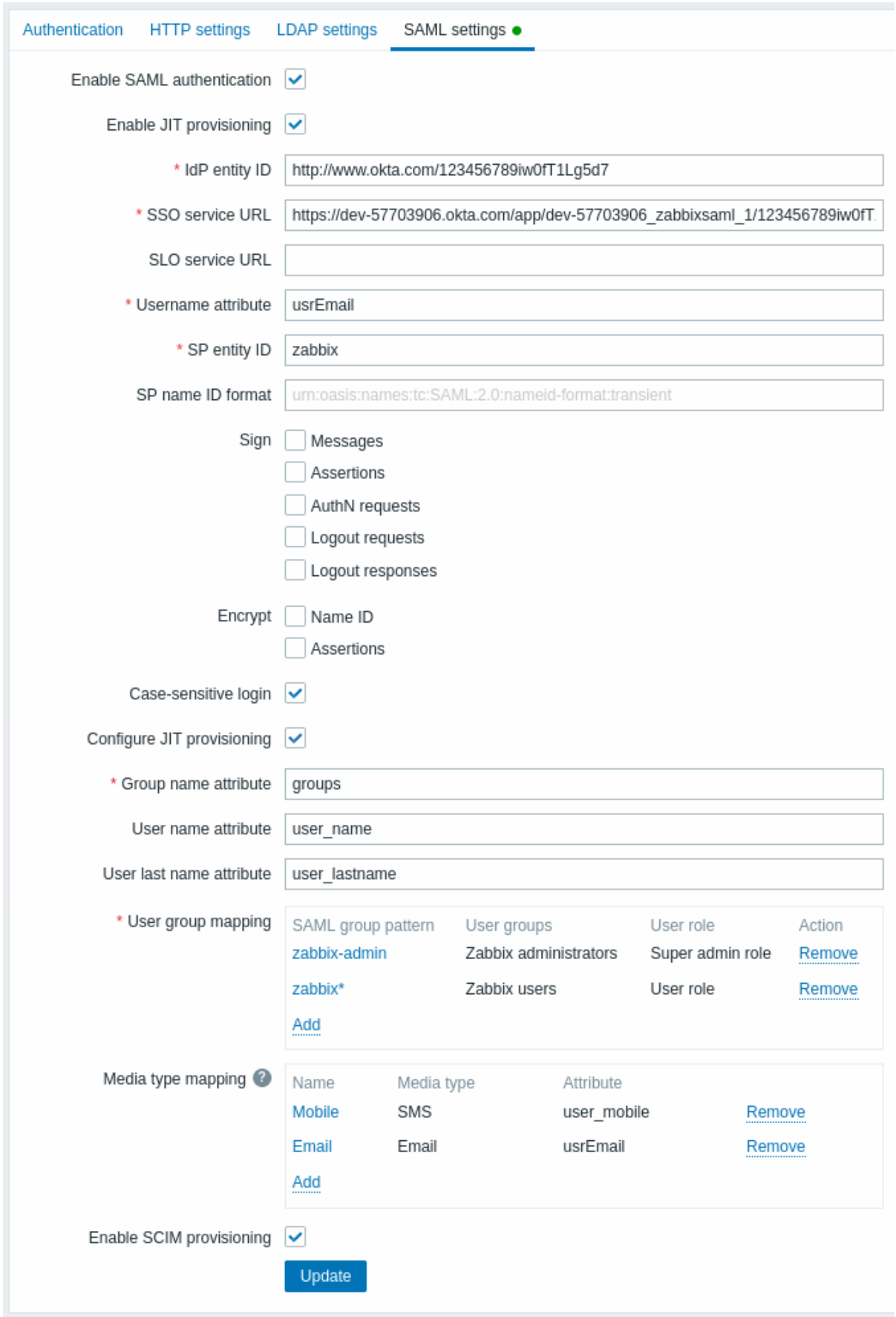

8. Во всплывающем окне назначьте приложение пользователям, которые будут использовать SAML 2.0 для аутентификации в Zabbix, затем нажмите *Save and go back*.

9. Перейдите на вкладку "Sign On" и нажмите кнопку *View Setup Instructions*.

**Инструкции** по настройке откроются в новой вкладке; оставьте эту вкладку открытой во время настройки Zabbix.

Конфигурация Zabbix

1. В Zabbix перейдите в настройки SAML и заполните параметры конфигурации на основе инструкций по настройке от Okta:




| Поле Zabbix | Поле настройки в Okta | Пример значения |
|---|---|---|
| *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` |

| Поле Zabbix | Поле настройки в Okta | Пример значения |
|---|---|---|
| *User name attribute* | Attribute name | `user_name` |
| *User last name attribute* | Attribute name | `user_lastname` |

Также требуется настроить сопоставление групп пользователей и медиа.

2. Загрузите сертификат, указанный в инструкциях по настройке SAML в Okta, в папку *ui/conf/certs* под именем idp.crt.

Установите для него права 644, выполнив:

```
chmod 644 idp.crt
```

3. Если для *Assertion Encryption* в Okta установлено значение "Encrypted", в Zabbix также должен быть отмечен флажок "Assertions" для параметра *Encrypt*.

4. Нажмите кнопку "Update", чтобы сохранить эти настройки.

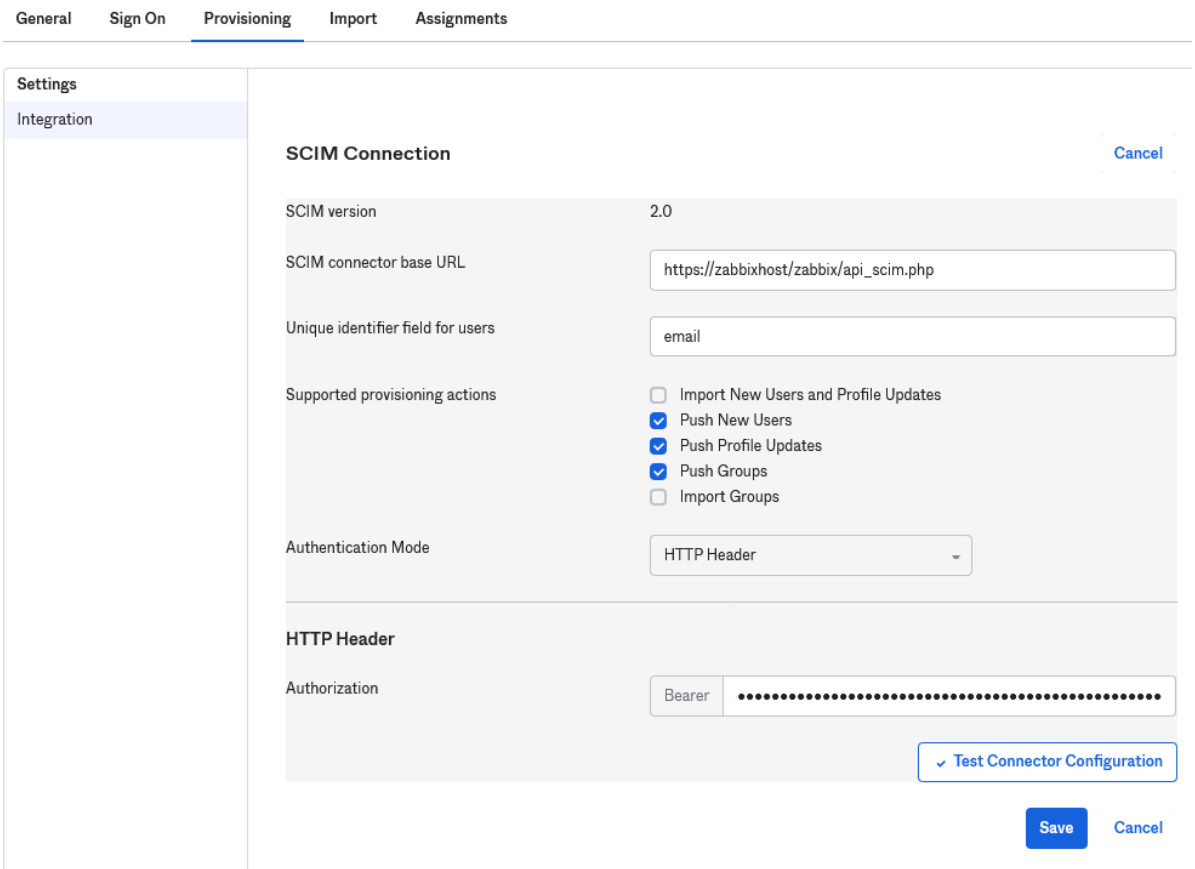
SCIM provisioning

1. Чтобы включить SCIM provisioning, перейдите в "General" -> "App Settings" приложения в Okta.

Установите флажок *Enable SCIM provisioning*. В результате появится новая вкладка *Provisioning*.

2. Перейдите на вкладку "Provisioning", чтобы настроить SCIM connection:

- В *SCIM connector base URL* укажите путь к веб-интерфейсу Zabbix и добавьте к нему `api_scim.php`, то есть:<br>`https://<your-zabbix-url>/zabbix/api_scim.php`
- *Unique identifier field for users*: `email`
- *Authentication mode*: `HTTP header`
- В *Authorization* укажите действительный API token с правами Super admin




**Attention:**
Если у вас возникают проблемы с аутентификацией, см. Authorization header forwarding.

3. Нажмите *Test Connector Configuration*, чтобы проверить connection. Если все верно, будет отображено сообщение об успешном выполнении.

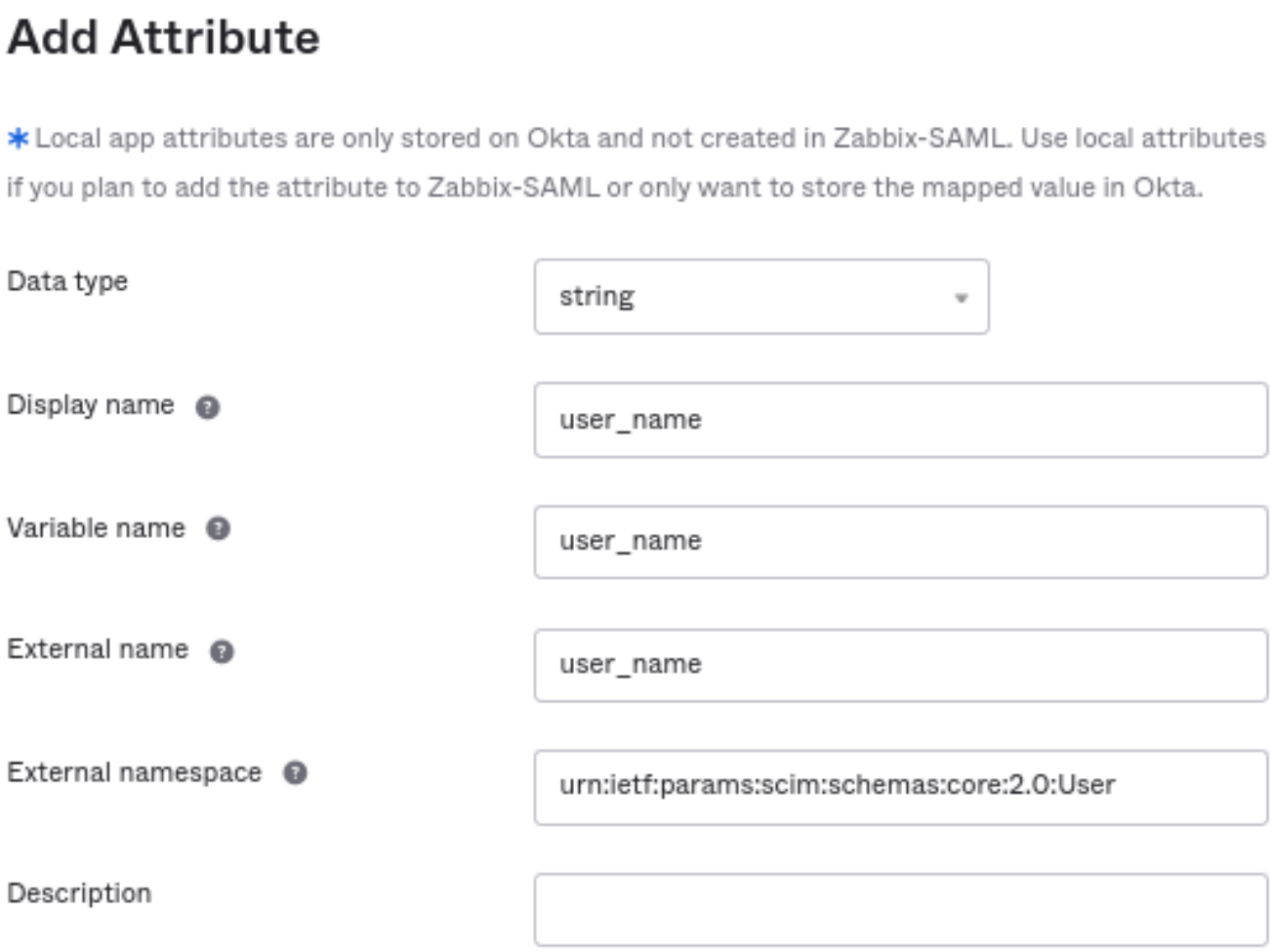
4. В "Provisioning" -> "To App" убедитесь, что отмечены следующие флажки:

- Create Users
- Update User Attributes
- Deactivate Users

Это гарантирует, что эти типы запросов будут отправляться в Zabbix.

5. Убедитесь, что все атрибуты, определенные в SAML, определены и в SCIM. Вы можете открыть редактор профиля для вашего приложения в ”Provisioning” -> ”To App”, нажав *Go to Profile Editor*.

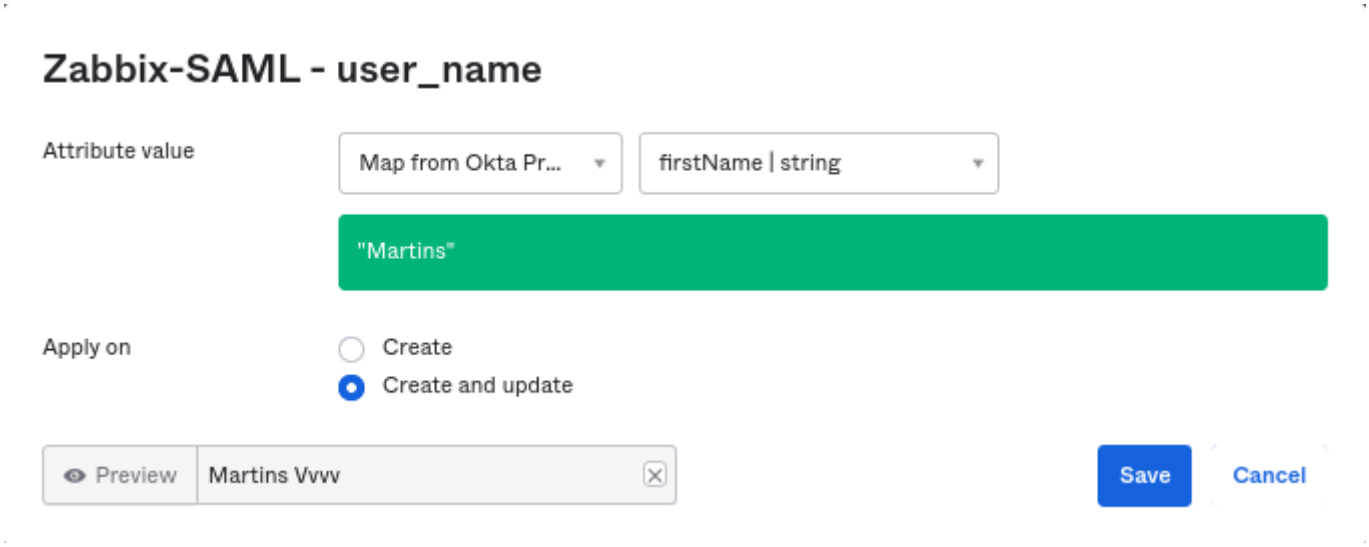
Нажмите *Add Attribute*. Заполните значения *Display name*, *Variable name*, *External name* именем атрибута SAML, например `user_name`.




*External namespace* должен совпадать со схемой пользователя: `urn:ietf:params:scim:schemas:core:2.0:User`

6. Перейдите в ”Provisioning” -> ”To App” -> ”Attribute Mappings” вашего приложения. Внизу нажмите *Show Unmapped Attributes*. Появятся недавно добавленные атрибуты.

7. Сопоставьте каждый добавленный атрибут.




8. Добавьте пользователей на вкладке ”Assignments”. Предварительно пользователи должны быть добавлены в *Directory -> People*. Все эти назначения будут отправлены в Zabbix в виде запросов.

9. Добавьте группы на вкладке ”Push Groups”. Шаблон сопоставления групп пользователей в настройках Zabbix SAML должен соответствовать группе, указанной здесь. Если совпадения нет, пользователь не может быть создан в Zabbix.

Информация о членах группы отправляется каждый раз при внесении изменений.

**14 Настройка SAML с OneLogin**

Обзор

В этом разделе приведены рекомендации по настройке единого входа и инициализации пользователей в Zabbix из OneLogin с использованием аутентификации SAML 2.0.

Настройка OneLogin

Создание приложения

1. Войдите в свою учетную запись OneLogin. Для целей тестирования вы можете создать бесплатную учетную запись разработчика в OneLogin.

2. В веб-интерфейсе OneLogin перейдите в *Applications → Applications*.

3. Нажмите ”Add App” и найдите подходящее приложение. Рекомендации на этой странице основаны на примере приложения *SCIM Provisioner with SAML (SCIM v2 Enterprise, full SAML)*.

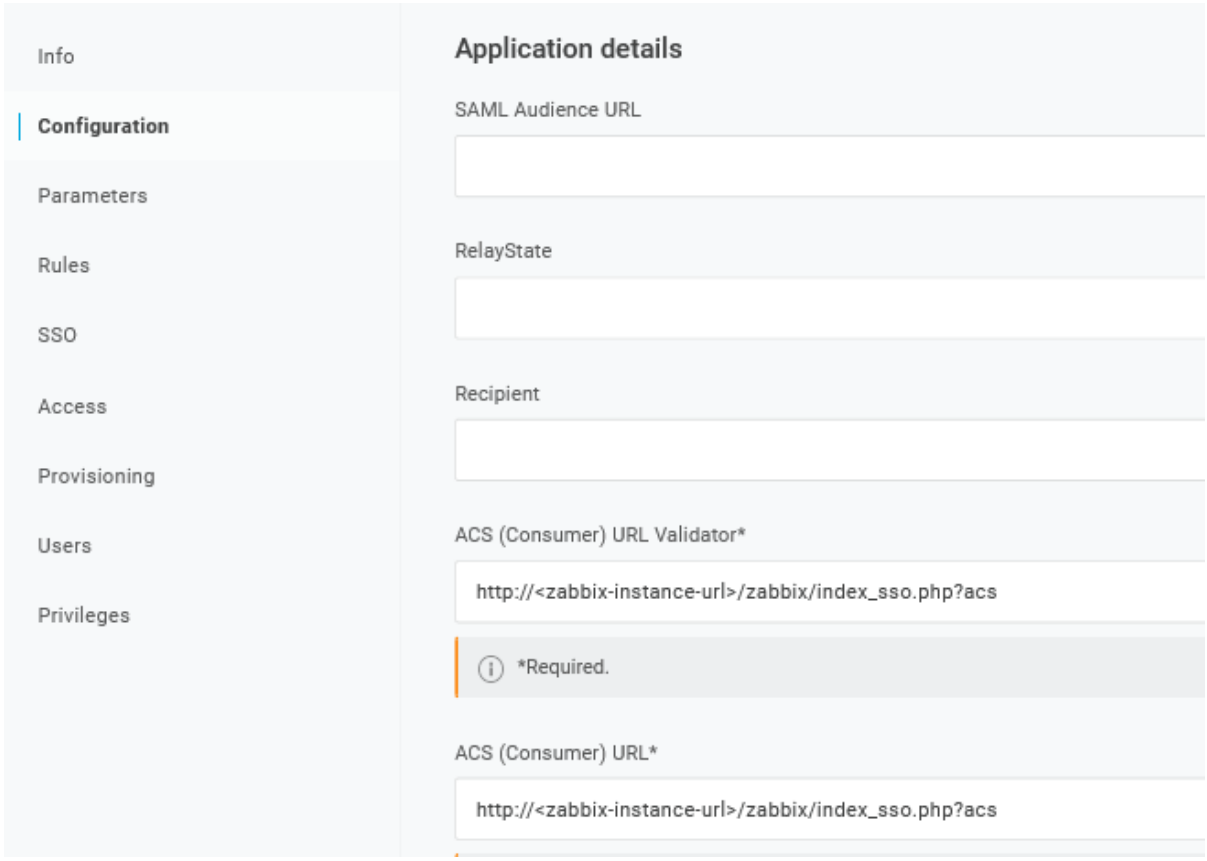
4. Для начала вы можете изменить отображаемое имя вашего приложения. Также можно добавить значок и сведения о приложении. После этого нажмите *Save*.

Настройка SSO/SCIM provisioning

1. В *Configuration -> Application details* задайте конечную точку единого входа Zabbix `http://<zabbix-instance-url>/zabbix/ind`
в качестве значения следующих полей:

- *ACS (Consumer) URL Validator*
- *ACS (Consumer) URL*

Обратите внимание на использование ”http”, а не ”https”, чтобы параметр `acs` не был обрезан в запросе.




Также можно использовать "https". Чтобы это работало с Zabbix, необходимо добавить в `conf/zabbix.conf.php` следующую строку:

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

Остальные параметры оставьте со значениями по умолчанию.

2. В *Configuration -> API connection* задайте следующие значения:

- *SCIM Base URL*: `https://<zabbix-instance-url>/zabbix/api_scim.php`
- *SCIM JSON Template*: должен содержать все пользовательские атрибуты, которые вы хотите передавать в Zabbix через SCIM, например `user_name`, `user_lastname`, `user_email` и `user_mobile`:

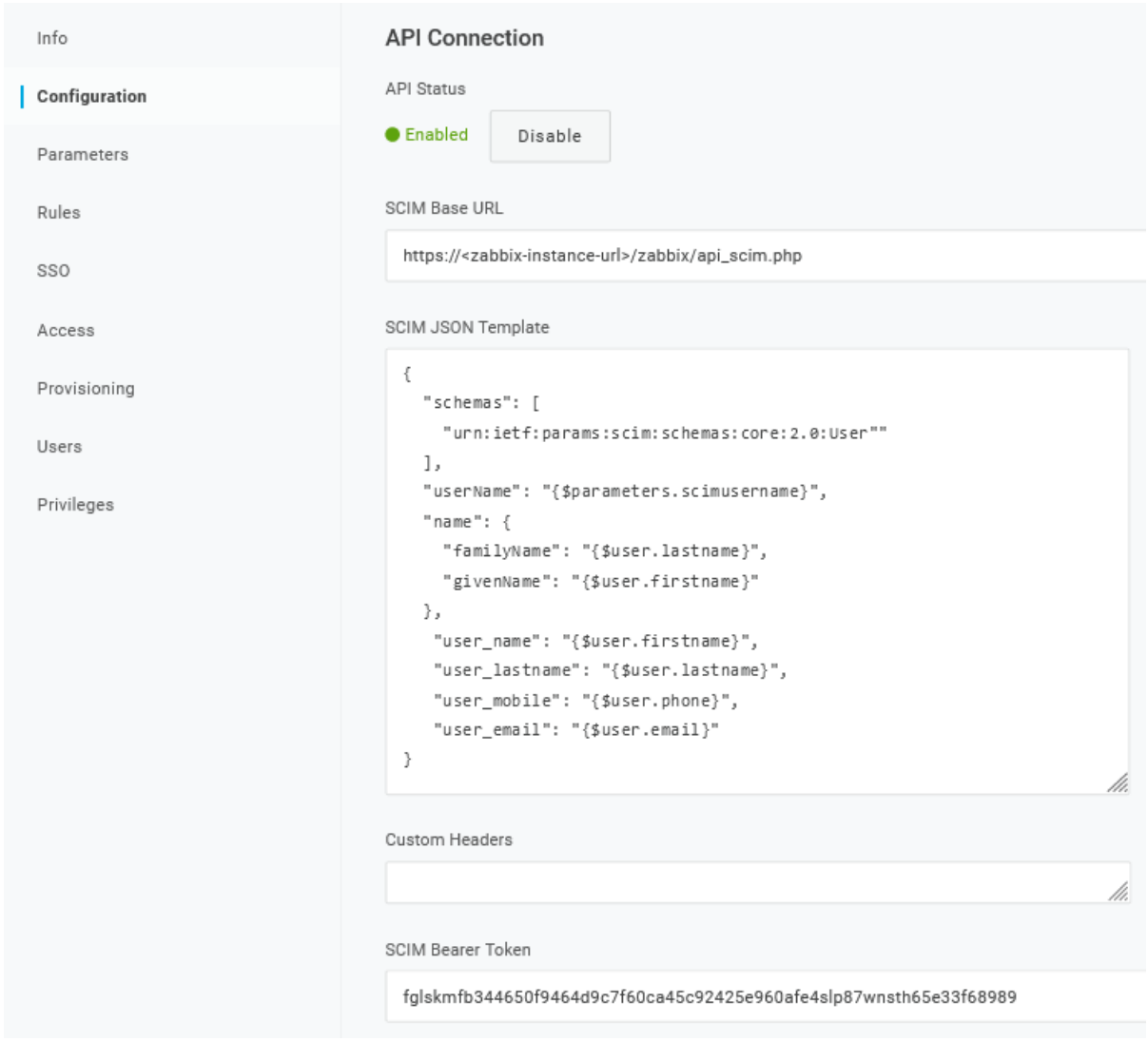
```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User"
  ],
  "userName": "{$parameters.scimusername}",
  "name": {
    "familyName": "{$user.lastname}",
    "givenName": "{$user.firstname}"
  },
   "user_name": "{$user.firstname}",
   "user_lastname": "{$user.lastname}",
   "user_mobile": "{$user.phone}",
   "user_email": "{$user.email}"
}
```

Имена атрибутов произвольны. Можно использовать другие имена атрибутов, однако требуется, чтобы они соответствовали значениям соответствующих полей в настройках Zabbix SAML.

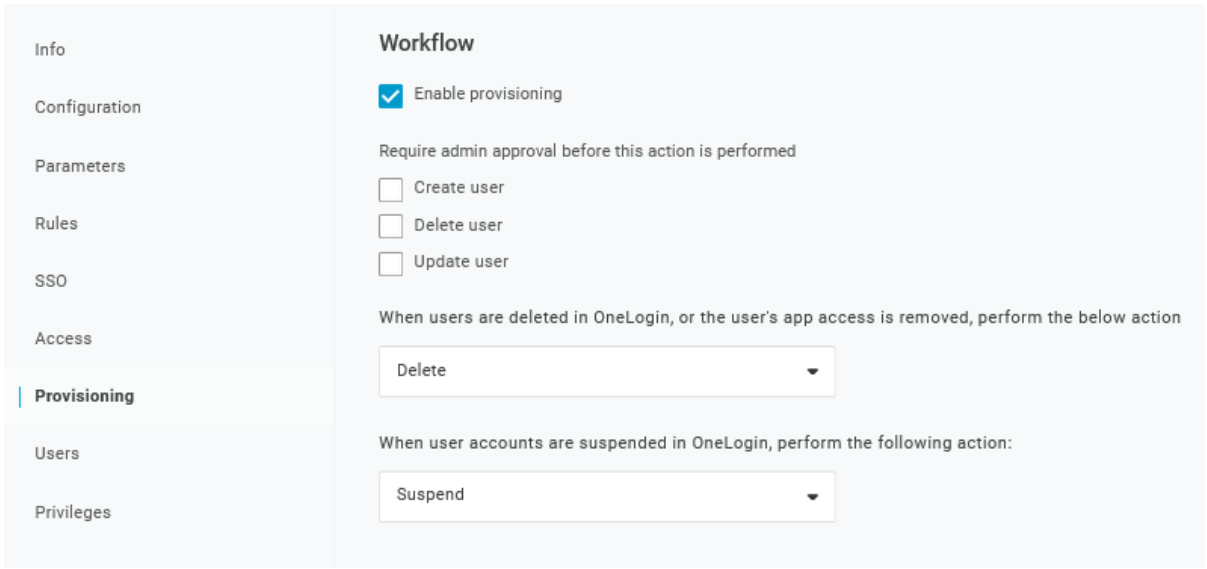
Обратите внимание, что для работы provisioning пользователей OneLogin должен получать в ответ атрибут `name` с `givenName` и `familyName`, даже если это не требуется поставщиком услуг. Поэтому необходимо указать это в схеме в разделе конфигурации приложения.

- *SCIM Bearer Token*: введите API token Zabbix с правами Super admin.

Нажмите *Enable*, чтобы активировать соединение.

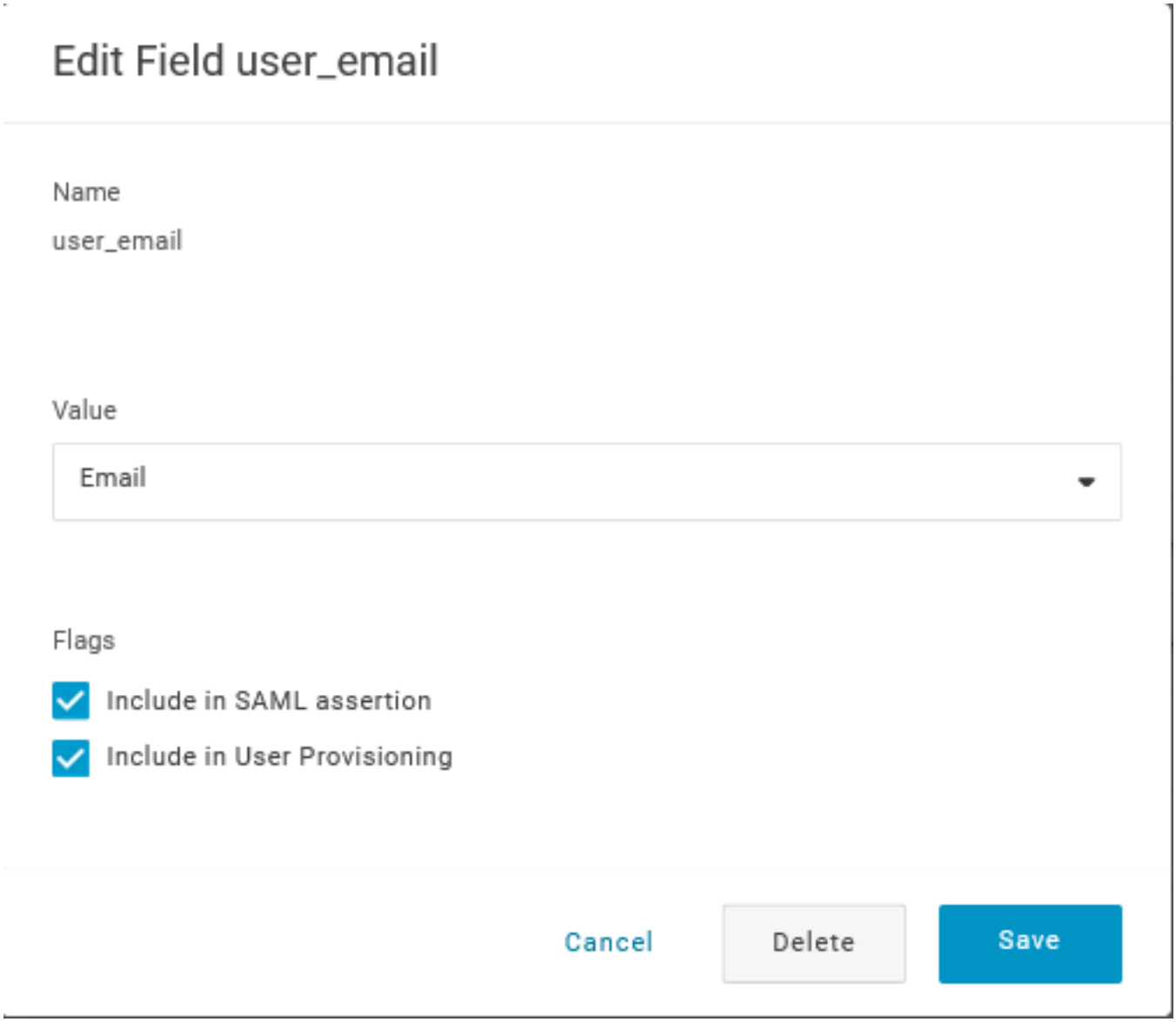



3. На странице *Provisioning* включите опцию Provisioning:



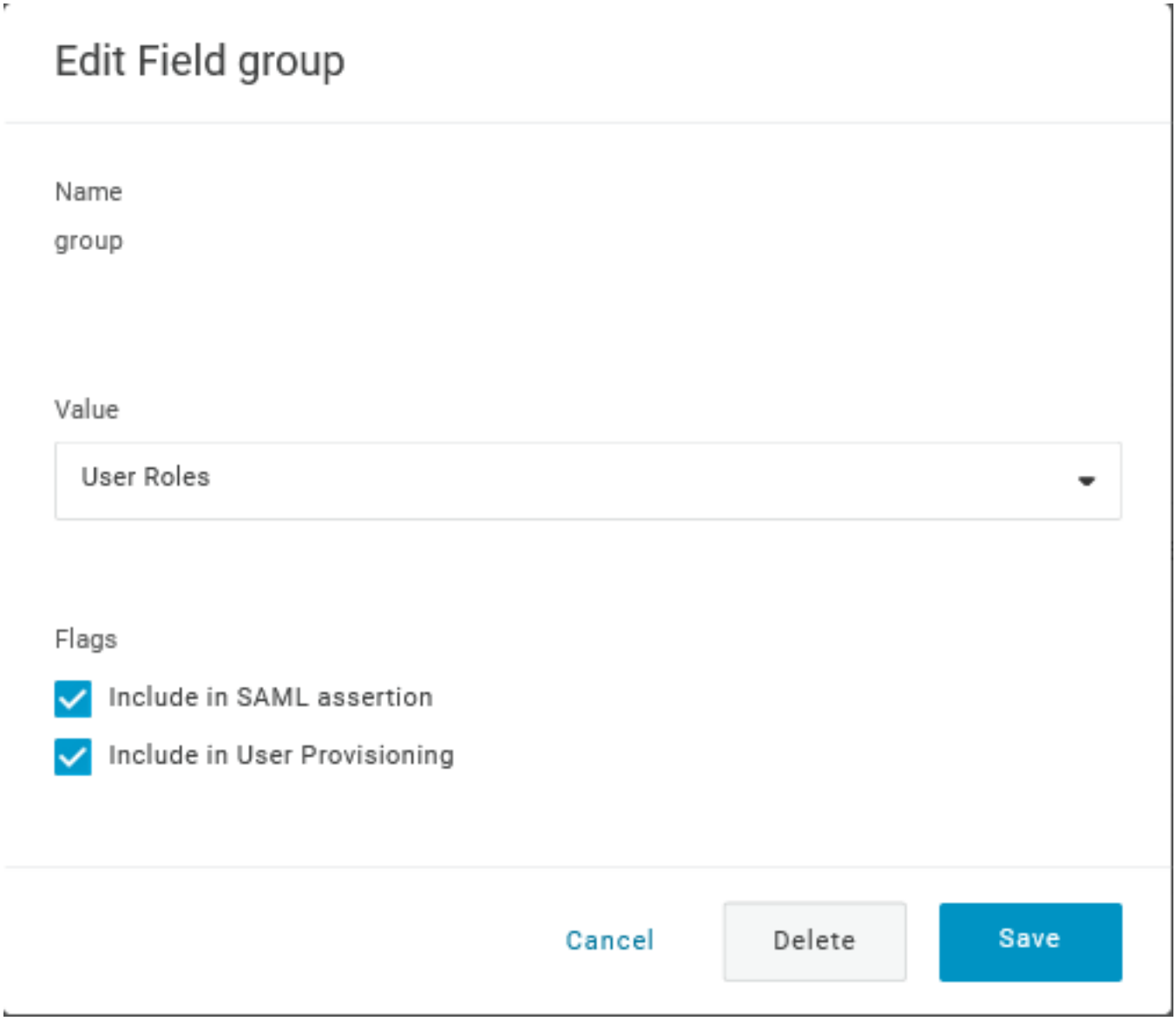

4. Страница *Parameters* содержит список параметров по умолчанию:

- Убедитесь, что значение 'scimusername' совпадает со значением логина пользователя в OneLogin (например, email);
- Отметьте опцию *Include in User Provisioning* для параметра 'Groups';
- Нажмите "+" для создания пользовательских параметров, которые требуются для SAML assertions и provisioning пользователей, таких как `user_name`, `user_lastname`, `user_email` и `user_mobile`:

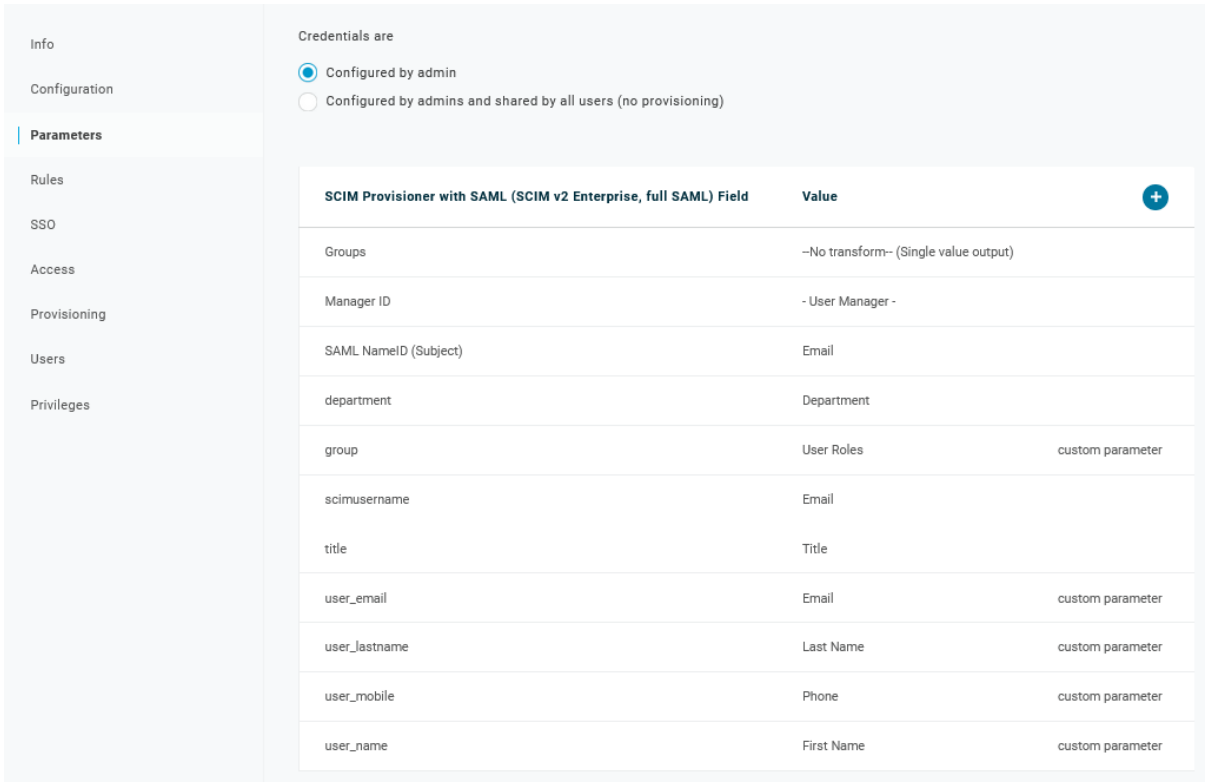



При добавлении параметра обязательно отметьте обе опции: *Include in SAML assertion* и *Include in User Provisioning*.

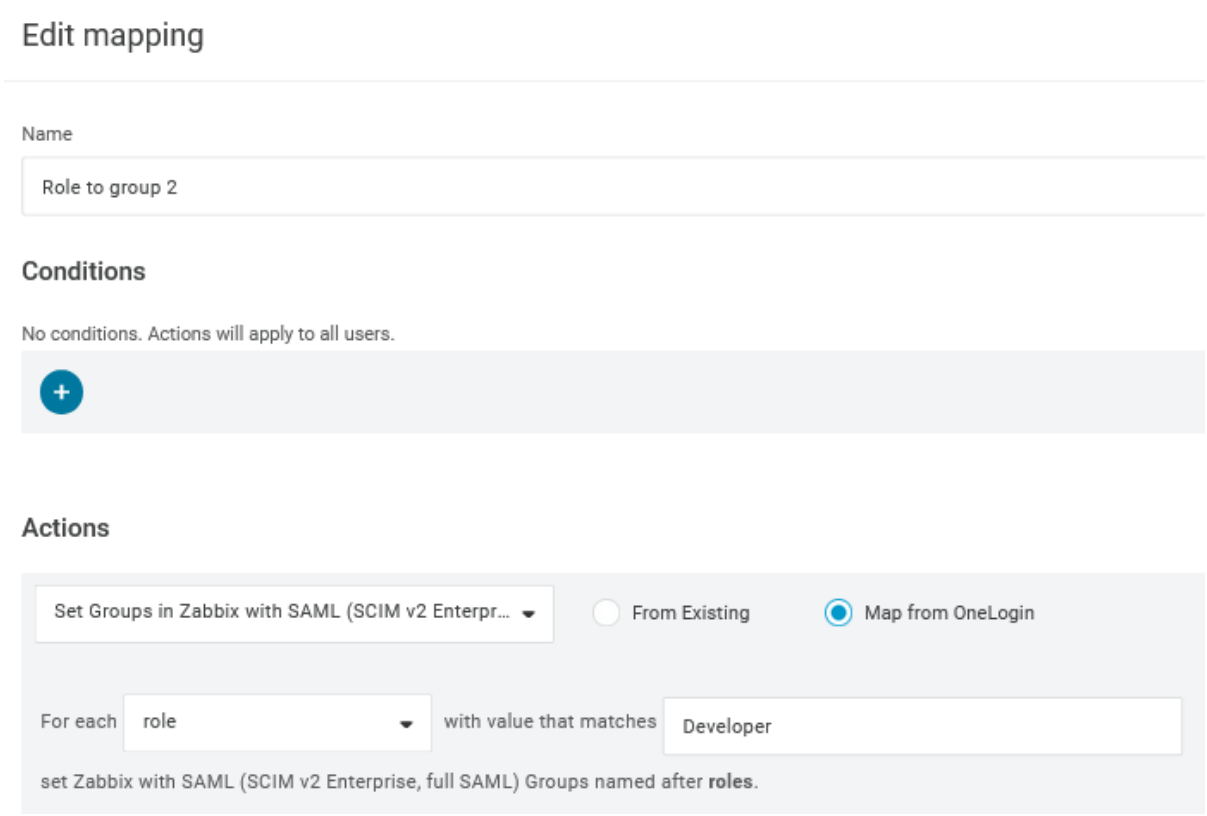
- Добавьте параметр 'group', который соответствует ролям пользователей в OneLogin. Роли пользователей будут передаваться в виде строки, разделенной точкой с запятой `;`. Роли пользователей OneLogin будут использоваться для создания групп пользователей в Zabbix:




Проверьте список параметров:



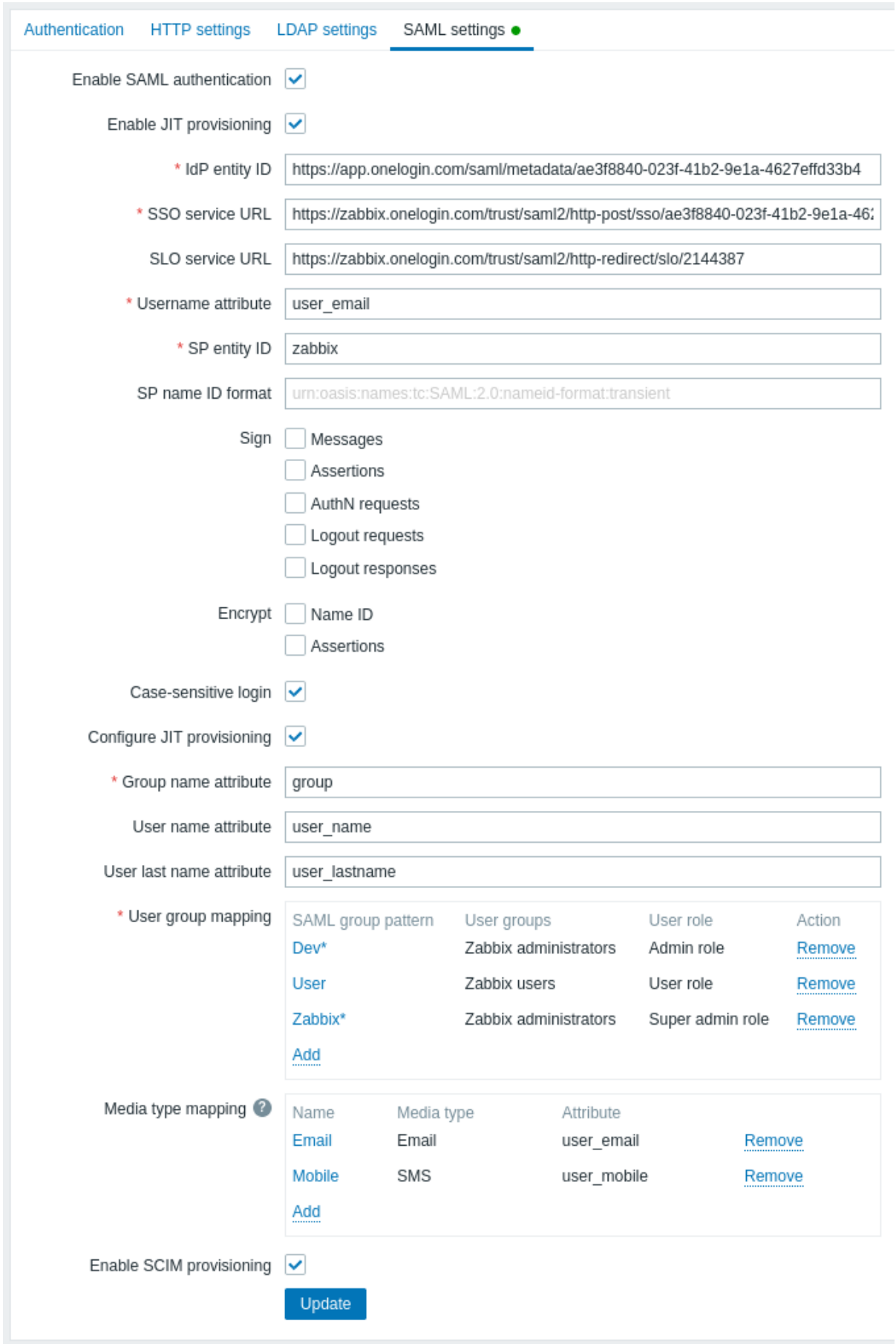
5. На странице *Rules* создайте сопоставления ролей пользователей с параметром Groups по умолчанию.




Можно использовать регулярное выражение для передачи определенных ролей как групп. Имена ролей не должны содержать `;`, так как OneLogin использует этот символ в качестве разделителя при отправке атрибута с несколькими ролями.

Конфигурация Zabbix

1. В Zabbix перейдите в настройки SAML и заполните параметры конфигурации на основе конфигурации OneLogin:




| Поле Zabbix | Поле настройки в OneLogin | Пример значения |
|---|---|---|
| *IdP entity ID* | Issuer URL<br>(см. вкладку SSO вашего приложения в OneLogin) | |

| Поле Zabbix | Поле настройки в OneLogin | Пример значения |
|---|---|---|
| *SSO service URL* | SAML 2.0 Endpoint (HTTP)<br>(см. вкладку SSO вашего приложения в OneLogin) | |
| *SLO service URL* | SLO Endpoint (HTTP)<br>(см. вкладку SSO вашего приложения в 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` |

Также требуется настроить сопоставление групп пользователей. Сопоставление медиа является необязательным. Нажмите *Update*, чтобы сохранить эти настройки.

2. Скачайте сертификат, предоставленный OneLogin, и поместите его в `conf/certs` установки веб-интерфейса Zabbix под именем idp.crt.

Установите для него права 644, выполнив:

```
chmod 644 idp.crt
```

Скачать сертификат в OneLogin можно в *Applications* -> *SSO* -> нажмите *View details* под текущим сертификатом.

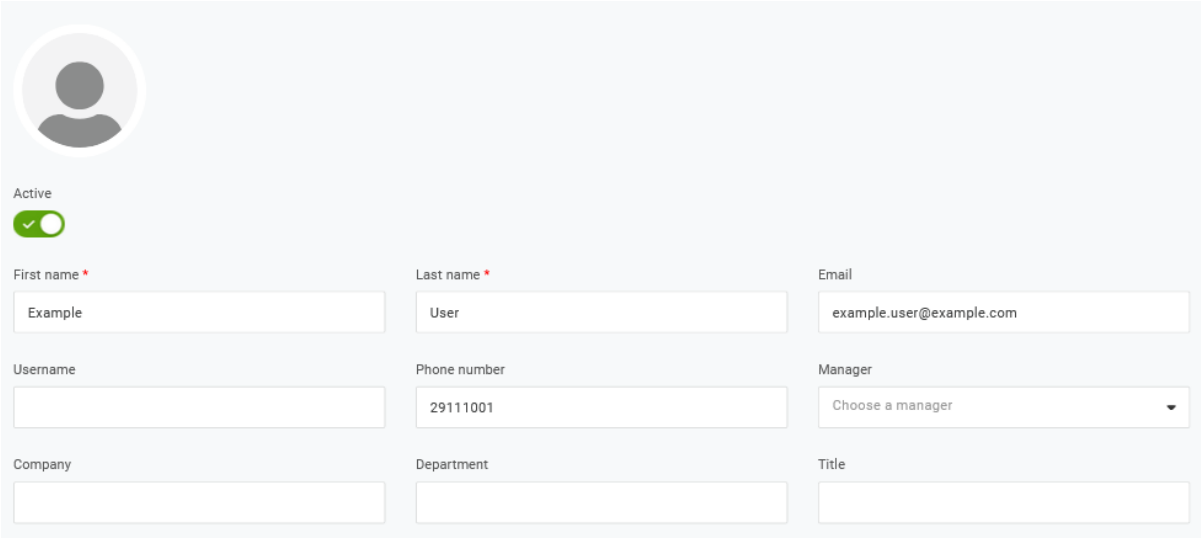
Можно использовать другое имя и расположение сертификата. В этом случае обязательно добавьте в `conf/zabbix.conf.php` следующую строку:

```
$SSO['IDP_CERT'] = 'path/to/certname.crt';
```

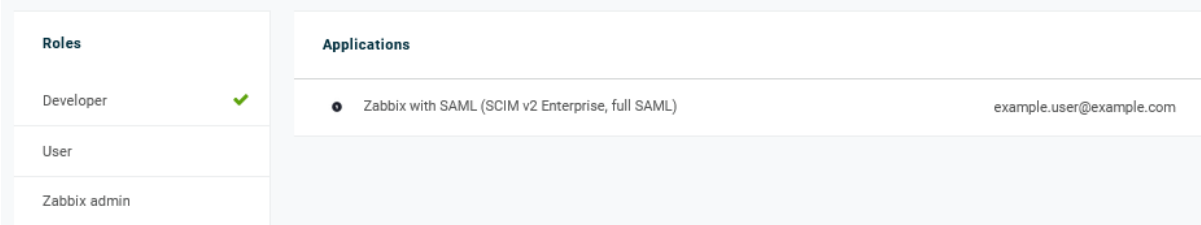
Инициализация пользователей SCIM

При включённой инициализации пользователей теперь можно добавлять/обновлять пользователей и их роли в OneLogin и немедленно получать их в Zabbix.

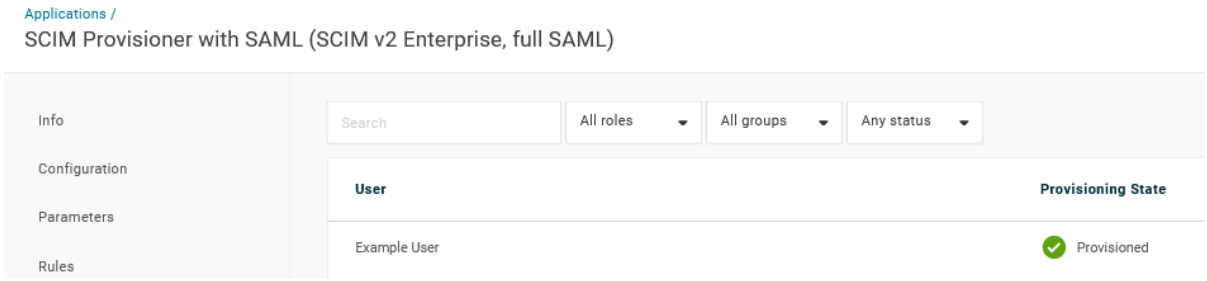
Например, вы можете создать нового пользователя:



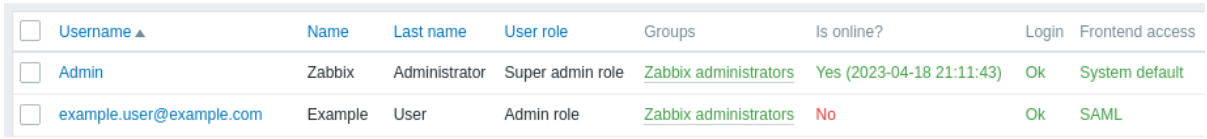

Добавьте его в роль пользователя и приложение, которое будет инициализировать пользователя:




При сохранении пользователя он будет инициализирован в Zabbix. В Application -> Users вы можете проверить статус инициализации текущих пользователей приложения:




Если инициализация прошла успешно, пользователь будет виден в списке пользователей 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 |

### 15 Настройка запланированных отчетов

Обзор

В этом разделе представлены инструкции по установке веб-сервиса Zabbix, а также настройке Zabbix, чтобы включить генерацию отчётов по расписанию.

Установка

Чтобы активировать создание отчётов по расписанию, должны быть установлены новый процесс веб-сервис Zabbix и браузер Google Chrome. Веб-сервис может быть установлен на том же компьютере, где установлен Zabbix сервер, или на другой машине. Браузер Google Chrome должен быть установлен на том же компьютере, где установлен веб-сервис.

Официальный пакет zabbix-web-service доступен в репозитории Zabbix. Браузер Google Chrome не включён в эти пакеты и должен быть установлен отдельно.

Чтобы скомпилировать веб-сервис Zabbix из исходных кодов, смотрите Установка веб-сервиса Zabbix.

После установки запустите zabbix_web_service на машине, где установлен веб-сервис:

```
zabbix_web_service
```

Настройка

Для обеспечения надлежащих коммуникаций между всеми задействованными элементами убедитесь, что параметры файла конфигурации сервера и файла конфигурации веб-интерфейса настроены должным образом.

Zabbix сервер

В файле конфигурации Zabbix сервера необходимо отредактировать следующие параметры: *WebServiceURL* и *StartReportWriters*.

**WebServiceURL**

Данный параметр необходим для активации коммуникаций с веб-сервисом. URL должен быть в формате `http[s]://  :  /report`.

- По умолчанию, веб-сервис слушает на порту 10053. В файле конфигурации веб-сервиса может быть указан другой порт.
- Указание пути `/report` является обязательным (этот путь жёстко зашит в коде и не может быть изменён).

Пример:

```
WebServiceURL=http://localhost:10053/report
```

**Attention:**
Настоятельно рекомендуется настроить шифрование между сервером Zabbix и веб-сервисом Zabbix с использованием сертификатов. По умолчанию данные, передаваемые между сервером Zabbix и веб-службой Zabbix, не шифруются, что может привести к несанкционированному доступу.

**StartReportWriters**

Данный параметр определяет, сколько процессов генерации отчётов («report writer») должно быть запущено. Если не задано или равно нулю, генерация отчётов отключена. Основываясь на количестве и частоте необходимых отчётов, можно активировать от 1 до 100 процессов генерации отчётов.

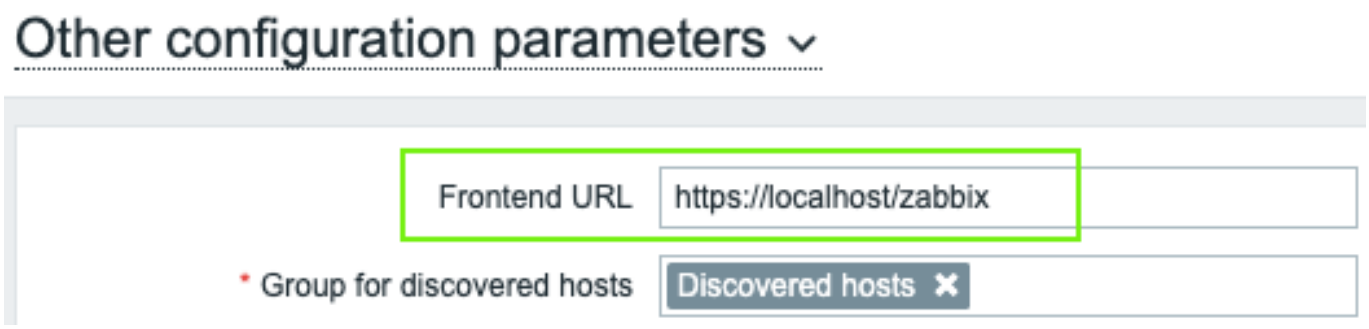
Пример:

```
StartReportWriters=3
```

Веб-интерфейс Zabbix

Чтобы разрешить коммуникации между веб-интерфейсом Zabbix и веб-сервисом Zabbix, должен быть выставлен параметр *URL веб-интерфейса (Frontend URL)* :

- Перейдите к разделу меню веб-интерфейса *Администрирование → Общие → Другое (Administration → General → Other)*
- Укажите полный URL веб-интерфейса Zabbix в параметре *URL веб-интерфейса (Frontend URL)*.




**Note:**
Чтобы убедиться, что всё работает правильно, после завершения процедуры первоначальной настройки Вы можете захотеть настроить и отослать тестовый отчёт.

**16 Дополнительные языки веб-интерфейса**

Обзор

Чтобы использовать в веб-интерфейсе Zabbix любой язык, кроме английского, на веб-сервере должна быть установлена соответствующая локаль. Кроме того, для работы переводов требуется расширение PHP gettext.

Установка локалей

Чтобы вывести список всех установленных языков, выполните:

```
locale -a
```

Если нужные языки отсутствуют в списке, откройте файл `/etc/locale.gen` и раскомментируйте требуемые локали. Поскольку Zabbix использует кодировку UTF-8, необходимо выбрать локали с набором символов UTF-8.

Теперь выполните:

```
locale-gen
```

Перезапустите веб-сервер. Обратите внимание: если PHP-FPM установлен и используется веб-сервером, его также нужно перезапустить вместе с веб-сервером.

Теперь локали должны быть установлены. Для появления новых языков может потребоваться обновить страницу веб-интерфейса Zabbix в браузере с помощью `Ctrl` + `F5`.

Установка Zabbix

Если Zabbix устанавливается напрямую из репозитория Zabbix git, файлы перевода следует создавать вручную. Чтобы создать файлы перевода, выполните:

```
make gettext
locale/make_mo.sh
```

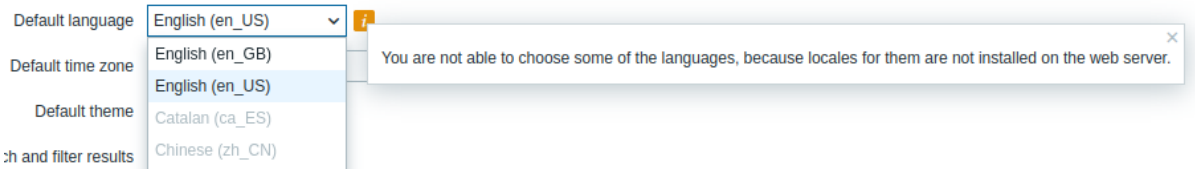
Этот шаг не требуется при установке Zabbix из пакетов или исходных файлов `tar.gz`.

Выбор языка

В веб-интерфейсе Zabbix есть несколько способов выбрать язык:

- При установке веб-интерфейса — в мастере установки веб-интерфейса. Выбранный язык будет установлен как системный по умолчанию.
- После установки системный язык по умолчанию можно изменить в разделе *Administration* > *General* > *GUI*.
- Язык для конкретного пользователя можно изменить в профиле пользователя.

Если для языка на машине не установлен locale, этот язык будет отображаться в селекторе языков Zabbix серым цветом. Если отсутствует хотя бы один locale, рядом с селектором языков отображается оранжевый значок. При нажатии на этот значок будет показано следующее сообщение: ”You are not able to choose some of the languages, because locales for them are not installed on the web server.”




**2. Конфигурация демонов**

В этом разделе описаны параметры конфигурационных файлов для Zabbix сервера, прокси-сервера, агента, агента 2 и других процессов.

Для перехода к соответствующему подразделу используйте боковое меню.

**1. Zabbix сервер**

Обзор

Параметры, поддерживаемые файлом конфигурации сервера Zabbix (zabbix_server.conf), перечислены в этом разделе.

Параметры приведены без дополнительной информации. Нажмите на параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| AlertScriptsPath | Расположение пользовательских скриптов оповещений. |
| AllowRoot | Разрешить серверу работать от имени ’root’. |
| AllowSoftwareUpdateCheck | Разрешить веб-интерфейсу Zabbix получать сведения об обновлениях программного обеспечения с zabbix.com. |
| AllowUnsupportedDBVersions | Разрешить серверу работать с неподдерживаемыми версиями базы данных. |
| CacheSize | Размер кэша конфигурации. |
| CacheUpdateFrequency | Этот параметр определяет, как часто Zabbix будет выполнять обновление кэша конфигурации, в секундах. |
| DBHost | Узел сети (или каталог сокета) базы данных Zabbix. |
| DBName | Имя базы данных. |
| DBPassword | Пароль базы данных. |
| DBPort | Порт узла сети базы данных Zabbix. |
| DBSchema | Имя схемы базы данных. Используется для PostgreSQL. |
| DBSocket | Путь к файлу сокета MySQL. |
| DBUser | Пользователь базы данных. |
| DBTLSConnect | Указание этого параметра со значением, заданным в опции, принудительно использует TLS-соединение с базой данных. |
| DBTLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA(ов) для проверки сертификата базы данных. |
| DBTLSCertFile | Полный путь к файлу, содержащему сертификат сервера Zabbix для аутентификации в базе данных. |
| DBTLSKeyFile | Полный путь к файлу, содержащему закрытый ключ для аутентификации в базе данных. |
| DBTLSCipher | Список шифров шифрования, которые сервер Zabbix разрешает для протоколов TLS вплоть до TLS v1.2. Поддерживается только для MySQL. |
| DBTLSCipher13 | Список наборов шифров шифрования, которые сервер Zabbix разрешает для протокола TLS v1.3. Поддерживается только для MySQL, начиная с версии 8.0.16. |
| DebugLevel | Уровень отладки. |
| EnableGlobalScripts | Включить глобальные скрипты на сервере Zabbix. |
| ExportDir | Каталог для экспорта событий, истории и трендов в реальном времени в формате JSON с разделением по строкам. Если задан, включает экспорт в реальном времени. |
| ExportFileSize | Максимальный размер одного файла экспорта в байтах. |
| ExportType | Список типов сущностей, разделённых запятыми (events, history, trends), для экспорта в реальном времени (по умолчанию все типы). |
| ExternalScripts | Расположение внешних скриптов. |

| Parameter | Description |
|---|---|
| Fping6Location | Расположение fping6. |
| FpingLocation | Расположение fping. |
| FrontendAllowedIP | Список IP-адресов или диапазонов CIDR, которым разрешено подключаться из веб-интерфейса. |
| HANodeName | Имя узла кластера высокой доступности. |
| HistoryCacheSize | Размер кэша истории. |
| HistoryIndexCacheSize | Размер кэша индекса истории. |
| HistoryStorageDateIndex | Включить предварительную обработку значений истории в хранилище истории для сохранения значений в разных индексах на основе даты. |
| HistoryStorageURL | HTTP[S]-URL хранилища истории. |
| HistoryStorageTypes | Список типов значений, разделённых запятыми, которые будут отправляться в хранилище истории. |
| HousekeepingFrequency | Этот параметр определяет, как часто Zabbix будет выполнять процедуру очистки, в часах. |
| Include | В файл конфигурации можно включать отдельные файлы или все файлы из каталога. |
| JavaGateway | IP-адрес (или имя узла сети) Java gateway Zabbix. |
| JavaGatewayPort | Порт, на котором Java gateway Zabbix принимает подключения. |
| ListenBacklog | Максимальное число ожидающих подключений в очереди TCP. |
| ListenIP | Список IP-адресов, разделённых запятыми, на которых должен прослушивать trapper. |
| ListenPort | Порт прослушивания для trapper. |
| LoadModule | Модуль, который нужно загрузить при запуске сервера. |
| LoadModulePath | Полный путь к расположению модулей сервера. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogSlowQueries | Определяет, сколько времени может выполняться запрос к базе данных до его записи в журнал, в миллисекундах. |
| LogType | Тип вывода журнала. |
| MaxConcurrentChecksPerPoller | Максимальное число асинхронных проверок, которые могут выполняться одновременно каждым опросчиком HTTP agent, agent или SNMP. |
| MaxHousekeeperDelete | Не более 'MaxHousekeeperDelete' строк (соответствующих [tablename], [field], [value]) будет удалено за одну задачу в одном цикле очистки. |
| NodeAddress | IP-адрес или имя узла сети с необязательным портом, чтобы переопределить способ подключения веб-интерфейса к серверу. |
| PidFile | Имя файла PID. |
| ProblemHousekeepingFrequency | Определяет, как часто Zabbix будет удалять проблемы для удалённых триггеров. |
| ProxyConfigFrequency | Определяет, как часто сервер Zabbix отправляет данные конфигурации прокси Zabbix. |
| ProxyDataFrequency | Определяет, как часто сервер Zabbix запрашивает данные истории у прокси Zabbix. |
| ServiceManagerSyncFrequency | Определяет, как часто Zabbix будет синхронизировать конфигурацию менеджера служб. |
| SMSDevices | Список файлов модемов, разделённых запятыми, которые разрешено использовать серверу Zabbix. |
| SNMPTrapperFile | Временный файл, используемый для передачи данных от демона SNMP trap к серверу. |
| SocketDir | Каталог для хранения IPC-сокетов, используемых внутренними службами Zabbix. |
| SourceIP | Исходный IP-адрес. |
| SSHKeyLocation | Расположение открытых и закрытых ключей для проверок и действий SSH. |
| SSLCertLocation | Расположение файлов SSL-сертификатов клиента для аутентификации клиента. |
| SSLKeyLocation | Расположение файлов закрытых ключей SSL для аутентификации клиента. |
| SSLCALocation | Переопределяет расположение файлов центра сертификации (CA) для проверки сертификата сервера SSL. |
| StartAgentPollers | Количество предварительно порождённых экземпляров асинхронных опросчиков agent Zabbix. |
| StartAlerters | Количество предварительно порождённых экземпляров обработчиков оповещений. |
| StartBrowserPollers | Количество предварительно порождённых экземпляров опросчиков элементов данных браузера. |
| StartConnectors | Количество предварительно порождённых экземпляров рабочих процессов коннекторов. |
| StartDBSyncers | Количество предварительно порождённых экземпляров синхронизаторов истории. |
| StartDiscoverers | Количество предварительно порождённых экземпляров рабочих процессов обнаружения. |
| StartEscalators | Количество предварительно порождённых экземпляров эскалаторов. |
| StartHistoryPollers | Количество предварительно порождённых экземпляров опросчиков истории. |
| StartHTTPAgentPollers | Количество предварительно порождённых экземпляров асинхронных опросчиков HTTP agent. |
| StartHTTPPollers | Количество предварительно порождённых экземпляров опросчиков HTTP. |
| StartIPMIPollers | Количество предварительно порождённых экземпляров опросчиков IPMI. |

| Parameter | Description |
|---|---|
| StartJavaPollers | Количество предварительно порождённых экземпляров опросчиков Java. |
| StartLLDProcessors | Количество предварительно порождённых экземпляров рабочих процессов низкоуровневого обнаружения (LLD). |
| StartODBCPollers | Количество предварительно порождённых экземпляров опросчиков ODBC. |
| StartPingers | Количество предварительно порождённых экземпляров ICMP-pinger. |
| StartPollersUnreachable | Количество предварительно порождённых экземпляров опросчиков для недоступных узлов сети (включая IPMI и Java). |
| StartPollers | Количество предварительно порождённых экземпляров опросчиков. |
| StartPreprocessors | Количество предварительно запущенных экземпляров рабочих процессов предварительной обработки. |
| StartProxyPollers | Количество предварительно порождённых экземпляров опросчиков для пассивных прокси. |
| StartReportWriters | Количество предварительно порождённых экземпляров генераторов отчётов. |
| StartSNMPPollers | Количество предварительно порождённых экземпляров асинхронных опросчиков SNMP. |
| StartSNMPTrapper | Если установлено значение 1, будет запущен процесс SNMP trapper. |
| StartTimers | Количество предварительно порождённых экземпляров таймеров. |
| StartTrappers | Количество предварительно порождённых экземпляров trapper. |
| StartVMwareCollectors | Количество предварительно порождённых экземпляров сборщиков VMware. |
| StatsAllowedIP | Список IP-адресов, разделённых запятыми, при необходимости в нотации CIDR, или DNS-имён внешних экземпляров Zabbix. Запрос статистики будет принят только с адресов, перечисленных здесь. |
| Timeout | Указывает, сколько времени ожидать (в секундах) установления соединения и обмена данными с прокси Zabbix, agent, веб-службой, а также для проверок SNMP (кроме элементов данных SNMP `walk[OID]` и `get[OID]`). |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA(ов) для проверки сертификата узла, используемые для зашифрованных соединений между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат сервера или цепочку сертификатов, используемые для зашифрованных соединений между компонентами Zabbix. |
| TLSCipherAll | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherAll13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherCert | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherCert13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherPSK | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSCipherPSK13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSCRLFile | Полный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных соединений между компонентами Zabbix. |
| TLSFrontendAccept | Какие входящие подключения принимать от веб-интерфейса. |
| TLSFrontendCertIssuer | Разрешённый издатель сертификата веб-интерфейса. |
| TLSFrontendCertSubject | Разрешённый субъект сертификата веб-интерфейса. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ сервера, используемый для зашифрованных соединений между компонентами Zabbix. |
| TLSListen | Управляет TLS на сокете trapper. |
| TmpDir | Временный каталог. |
| TrapperTimeout | Указывает тайм-аут в секундах для:<br>- получения исторических данных от прокси Zabbix;<br>- отправки данных конфигурации прокси Zabbix;<br>- выполнения глобального скрипта или удалённой команды на сервере Zabbix. |
| TrendCacheSize | Размер кэша трендов. |
| TrendFunctionCacheSize | Размер кэша функций трендов. |
| UnavailableDelay | Определяет, как часто узел сети проверяется на доступность в период недоступности. |
| UnreachableDelay | Определяет, как часто узел сети проверяется на доступность в период недостижимости. |
| UnreachablePeriod | Определяет, через сколько секунд недостижимости узел сети считается недоступным. |
| User | Сбросить привилегии до конкретного существующего пользователя в системе. |

| Parameter | Description |
|---|---|
| ValueCacheSize | Размер кэша значений истории. |
| Vault | Указывает поставщика vault. |
| VaultDBPath | Указывает расположение, откуда учётные данные базы данных должны извлекаться по ключам. |
| VaultPrefix | Пользовательский префикс для пути или запроса vault. |
| VaultTLSCertFile | Имя файла SSL-сертификата, используемого для аутентификации клиента. |
| VaultTLSKeyFile | Имя файла закрытого ключа SSL, используемого для аутентификации клиента. |
| VaultToken | Токен аутентификации HashiCorp vault. |
| VaultURL | HTTP[S]-URL сервера vault. |
| VMwareCacheSize | Размер общей памяти для хранения данных VMware. |
| VMwareFrequency | Задержка в секундах между сбором данных из одной службы VMware. |
| VMwarePerfFrequency | Задержка в секундах между получением статистики счётчиков производительности из одной службы VMware. |
| VMwareTimeout | Максимальное число секунд, которое сборщик vmware будет ждать ответа от службы VMware. |
| WebDriverURL | HTTP[S]-URL интерфейса WebDriver. |
| WebServiceURL | HTTP[S]-URL веб-службы Zabbix в формате <host:port>/report. |

Все параметры являются необязательными, если явно не указано, что параметр обязателен.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию демона, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды.
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Подробности параметров

AlertScriptsPath

Местоположение пользовательских скриптов оповещений (зависит от указанной при компиляции переменной *datadir*).

По умолчанию: `/usr/local/share/zabbix/alertscripts`

AllowRoot

Разрешение серверу запускаться под «root». Если отключено, а сервер запускается из-под «root», сервер вместо этого будет пытаться переключиться на пользователя «zabbix». Игнорируется, если сервер запускается под обычным пользователем.

По умолчанию: 0<br> Значения: 0 — не разрешено; 1 — разрешено

AllowSoftwareUpdateCheck

Разрешение веб-интерфейсу Zabbix получать с zabbix.com информацию об обновлениях ПО.

По умолчанию: 1<br> Значения: 0 — не разрешено; 1 — разрешено

AllowUnsupportedDBVersions

Разрешение серверу работать с неподдерживаемыми версиями баз данных

По умолчанию: 0<br> Значения: 0 — не разрешено; 1 — разрешено

CacheSize

Размер кэша конфигурации, в байтах. Размер разделяемой памяти (shared memory) для хранения данных узлов сети, элементов данных и триггеров.

По умолчанию: 32M<br> Диапазон: 128K-64G

CacheUpdateFrequency

Этот параметр определяет, как часто Zabbix будет выполнять процедуру обновления кэша конфигурации, в секундах. Смотрите также опции управления работой.

По умолчанию: 10<br> Диапазон: 1-3600

DBHost

Узел сети (или каталог сокета) базы данных Zabbix.

Для MySQL:

- `localhost` или пустая строка используют сокет UNIX-домена по умолчанию.

Для PostgreSQL:

- `localhost` разрешается через DNS (обычно в 127.0.0.1).
- Пустая строка использует сокет UNIX-домена по умолчанию.
- Путь (например, `/var/run/pgbouncer`) приводит к использованию сокета UNIX по этому пути.
- Список, разделенный запятыми, может содержать несколько значений `host:port`. Каждый узел сети проверяется по порядку, пока не будет установлено соединение с правом чтения и записи. Пример: `DBHost=localhost:5431,127.0.0.1:`

По умолчанию: `localhost`

DBName

Имя базы данных.

Обязательно: Да

DBPassword

Пароль базы данных. Закомментируйте эту строку, если пароль не используется.

DBPort

Порт узла сети базы данных Zabbix (см. DBHost).[3]

По умолчанию для MySQL: 3306

По умолчанию для PostgreSQL: 5432

Диапазон: 1024-65535

DBSchema

Имя схемы базы данных. Используется для PostgreSQL.

DBSocket

Путь к файлу сокета MySQL.[3]

DBUser

Пользователь базы данных.

DBTLSConnect

Установка этой опции в следующие значения принуждает использовать TLS подключение к базе данных:<br>*required* — подключаться с использованием TLS<br>*verify_ca* — подключаться с использованием TLS и проверкой сертификата<br>*verify_full* — подключаться с использованием TLS, проверкой сертификата и проверкой, что идентификатор базы данных, указанный параметром DBHost, совпадает с его сертификатом<br><br>Для MySQL, начиная с версии 5.7.11, и PostgreSQL поддерживаются следующие значения: `required`, `verify_ca`, `verify_full`. Для MariaDB, начиная с версии 10.2.6, поддерживаются значения `required` и `verify_full`.<br>По умолчанию значение на задано, а поведение зависит от конфигурации базы данных.

DBTLSCAFile

Полный путь к файлу, содержащему сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата базы данных.

Обязательное: нет (да, если DBTLSConnect выставлен в одно из: *verify_ca* или *verify_full*)

DBTLSCertFile

Полный путь к файлу, содержащему сертификат Zabbix сервера для аутентификации к базе данных.

DBTLSKeyFile

Полный путь к файлу, содержащему закрытый ключ (private key) для аутентификации к базе данных

DBTLSCipher

Список алгоритмов шифрования, разрешённых со стороны Zabbix сервера для протокола TLS до версии TLSv1.2 включительно. Поддерживается только для MySQL.

DBTLSCipher13

Список алгоритмов шифрования, разрешённых со стороны Zabbix сервера для протокола TLSv1.3. Поддерживается только для MySQL, начиная с версии 8.0.16.

DebugLevel

Задаёт уровень журналирования:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критичная информация<br>*2* — информация об ошибках<br>*3* — предупреждения<br>*4* — для отладки (записывается очень много информации)<br>*5* — расширенная отладка (записывается еще больше информации).<br>Смотрите также опции управления работой.

По умолчанию: 3<br> Диапазон: 0-5

EnableGlobalScripts

Разрешает глобальные скрипты на Zabbix сервере.<br> Примечание: Выполнение глобальных скриптов по умолчанию включено. Однако, для новых инсталляций, начиная с Zabbix 7.0, EnableGlobalScripts явно устанавливается в 0 (отключено).

По умолчанию: 1<br> Значения: 0 — отключено; 1 — включено

ExportDir

Директория для экспорта в режиме реального времени событий, истории и динамики изменений в JSON формате с разделением новой строкой. Если задано, активируется экспорт в режиме реального времени.

ExportFileSize

Максимальный размер каждого файла с экспортированными данными, в байтах. Используется для ротации, если задан параметр `ExportDir`.

По умолчанию: 1G<br> Диапазон: 1M-1G

ExportType

Список (через запятую) типов объектов (событий, истории и динамики изменений — events, history, trends) для экспорта в режиме реального времени (по умолчанию — все типы). Имеет смысл, только если задано значение ExportDir.<br>*Обратите внимание*, что если ExportType задано, а ExportDir — нет, то это ошибка конфигурации и сервер не запустится.

Пример для экспорта данных истории и динамики изменений:

```
ExportType=history,trends
```

Пример для экспорта только событий:

```
ExportType=events
```

ExternalScripts

Расположение внешних скриптов (зависит от указанной при компиляции переменной `datadir`).

По умолчанию: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Расположение утилиты fping6. Убедитесь, что владельцем бинарного файла fping6 является root и флаг SUID установлен. Оставьте пустым («Fping6Location=»), если ваша утилита fping поддерживает обработку адресов IPv6.

По умолчанию: `/usr/sbin/fping6`

FpingLocation

Расположение утилиты fping. Убедитесь, что владельцем бинарного файла fping6 является root и флаг SUID установлен.

По умолчанию: `/usr/sbin/fping`

FrontendAllowedIP

Список IP-адресов или диапазонов CIDR, разделенных запятыми, которым разрешено подключаться из веб-интерфейса. Подключение к веб-интерфейсу будет принято только с адресов, перечисленных здесь, если задан этот параметр. По умолчанию для запросов веб-интерфейса принимаются все подключения. Если поддержка IPv6 включена, то '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' считаются эквивалентными, а '::/0' разрешит любой IPv4- или IPv6-адрес. '0.0.0.0/0' можно использовать, чтобы разрешить любой IPv4-адрес.

Пример:

```
FrontendAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

HANodeName

Имя ноды кластера высокой доступности. Когда пусто, сервер работает в автономном режиме и создаётся нода с пустым именем.

HistoryCacheSize

Размер кэша истории, в байтах. Размер общей памяти для хранения данных истории.

По умолчанию: 16M<br> Диапазон: 128K-16G

HistoryIndexCacheSize

Размер кэша индекса истории, в байтах. Размер общей памяти для индексирования данных истории, хранящихся в кэше истории. Для кэша индексов требуется примерно 100 байт на хранение одного элемента данных.

По умолчанию: 4M<br> Диапазон: 128K-16G

HistoryStorageDateIndex

Включение предварительной обработки значений истории в хранилище истории для сохранения значений в разных индексах на основе даты.

По умолчанию: 0<br> Значения: 0 — отключено; 1 — включено

HistoryStorageURL

HTTP[S] URL хранилища истории. Этот параметр используется для настройки Elasticsearch.

HistoryStorageTypes

Список (через запятую) типов значений для отправки в хранилище истории. Этот параметр используется для настройки Elasticsearch.

По умолчанию: `uint,dbl,str,log,text`

HousekeepingFrequency

Этот параметр определяет, как часто Zabbix будет выполнять процедуру очистки базы (в часах). Автоматическая очистка базы данных удаляет из базы устаревшую информацию.<br>*Обратите внимание*: Для предотвращения перегрузки функции очистки (к примеру, когда периоды хранения данных истории и динамики изменения сильно уменьшены), не более чем 4 периода кратных HousekeepingFrequency часов устаревшей истории будет удалено за один цикл очистки по каждому элементу данных. Таким образом, если HousekeepingFrequency равен 1 часу, то за один цикл будет удалено не более чем 4 часа устаревшей истории (начиная с самой старой записи).<br>*На заметку*: Для снижения нагрузки на сервер запуск процесса очистки отложен на 30 минут после запуска сервера. Таким образом, если HousekeepingFrequency равен 1 часу, самая первая процедура очистки запустится через 30 минут после запуска сервера, и затем будет повторяться с задержкой в один час.<br>Имеется возможность отключить автоматическую очистку истории, указав HousekeepingFrequency равным 0. В этом случае процедуру очистки истории можно запустить только с помощью опции управления работой *housekeeper_execute* и периодом удаления устаревшей информации является 4-кратный период, начиная с последнего цикла удаления истории, но не менее чем 4 часа и не более 4 дней.<br>Смотрите также опции административных функций.

По умолчанию: 1<br> Диапазон: 0-24

Include

Вы можете включить в файл конфигурации отдельные файлы или все файлы в каталоге. Чтобы включить только соответствующие файлы в указанном каталоге, для сопоставления с шаблоном поддерживается подстановочный знак звёздочки. Смотрите специальные заметки по поводу ограничений.

Пример:

```
Include=/      /   / /    /    /*.conf
```

JavaGateway

IP адрес (или имя хоста) Zabbix Java gateway. Требуется, только если запущены поллеры Java.

JavaGatewayPort

Порт, который слушает Zabbix Java gateway.

По умолчанию: 10052<br> Диапазон: 1024-32767

ListenBacklog

Максимальное количество соединений в состоянии ожидания в очереди TCP.<br>Значение по умолчанию является зашитой в коде константой, которая зависит от системы.<br>Максимально поддерживаемое значение зависит от системы, слишком большие значения могу быть молча урезаны до «указанного в реализации максимума» (’implementation-specified maximum’).

По умолчанию: SOMAXCONN<br> Диапазон: 0 – INT_MAX

ListenIP

Список (через запятую) IP адресов, которые должен слушать траппер.<br>Если этот параметр не указан, траппер будет слушать все сетевые интерфейсы.

По умолчанию: 0.0.0.0

ListenPort

Порт, который слушает траппер.

По умолчанию: 10051<br> Диапазон: 1024-32767

LoadModule

Модуль, загружаемый при запуске сервера. Модули используются для расширения функциональности сервера. Либо модуль должен находиться в папке, указанной параметром LoadModulePath, либо путь должен быть указан перед именем модуля. Если указанный путь является абсолютным (начинается с «/»), то параметр LoadModulePath игнорируется.<br>Форматы:<br>LoadModule=<module.so><br>LoadModule=<путь/module.so><br>LoadModule= </абс_путь/modul
добавлять несколько параметров LoadModule.

LoadModulePath

Абсолютный путь к папке с серверными модулями. Значение по умолчанию зависит от опций компиляции.

LogFile

Имя файла журнала.

Обязательное: Да, если LogType выставлен в *file*; в противном случае нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключение автоматической ротации журнала.<br>*Примечание*: Если лимит размера файла достигнут, а ротация по каким-либо причинам не удалась, существующий файл журнала усекается и начинается заново.

По умолчанию: 1<br> Диапазон: 0-1024<br> Обязательное: да, если LogType выставлен в *file*; в противном случае нет

LogSlowQueries

Определяет, как долго могут выполняться запросы к базе данных до того как они будут записаны в журнал (в миллисекундах).<br>0 — не журналировать медленные запросы.<br>Эта опция активируется, начиная с DebugLevel=3.

По умолчанию: 0<br> Диапазон: 0-3600000

LogType

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*system* — запись журнала в syslog;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: file

MaxConcurrentChecksPerPoller

Максимальное количество асинхронных проверок, которые могут быть выполнены за раз каждым поллером HTTP агента, поллером агента или поллером SNMP. Смотрите StartHTTPAgentPollers, StartAgentPollers и StartSNMPPollers.

По умолчанию: 1000<br> Диапазон: 1-1000

MaxHousekeeperDelete

Удалять не более «MaxHousekeeperDelete» строк (в соответствии с [имя_таблицы], [поле], [значение]) на одну задачу за один цикл автоматической очистки базы данных.<br>Если значение равно 0, то тогда ограничение не используется вовсе. В этом случае вы должны знать, что вы делаете, чтобы не перегрузить базу данных.[2]<br>Этот параметр применяется только для удаления данных (истории и динамики изменений), оставленных уже удалёнными элементами данных.

По умолчанию: 5000<br> Диапазон: 0-1000000

NodeAddress

IP адрес или имя хоста с опциональным номером порта, чтобы переопределить, каким образом веб-интерфейс должен подключаться к серверу.<br>Формат: <адрес>[:порт]<br><br>Если IP или имя хоста не заданы, будет взято значение параметра ListenIP. Если и ListenIP на задано, будет использовано значение `localhost`.<br>Если не указан порт, будет взято значение параметра ListenPort. Если и ListenPort не задано, будет использовано значение 10051.<br><br>Эта опция может быть переопределена адресом, указанным в конфигурации веб-интерфейса.<br><br>Смотрите также: параметр HANodeName; Включение высокой доступности.

По умолчанию: `localhost:10051`

PidFile

Имя PID файла.

По умолчанию: `/tmp/zabbix_server.pid`

ProblemHousekeepingFrequency

Определяет, как часто Zabbix будет удалять проблемы для удалённых триггеров (в секундах).

По умолчанию: 60<br> Диапазон: 1-3600

ProxyConfigFrequency

Определяет, как часто Zabbix сервер будет отправлять данные конфигурации на Zabbix прокси (в секундах). Используется только с прокси, которые работают в пассивном режиме.

По умолчанию: 10<br> Диапазон: 1-604800

ProxyDataFrequency

Определяет, как часто Zabbix сервер будет запрашивать данные истории с Zabbix прокси (в секундах). Используется только с прокси, которые работают в пассивном режиме.

По умолчанию: 1<br> Диапазон: 1-3600

ServiceManagerSyncFrequency

Определяет, как часто Zabbix будет синхронизировать конфигурацию менеджера услуг (в секундах).

По умолчанию: 60<br> Диапазон: 1-3600

SMSDevices

Список файлов модема, разделенных запятыми, которым разрешено использовать сервер Zabbix.<br>Отправка SMS невозможна, если этот параметр не задан.

Пример:

```
SMSDevices=/dev/ttyUSB0,/dev/ttyUSB1
```

SNMPTrapperFile

Временный файл, используемый для передачи данных серверу от демона SNMP trap.<br>Должен быть таким же, как и в файле zabbix_trap_receiver.pl или файле конфигурации SNMPTT.

По умолчанию: `/tmp/zabbix_traps.tmp`

SocketDir

Папка для хранения IPC сокетов, которые используются внутренними службами Zabbix.

По умолчанию: `/tmp`

SourceIP

Локальный IP адрес для:

- исходящих подключений к Zabbix прокси и Zabbix агенту;
- безагентных подключений (VMware, SSH, JMX, SNMP, Telnet и простые проверки);
- подключений HTTP агента;
- запросов HTTP из JavaScript для скриптовых элементов данных;
- запросов HTTP из JavaScript во время предобработки;
- отсылки почтовых уведомлений (соединения к серверу SMTP);
- оповещений вебхук (HTTP соединения из JavaScript);
- подключений к Хранилищу.

SSHKeyLocation

Местоположение открытых и закрытых ключей для проверок и действий SSH.

SSLCertLocation

Местоположение файлов клиентских сертификатов SSL для аутентификации клиента.<br>Этот параметр используется только в веб-мониторинге.

SSLKeyLocation

Местоположение файлов закрытых ключей SSL для аутентификации клиента.<br>Этот параметр используется только в веб-мониторинге.

SSLCALocation

Переопределение местоположения файлов удостоверяющих центров (certificate authority, CA) для верификации SSL сертификатов сервера. Если не задано, будет использоваться общесистемная папка.<br>Обратите внимание, что значение этого параметра задаст опцию libcurl CURLOPT_CAPATH. Для libcurl версий до 7.42.0 эта опция имеет эффект, только если libcurl скомпилирован для использования OpenSSL. Для получения более подробной информации смотрите веб-страницу cURL [en].<br>Этот параметр используется в веб-мониторинге и при аутентификации SMTP.

StartAgentPollers

Количество экземпляров запускаемых при старте асинхронных процессов поллеров Zabbix агентов (agent poller). Смотрите MaxConcurrentChecksPerPoller.

По умолчанию: 1<br> Диапазон: 0-1000

StartAlerters

Количество экземпляров запускаемых при старте процессов оповещений (alerter).

По умолчанию: 3<br> Диапазон: 1-100

StartBrowserPollers

Количество экземпляров запускаемых при старте процессов поллеров браузерных элементов данных (browser poller).

По умолчанию: 1<br> Диапазон: 0-1000

StartConnectors

Количество экземпляров запускаемых при старте рабочих процессов коннектора (connector worker). Процесс менеджера коннекторов (connector manager) автоматически запускается при старте рабочего процесса коннектора.

По умолчанию: 0<br> Диапазон: 0-1000

StartDBSyncers

Количество экземпляров запускаемых при старте процессов записи истории в базу данных (history syncer).<br>*Примечание*: Будьте осторожны при изменении этого значения, его увеличение может принести больше вреда, чем пользы. Грубо, значения по умолчанию должно быть достаточно, чтобы обслуживать до 4000 NVPS.

По умолчанию: 4<br> Диапазон: 1-100

StartDiscoverers

Количество экземпляров запускаемых при старте процессов сетевого обнаружения (discoverer)[1].

По умолчанию: 5<br> Диапазон: 0-1000

StartEscalators

Количество экземпляров запускаемых при старте процессов эскалации действий (escalator).

По умолчанию: 1<br> Диапазон: 1-100

StartHistoryPollers

Количество экземпляров запускаемых при старте процессов поллеров истории (history poller).<br>Требуется только для вычисляемых проверок.

По умолчанию: 5<br> Диапазон: 0-1000

StartHTTPAgentPollers

Количество экземпляров запускаемых при старте асинхронных процессов поллеров HTTP агента (http agent poller). Смотрите MaxConcurrentChecksPerPoller.

По умолчанию: 1<br> Диапазон: 0-1000

StartHTTPPollers

Количество экземпляров запускаемых при старте процессов поллеров HTTP (HTTP pollers)[1].

По умолчанию: 1<br> Диапазон: 0-1000

StartIPMIPollers

Количество экземпляров запускаемых при старте процессов поллеров IPMI (ipmi poller)

По умолчанию: 0<br> Диапазон: 0-1000

StartJavaPollers

Количество экземпляров запускаемых при старте процессов поллеров Java (java poller)[1].

По умолчанию: 0<br> Диапазон: 0-1000

StartLLDProcessors

Количество экземпляров запускаемых при старте рабочих процессов низкоуровневого обнаружения (lld worker)[1].<br>Процесс LLD manager автоматически запускается при старте процесса LLD worker.

По умолчанию: 2<br> Диапазон: 1-100

StartODBCPollers

Количество экземпляров запускаемых при старте процессов поллеров ODBC (odbc poller)[1].

По умолчанию: 1<br> Диапазон: 0-1000

StartPingers

Количество экземпляров запускаемых при старте процессов пингеров ICMP (icmp pinger)[1].

По умолчанию: 1<br> Диапазон: 0-1000

StartPollersUnreachable

Количество экземпляров запускаемых при старте процессов поллеров недоступных устройств (unreachable poller) (в том числе IPMI и Java)[1].<br>По крайней мере один поллер недоступных устройств должен быть запущен, если запускаются обычные поллеры, IPMI или Java.

По умолчанию: 1<br> Диапазон: 0-1000

StartPollers

Количество экземпляров запускаемых при старте процессов поллеров (poller)[1].

По умолчанию: 5<br> Диапазон: 0-1000

StartPreprocessors

Количество предварительно запущенных экземпляров потоков предварительной обработки worker[1] следует задавать не меньше количества доступных ядер CPU. Следует задать больше worker, если предварительная обработка не ограничена CPU и содержит много сетевых запросов.

По умолчанию: 16<br> Диапазон: 1-1000

StartProxyPollers

Количество экземпляров запускаемых при старте процессов поллеров пассивных прокси (proxy poller)[1].

По умолчанию: 1<br> Диапазон: 0-250

StartReportWriters

Количество экземпляров запускаемых при старте процессов генерации отчётов (report writer).<br>Если выставлено в 0, генерация отчётов по расписанию отключена.<br>Процесс менеджера задач генерации отчётов по расписанию (report manager) автоматически запускается при старте процесса report writer.

По умолчанию: 0<br> Диапазон: 0-100

StartSNMPPollers

Количество экземпляров запускаемых при старте асинхронных процессов поллеров SNMP (snmp poller). Смотрите MaxConcurrentChecksPerPoller.

По умолчанию: 1<br> Диапазон: 0-1000

StartSNMPTrapper

Если выставлено в 1, процесс траппера SNMP (snmp trapper) будет запущен.

По умолчанию: 0<br> Диапазон: 0-1

StartTimers

Количество экземпляров запускаемых при старте процессов timer.<br>Процессы timer обрабатывают периоды обслуживания.

По умолчанию: 1<br> Диапазон: 1-1000

StartTrappers

Количество экземпляров запускаемых при старте процессов трапперов (trapper)[1].<br>Трапперы принимают входящие соединения от Zabbix sender'а, активных агентов и активных прокси.

По умолчанию: 5<br> Диапазон: 0-1000

StartVMwareCollectors

Количество экземпляров запускаемых при старте процессов сборщиков данных VMware (vmware collector).

По умолчанию: 0<br> Диапазон: 0-250

StatsAllowedIP

Список (через запятую) IP адресов, возможно в нотации CIDR, или DNS имён внешних экземпляров Zabbix. Запрос статистики будет приниматься только с перечисленных здесь адресов. Если параметр не задан, никакие запросы статистики приниматься не будут.<br>Если поддержка IPv6 включена, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» рассматриваются как одинаковые, а «::/0» разрешит любой адрес IPv4 или IPv6. Можно использовать «0.0.0.0/0» , чтобы разрешить любой адрес IPv4.

Пример:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Timeout

Указывает, как долго (в секундах) следует ожидать установления соединения и обмена данными с Zabbix прокси, агентом, веб-сервисом, а также для старых проверок SNMP (только OID-номер либо строка).<br>

Этот параметр определяет продолжительность различных коммуникационных операций:

- выполнение удалённой команды на Zabbix агенте
- выполнение команды SSH/Telnet
- запросы к веб-сервису Zabbix
- тайм-аут связи для запроса тестирования способа оповещения и метода `history.push`
- перепланирование элементов данных, когда интерфейс IPMI становится недоступным
- отправка ответа Zabbix прокси при сбое обмена данными из-за проблем с правами или шифрованием
- крайний срок для асинхронных сокетов IPC и опций управления работой
- соединения JMX
- получение статистики с удалённого Zabbix прокси или сервера
- отправка ответов на веб-интерфейс Zabbix
- запросы асинхронного поллера DNS
- ответы на сообщения heartbeat активных проверок
- получение данных (значений) Zabbix агента от активных агентов
- получение данных от утилиты Zabbix sender
- ответ, когда траппер не может обработать запрос
- отправка Zabbix агенту списка активных проверок

Этот тайм-аут **не** будет использоваться для тех проверок, которые имеют параметры гибкого тайм-аута, настраиваемые в веб-интерфейсе (на глобальном уровне, на уровне прокси-сервера или на уровне отдельных элементов данных). Например, элементы данных SNMP `walk[OID]` и `get[OID]` используют настраиваемый тайм-аут из веб-интерфейса; а старые проверки SNMP по-прежнему используют значение тайм-аута сервера.

По умолчанию: 3<br> Диапазон: 1-30

TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Абсолютный путь к файлу, который содержит сертификат или цепочку сертификатов сервера, используется для зашифрованных соединений между компонентами Zabbix.

TLSCipherAll

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов и общего ключа (PSK).

Пример:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов и общего ключа (PSK).

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Пример для OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов.

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Пример для OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов.

TLSCipherPSK

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе общего ключа (PSK).

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Пример для OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе общего ключа (PSK).

Пример:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCRLFile

Абсолютный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSFrontendAccept

Какие входящие подключения принимать от веб-интерфейса.

Можно указать несколько значений, разделяя их запятыми:

- unencrypted - принимать подключения без шифрования.
- cert - принимать подключения, защищенные с помощью TLS и сертификата.

По умолчанию: `unencrypted`

TLSFrontendCertIssuer

Допустимый издатель сертификата веб-интерфейса.

TLSFrontendCertSubject

Допустимый subject сертификата веб-интерфейса.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ сервера (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSListen

Управляет TLS на сокете trapper.

Поддерживаемые значения:

- required - принимать только TLS-соединения

TmpDir

Папка с временными данными.

По умолчанию: `/tmp`

TrapperTimeout

Указывает время ожидания в секундах для:

- получения данных истории от Zabbix прокси;
- отправки данных конфигурации на Zabbix прокси;
- выполнения глобального скрипта или выполнения удалённой команды на сервере Zabbix.

По умолчанию: 300<br> Диапазон: 1-300

TrendCacheSize

Размер кэша трендов в байтах.<br>Размер общей памяти для хранения данных трендов.

По умолчанию: `4M`<br> Диапазон: 128K-16G

TrendFunctionCacheSize

Размер кэша функций динамики изменений, в байтах.<br>Размер разделяемой памяти для хранения данных вычисляемых функций динамики изменений.

По умолчанию: `4M`<br> Диапазон: 128K-2G

UnavailableDelay

Определяет, как часто узел сети будет проверяться на доступность в период его недоступности, в секундах.

По умолчанию: 60<br> Диапазон: 1-3600

UnreachableDelay

Определяет, как часто узел сети будет проверяться на доступность в период его недостижимости, в секундах.

По умолчанию: 15<br> Диапазон: 1-3600

UnreachablePeriod

Определяет, через сколько секунд недостижимости узел сети считается недоступным.

По умолчанию: 45<br> Диапазон: 1-3600

User

Использование привилегий указанного, существующего пользователя системы.<br>Имеет эффект, только если запускается под «root» и AllowRoot отключён.

По умолчанию: `zabbix`

ValueCacheSize

Размер кэша для хранения значений истории, в байтах.<br>Разделяемая память для кэширования запросов к данным истории элементов данных.<br>Значение 0 отключит кэш значений (не рекомендуется).<br>В случае, если кэш значений исчерпает разделяемую память, предупреждающее сообщение будет записано в журнал сервера каждые 5 минут.

По умолчанию: 8M<br> Диапазон: 0,128K-64G

Vault

Указывает поставщика хранилища:<br>*HashiCorp* — HashiCorp KV Secrets Engine version 2<br>*CyberArk* — CyberArk Central Credential Provider<br>Должен совпадать с поставщиком хранилища, указанном в веб-интерфейсе.

По умолчанию: `HashiCorp`

VaultDBPath

Путь к хранилищу или запрос, в зависимости от хранилища, откуда по ключам будут извлекаться данные для аутентификации к базе данных.

Ключи, используемые для HashiCorp, — «password» и «username».

Пример пути с VaultPrefix=/v1/secret/data/zabbix/:

```
database
```

Пример пути без VaultPrefix:

```
secret/zabbix/database
```

Ключи, используемые для CyberArk, — «Content» и «UserName».

Пример:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_server_database
```

Этот параметр может использоваться, только если параметры DBUser и DBPassword не указаны.

VaultPrefix

Пользовательский префикс для пути или запроса Хранилища, в зависимости от Хранилища. Если не указано, будут использоваться наиболее подходящие значения по умолчанию.<br>Обратите внимание, что если VaultPrefix не указан, то для HashiCorp после точки монтирования автоматически добавляется «data».

Пример префикса для Hashicorp:

```
v1/secret/data/zabbix/
```

Пример префикса для Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Имя файла SSL сертификата, используемого для аутентификации клиента.<br>Файл сертификата должен быть в формате PEM1.<br>Если файл сертификата также содержит закрытый ключ, оставьте поле файла ключа SSL пустым.<br>Папка, содержащая этот файл, указывается параметром конфигурации SSLCertLocation.<br>Эту опцию можно опустить, но она рекомендуется для хранилища CyberArkCCP.

VaultTLSKeyFile

Имя файла закрытого ключа SSL, используемого для аутентификации клиента.<br> Файл закрытого ключа должен быть в формате PEM1.<br> Каталог, содержащий этот файл, указывается с помощью параметра конфигурации SSLKeyLocation.<br>Этот параметр можно не указывать, но для хранилища CyberArkCCP его использование рекомендуется.

VaultToken

Токен аутентификации Хранилища HashiCorp, который должен быть сгенерирован исключительно для сервера Zabbix с разрешениями только на чтение на пути, указанные в макросах Хранилища, и разрешениями только на чтение на пути, указанные в необязательном параметре конфигурации VaultDBPath.<br>Если одновременно заданы и параметр VaultToken, и переменная окружения VAULT_TOKEN, то это считается ошибкой.

Обязательное: Да, если Vault задан в значение *HashiCorp*; в противном случае — нет

VaultURL

HTTP[S] URL сервера Хранилища. Если SSLCALocation не указан, будет использоваться общесистемная папка сертификатов удостоверяющих центров (certificate authority, CA).

По умолчанию: `https://127.0.0.1:8200`

VMwareCacheSize

Размер разделяемой памяти для хранения данных VMware.<br>Для мониторинга использования кэша VMware можно использовать внутренние проверки VMware zabbix[vmware,buffer,...] (смотрите раздел о внутренних проверках).<br>Обратите внимание, что разделяемая память не выделяется, если не настроено на запуск ни одного экземпляра коллекторов VMware.

По умолчанию: 8M<br> Диапазон: 256K-2G

VMwareFrequency

Задержка в секундах между сбором данных с одной службы VMware.<br>Эту задержку необходимо установить в наименьшее из значений интервала обновления у любого элемента данных мониторинга VMware.

По умолчанию: 60<br> Диапазон: 10-86400

VMwarePerfFrequency

Задержка в секундах между получением статистики по счётчикам производительности с одной службы VMware. Эту задержку необходимо установить в наименьшее из значений интервала обновления у любого элемента данных мониторинга VMware, который использует счётчики производительности VMware.

По умолчанию: 60<br> Диапазон: 10-86400

VMwareTimeout

Максимальное количество секунд, которое коллектор vmware потратит на ожидание ответа от службы VMware (vCenter или ESX гипервизора).

По умолчанию: 10<br> Диапазон: 1-300

WebServiceURL

HTTP[S] URL к веб-сервису Zabbix в формате `http[s]://  :  /report`.

Пример:

```
WebServiceURL=http://localhost:10053/report
```

Примечание: схема (`http://`) может быть опущена только для не-TLS соединений (HTTP); если настроено TLS, обязательно должно быть использовано `https://`.

WebDriverURL

HTTP[S] URL интерфейса WebDriver'а.

Пример (используется с автономным сервером Selenium WebDriver):

```
WebDriverURL=http://localhost:4444
```

Примечания

[1] Обратите внимание, что слишком большое количество процессов сбора данных (поллеров, поллеров недоступных устройств, поллеров ODBC, поллеров HTTP, поллеров Java, пингеров, трапперов, поллеров прокси) вместе с IPMI менеджером, SNMP траппером, рабочими процессами предварительной обработки и обнаружения, могут исчерпать ограничение количества файловых дескрипторов на один процесс для менеджера предварительной обработки.

**Warning:**
Исчерпание лимита файловых дескрипторов приведёт к остановке сервера Zabbix, обычно вскоре после запуска, но иногда может занять более длительное время. Чтобы избежать таких проблем, просмотрите файл конфигурации сервера Zabbix, чтобы оптимизировать количество одновременных проверок и процессов. Кроме того, при необходимости убедитесь, что лимит файловых дескрипторов установлен достаточно высоким, путём проверки и настройки системных ограничений.

[2] При удалении большого количества элементов данных возрастает нагрузка на базу данных, поскольку процессу очистки базы данных (housekeeper) потребуется удалить все данные истории, относящиеся к этим элементам данных. Для примера, если нам нужно удалить всего 1 прототип элемента данных из шаблона, но этот шаблон связан с 50 узлами сети и на каждом из них на основе этого прототипа сгенерировано по 100 реальных элементов данных, то потребуется удалить суммарно 5000 элементов данных (1*50*100). Если для параметра MaxHousekeeperDelete задано значение 500 (MaxHousekeeperDelete=500), то для удаляемых элементов данных процесс housekeeper должен будет за один цикл удалять из таблиц истории и динамики изменений до 2500000 значений (5000*500).

[3] В конфигурации сервера параметры DBSocket и DBPort являются взаимоисключающими. Укажите только один из них или оставьте оба незаданными.

## 2. Zabbix прокси

Обзор

Параметры, поддерживаемые файлом конфигурации прокси Zabbix (zabbix_proxy.conf), перечислены в этом разделе.

Параметры перечислены без дополнительной информации. Нажмите на параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| AllowRoot | Разрешить прокси запускаться от имени 'root'. |
| AllowUnsupportedDBVersions | Разрешить прокси работать с неподдерживаемыми версиями базы данных. |
| CacheSize | Размер кэша конфигурации. |
| ConfigFrequency | Этот параметр **устарел** (используйте вместо него ProxyConfigFrequency).<br>Как часто прокси получает данные конфигурации от сервера Zabbix, в секундах. |
| DataSenderFrequency | Прокси будет отправлять собранные данные на сервер каждые N секунд. |
| DBHost | Узел сети (или каталог сокета) базы данных Zabbix. |
| DBName | Имя базы данных или путь к файлу базы данных для SQLite3. |
| DBPassword | Пароль базы данных. |
| DBPort | Порт узла сети базы данных Zabbix. |
| DBSchema | Имя схемы базы данных. Используется для PostgreSQL. |
| DBSocket | Путь к файлу сокета MySQL. |
| DBUser | Пользователь базы данных. |
| DBTLSConnect | Указание этого параметра со значением из списка принудительно включает использование TLS-соединения с базой данных. |
| DBTLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA(ов) для проверки сертификата базы данных. |
| DBTLSCertFile | Полный путь к файлу, содержащему сертификат прокси Zabbix для аутентификации в базе данных. |
| DBTLSKeyFile | Полный путь к файлу, содержащему закрытый ключ для аутентификации в базе данных. |
| DBTLSCipher | Список шифров, которые прокси Zabbix разрешает для протоколов TLS вплоть до TLS v1.2.<br>Поддерживается только для MySQL. |
| DBTLSCipher13 | Список наборов шифров, которые прокси Zabbix разрешает для протокола TLS v1.3.<br>Поддерживается только для MySQL, начиная с версии 8.0.16. |
| DebugLevel | Уровень отладки. |
| EnableRemoteCommands | Разрешены ли удаленные команды от сервера Zabbix. |
| ExternalScripts | Расположение внешних скриптов. |
| Fping6Location | Расположение fping6. |
| FpingLocation | Расположение fping. |
| HistoryCacheSize | Размер кэша истории. |
| HistoryIndexCacheSize | Размер кэша индекса истории. |
| Hostname | Уникальное, чувствительное к регистру имя прокси. |
| HostnameItem | Элемент данных, используемый для задания Hostname, если он не определен. |
| HousekeepingFrequency | Как часто Zabbix будет выполнять процедуру очистки, в часах. |
| Include | В файл конфигурации можно включать отдельные файлы или все файлы в каталоге. |
| JavaGateway | IP-адрес (или имя узла сети) Java gateway Zabbix. |
| JavaGatewayPort | Порт, на котором Java gateway Zabbix ожидает подключения. |
| ListenBacklog | Максимальное число ожидающих подключений в очереди TCP. |
| ListenIP | Список IP-адресов, разделенных запятыми, на которых должен слушать trapper. |
| ListenPort | Порт прослушивания для trapper. |
| LoadModule | Модуль, который нужно загрузить при запуске прокси. |
| LoadModulePath | Полный путь к расположению модулей прокси. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogRemoteCommands | Включить запись выполненных команд оболочки как предупреждений. |
| LogSlowQueries | Через какое время запрос к базе данных будет записан в журнал. |
| LogType | Тип вывода журнала. |
| MaxConcurrentChecksPerPoller | Максимальное число асинхронных проверок, которые могут выполняться одновременно каждым HTTP agent poller, agent poller или SNMP poller. |
| PidFile | Имя файла PID. |
| ProxyBufferMode | Определяет механизм хранения данных истории, обнаружения и авто-регистрации (disk/memory/hybrid). |
| ProxyConfigFrequency | Как часто прокси получает данные конфигурации от сервера Zabbix, в секундах. |

| Parameter | Description |
|---|---|
| ProxyLocalBuffer | Прокси будет хранить данные локально в течение N часов, даже если данные уже синхронизированы с сервером. |
| ProxyMemoryBufferAge | Максимальный возраст данных в буфере памяти прокси, в секундах. |
| ProxyMemoryBufferSize | Размер кэша в общей памяти для собранных данных истории, обнаружения и авто-регистрации. |
| ProxyMode | Режим работы прокси (active/passive). |
| ProxyOfflineBuffer | Прокси будет хранить данные в течение N часов в случае отсутствия связи с сервером Zabbix. |
| Server | Если ProxyMode установлен в active mode: IP-адрес сервера Zabbix или DNS-имя (address:port) либо кластер (address:port;address2:port), откуда получать данные конфигурации и куда отправлять данные.<br>Если ProxyMode установлен в passive mode: Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имена сервера Zabbix. |
| SNMPTrapperFile | Временный файл, используемый для передачи данных от демона SNMP trap к прокси. |
| SocketDir | Каталог для хранения IPC-сокетов, используемых внутренними службами Zabbix. |
| SourceIP | Исходный IP-адрес. |
| SSHKeyLocation | Расположение открытых и закрытых ключей для проверок и действий SSH. |
| SSLCertLocation | Расположение файлов клиентского сертификата SSL для аутентификации клиента. |
| SSLKeyLocation | Расположение файлов закрытого ключа SSL для аутентификации клиента. |
| SSLCALocation | Переопределяет расположение файлов центра сертификации (CA) для проверки сертификата сервера SSL. |
| StartAgentPollers | Число предварительно порожденных экземпляров асинхронных agent pollers Zabbix. |
| StartBrowserPollers | Число предварительно порожденных экземпляров pollers для элементов данных browser. |
| StartDBSyncers | Число предварительно порожденных экземпляров синхронизаторов истории. |
| StartDiscoverers | Число предварительно порожденных экземпляров обработчиков обнаружения. |
| StartHTTPAgentPollers | Число предварительно порожденных экземпляров асинхронных HTTP agent pollers. |
| StartHTTPPollers | Число предварительно порожденных экземпляров HTTP pollers. |
| StartIPMIPollers | Число предварительно порожденных экземпляров IPMI pollers. |
| StartJavaPollers | Число предварительно порожденных экземпляров Java pollers. |
| StartODBCPollers | Число предварительно порожденных экземпляров ODBC pollers. |
| StartPingers | Число предварительно порожденных экземпляров ICMP pingers. |
| StartPollersUnreachable | Число предварительно порожденных экземпляров pollers для недоступных узлов сети (включая IPMI и Java). |
| StartPollers | Число предварительно порожденных экземпляров pollers. |
| StartPreprocessors | Число предварительно запущенных экземпляров обработчиков предварительной обработки. |
| StartSNMPPollers | Число предварительно порожденных экземпляров асинхронных SNMP pollers. |
| StartSNMPTrapper | Если установлено значение 1, будет запущен процесс SNMP trapper. |
| StartTrappers | Число предварительно порожденных экземпляров trappers. |
| StartVMwareCollectors | Число предварительно порожденных экземпляров сборщиков VMware. |
| StatsAllowedIP | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имена внешних экземпляров Zabbix. Запрос статистики будет принят только с адресов, перечисленных здесь. |
| Timeout | Определяет, сколько времени ждать (в секундах) установления соединения и обмена данными с прокси Zabbix, агентом, веб-службой, а также для SNMP-проверок (кроме элементов данных SNMP `walk[OID]` и `get[OID]`). |
| TLSAccept | Какие входящие подключения от сервера Zabbix принимать. |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA(ов) для проверки сертификата узла, используемые для зашифрованных соединений между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат сервера или цепочку сертификатов, используемые для зашифрованных соединений между компонентами Zabbix. |
| TLSCipherAll | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherAll13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherCert | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherCert13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |

| Parameter | Description |
|---|---|
| TLSCipherPSK | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSCipherPSK13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSConnect | Как прокси должен подключаться к серверу Zabbix. |
| TLSCRLFile | Полный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных соединений между компонентами Zabbix. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ прокси, используемый для зашифрованных соединений между компонентами Zabbix. |
| TLSListen | Управляет TLS на сокете trapper. |
| TLSPSKFile | Полный путь к файлу, содержащему предварительно общий ключ прокси, используемый для зашифрованных соединений с сервером Zabbix. |
| TLSPSKIdentity | Строка идентификатора предварительно общего ключа, используемая для зашифрованных соединений с сервером Zabbix. |
| TLSServerCertIssuer | Разрешенный издатель сертификата сервера. |
| TLSServerCertSubject | Разрешенный субъект сертификата сервера. |
| TmpDir | Временный каталог. |
| TrapperTimeout | Определяет тайм-аут в секундах для:<br>- получения данных конфигурации от сервера Zabbix;<br>- выполнения глобального скрипта или удаленной команды. |
| UnavailableDelay | Как часто узел сети проверяется на доступность в период недоступности. |
| UnreachableDelay | Как часто узел сети проверяется на доступность в период недостижимости. |
| UnreachablePeriod | Через сколько секунд недостижимости считать узел сети недоступным. |
| User | Сбросить привилегии до конкретного существующего пользователя в системе. |
| Vault | Поставщик vault. |
| VaultDBPath | Расположение, откуда учетные данные базы данных должны извлекаться по ключам. |
| VaultPrefix | Пользовательский префикс для пути или запроса vault. |
| VaultTLSCertFile | Имя файла сертификата SSL, используемого для аутентификации клиента. |
| VaultTLSKeyFile | Имя файла закрытого ключа SSL, используемого для аутентификации клиента. |
| VaultToken | Токен аутентификации HashiCorp vault. |
| VaultURL | URL HTTP[S] сервера vault. |
| VMwareCacheSize | Размер общей памяти для хранения данных VMware. |
| VMwareFrequency | Задержка в секундах между сбором данных от одной службы VMware. |
| VMwarePerfFrequency | Задержка в секундах между получением статистики счетчиков производительности от одной службы VMware. |
| VMwareTimeout | Максимальное число секунд, которое сборщик vmware будет ждать ответа от службы VMware. |
| WebDriverURL | URL HTTP[S] интерфейса WebDriver. |

Все параметры являются необязательными, если явно не указано, что параметр обязателен.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию демона, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды.
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Подробности параметров

AllowRoot

Разрешение прокси серверу запускаться под «root». Если отключено, а прокси запускается из-под «root», прокси сервер вместо этого будет пытаться переключиться на пользователя «zabbix». Игнорируется, если запускается под обычным пользователем.

По умолчанию: 0<br> Значения: 0 — не разрешено; 1 — разрешено

AllowUnsupportedDBVersions

Разрешение прокси серверу работать с неподдерживаемыми версиями баз данных

По умолчанию: 0<br> Значения: 0 — не разрешено; 1 — разрешено

CacheSize

Размер кэша конфигурации, в байтах. Размер разделяемой памяти (shared memory) для хранения данных узлов сети и элементов данных.

По умолчанию: 32M<br> Диапазон: 128K-64G

ConfigFrequency

Этот параметр **устарел** (используйте вместо него ProxyConfigFrequency).<br>Как часто прокси забирает данные конфигурации с сервера Zabbix (в секундах).<br>Параметр активного прокси. Игнорируется для пассивных прокси (смотрите параметр ProxyMode).

По умолчанию: 3600<br> Диапазон: 1-604800

DataSenderFrequency

Прокси будет отсылать собранную информацию на сервер каждые N секунд. Обратите внимание, что активный прокси всё ещё будет опрашивать Zabbix сервер каждую секунду на предмет заданий удалённых команд.<br>Параметр активного прокси. Игнорируется для пассивных прокси (смотрите параметр ProxyMode).

По умолчанию: 1<br> Диапазон: 1-3600

DBHost

Узел сети (или каталог сокета) базы данных Zabbix.

Для MySQL:

- localhost или пустая строка используют сокет UNIX-домена по умолчанию.

Для PostgreSQL:

- localhost разрешается через DNS (обычно в 127.0.0.1).
- Пустая строка использует сокет UNIX-домена по умолчанию.
- Путь (например, /var/run/pgbouncer) приводит к использованию сокета UNIX по этому пути.
- Список, разделенный запятыми, может содержать несколько значений host:port. Каждый узел сети проверяется по порядку, пока не будет установлено соединение с правом чтения и записи. Пример: DBHost=localhost:5431,127.0.0.1:

По умолчанию: localhost

DBName

Имя базы данных или путь к файлу базы данных для SQLite3 (многопроцессная архитектура Zabbix не позволяет использовать базу данных в памяти, например :memory:, file::memory:?cache=shared или file:memdb1?mode=memory&cache=
Не пытайтесь использовать ту же базу данных, что и сервер Zabbix.

Обязательно: Да

DBPassword

Пароль базы данных. Закомментируйте эту строку, если пароль не используется. Игнорируется при использовании SQLite.

DBPort

Порт узла сети базы данных Zabbix (см. DBHost).[1] Игнорируется для SQLite.

По умолчанию для MySQL: 3306

По умолчанию для PostgreSQL: 5432

Диапазон: 1024-65535

DBSchema

Имя схемы базы данных. Используется для PostgreSQL.

DBSocket

Путь к файлу сокета MySQL.[1]<br>Порт базы данных, если не используется локальный сокет. Игнорируется для SQLite.

По умолчанию: 3306

DBUser

Пользователь базы данных. Игнорируется для SQLite.

DBTLSConnect

Установка этой опции принуждает использовать TLS подключение к базе данных:<br>*required* — подключаться с использованием TLS<br>*verify_ca* — подключаться с использованием TLS и проверкой сертификата<br>*verify_full*

— подключаться с использованием TLS, проверкой сертификата и проверкой, что идентификатор базы данных, указанный параметром DBHost, совпадает с его сертификатом<br><br>Для MySQL, начиная с версии 5.7.11, и PostgreSQL поддерживаются следующие значения: `required`, `verify_ca`, `verify_full`. Для MariaDB, начиная с версии 10.2.6, поддерживаются значения `required` и `verify_full`.<br>По умолчанию значение на задано, а поведение зависит от конфигурации базы данных.

DBTLSCAFile

Полный путь к файлу, содержащему сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата базы данных.

Обязательное: нет (да, если DBTLSConnect выставлен в одно из: *verify_ca* или *verify_full*)

DBTLSCertFile

Полный путь к файлу, содержащему сертификат Zabbix прокси сервера для аутентификации к базе данных.

DBTLSKeyFile

Полный путь к файлу, содержащему закрытый ключ (private key) для аутентификации к базе данных

DBTLSCipher

Список алгоритмов шифрования, разрешённых со стороны Zabbix прокси сервера для протокола TLS до версии TLSv1.2 включительно. Поддерживается только для MySQL.

DBTLSCipher13

Список алгоритмов шифрования, разрешённых со стороны Zabbix прокси сервера для протокола TLSv1.3. Поддерживается только для MySQL, начиная с версии 8.0.16.

DebugLevel

Укажите уровень отладки:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критическая информация;<br>*2* — информация об ошибках;<br>*3* — предупреждения; <br>*4* — для отладки (выдает много информации);<br>*5* — расширенная отладка (выдает еще больше информации).

По умолчанию: 3<br> Диапазон: 0-5

EnableRemoteCommands

Разрешены ли удалённые команды со стороны Zabbix сервера.

По умолчанию: 0<br> Значения: 0 — не разрешены; 1 — разрешены

ExternalScripts

Расположение внешних скриптов (зависит от указанной при компиляции переменной `datadir`).

По умолчанию: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Расположение утилиты fping6. Убедитесь, что владельцем бинарного файла fping6 является root и флаг SUID установлен. Оставьте пустым («Fping6Location=»), если ваша утилита fping поддерживает обработку адресов IPv6.

По умолчанию: `/usr/sbin/fping6`

FpingLocation

Расположение утилиты fping. Убедитесь, что владельцем бинарного файла fping является root и флаг SUID установлен.

По умолчанию: `/usr/sbin/fping`

HistoryCacheSize

Размер кэша истории в байтах. Размер общей памяти для хранения данных истории.

По умолчанию: `16M`<br> Диапазон: 128K-16G

HistoryIndexCacheSize

Размер кэша индекса истории, в байтах. Размер общей памяти для индексирования данных истории, хранящихся в кэше истории. Размер кэша индекса должен составлять примерно 100 байт на кэширование одного элемента данных.

По умолчанию: `4M`<br> Диапазон: 128K-16G

Hostname

Уникальное регистрозависимое имя прокси. Убедитесь, что имя прокси известно серверу.<br>Допустимые символы: алфавитно-цифровые, «.», « », «_» и «-». Максимальная длина: 128

По умолчанию: Задаётся параметром HostnameItem

HostnameItem

Элемент данных используется, если параметр Hostname не задан (работает на прокси аналогично как на агенте). Игнорируется, если задан параметр Hostname.<br> UserParameters, счетчики производительности или алиасы не поддерживаются, но поддерживается system.run[].

По умолчанию: system.hostname

HousekeepingFrequency

Как часто Zabbix будет выполнять процедуру очистки базы (в часах). Автоматическая очистка базы данных удаляет из базы устаревшую информацию.<br>*На заметку*: Для снижения нагрузки на прокси сервер запуск процесса очистки отложен на 30 минут после запуска прокси. Таким образом, если HousekeepingFrequency равен 1 часу, самая первая процедура очистки запустится через 30 минут после запуска прокси, и затем будет повторяться с задержкой в один час.<br>Имеется возможность отключить автоматическую очистку истории, указав HousekeepingFrequency равным 0. В этом случае процедуру очистки истории можно запустить только с помощью опции управления работой *housekeeper_execute*.<br>Смотрите также опции управления работой.

По умолчанию: 1<br> Диапазон: 0-24

Include

Вы можете включить отдельные файлы или все файлы в каталоге в файл конфигурации.<br>Чтобы включить в указанный каталог только соответствующие файлы, для сопоставления с шаблоном поддерживается подстановочный знак звёздочки.<br>Смотрите специальные заметки по поводу ограничений.

Пример:

```
Include=/      /   / /    /    /*.conf
```

JavaGateway

IP адрес (или имя хоста) Zabbix Java gateway. Требуется, только если запущены поллеры Java.

JavaGatewayPort

Порт, который слушает Zabbix Java gateway.

По умолчанию: 10052<br> Диапазон: 1024-32767

ListenBacklog

Максимальное количество соединений в состоянии ожидания в очереди TCP.<br>Значение по умолчанию является зашитой в коде константой, которая зависит от системы.<br>Максимально поддерживаемое значение зависит от системы, слишком большие значения могу быть молча урезаны до «указанного в реализации максимума» (‘implementation-specified maximum’).

По умолчанию: SOMAXCONN<br> Диапазон: 0 – INT_MAX

ListenIP

Список (через запятую) IP адресов, которые должен слушать траппер.<br>Если этот параметр не указан, траппер будет слушать все сетевые интерфейсы

По умолчанию: 0.0.0.0

ListenPort

Порт, который слушает траппер.

По умолчанию: 10051<br> Диапазон: 1024-32767

LoadModule

Модуль, загружаемый при запуске прокси. Модули используются для расширения функциональности прокси. Либо модуль должен находиться в папке, указанной параметром LoadModulePath, либо путь должен быть указан перед именем модуля. Если указанный путь является абсолютным (начинается с «/»), то параметр LoadModulePath игнорируется.<br>Форматы:<br>LoadModule=<module.so><br>LoadModule=<путь/module.so><br>LoadModule= </абс_путь/modul
добавлять несколько параметров LoadModule.

LoadModulePath

Абсолютный путь к папке с модулями прокси сервера. Значение по умолчанию зависит от опций компиляции.

LogFile

Имя файла журнала.

Обязательное: Да, если LogType выставлен в *file*; в противном случае — нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключение автоматической ротации журнала.<br>*Примечание*: Если лимит размера файла достигнут, а ротация по каким-либо причинам не удалась, существующий файл журнала усекается и начинается заново.

По умолчанию: 1<br> Диапазон: 0-1024<br>

LogRemoteCommands

Разрешение журналировать выполняемые команды оболочки как предупреждения.

По умолчанию: 0<br> Значения: 0 — запрещено, 1 — разрешено

Тип журнала

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*system* — запись журнала в syslog;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: `file`

LogSlowQueries

Как долго могут выполняться запросы к базе данных до того как они будут записаны в журнал (в миллисекундах).<br>0 — не журналировать медленные запросы.<br>Эта опция активируется, начиная с DebugLevel=3.

По умолчанию: 0<br> Диапазон: 0-3600000

MaxConcurrentChecksPerPoller

Максимальное количество асинхронных проверок, которые могут быть выполнены за раз каждым поллером HTTP агента, поллером агента или поллером SNMP. Смотрите StartHTTPAgentPollers, StartAgentPollers и StartSNMPPollers.

По умолчанию: 1000<br> Диапазон: 1-1000

PidFile

Имя PID файла.

По умолчанию: `/tmp/zabbix_proxy.pid`

ProxyBufferMode

Определяет механизм хранения данных истории, сетевого обнаружения и авторегистрации: *disk* - данные хранятся в базе данных и загружаются из базы данных; *memory* - данные хранятся в памяти и загружаются из памяти. Если буферу не хватает памяти, старые данные будут отброшены. При завершении работы буфер удаляется. *hybrid* - буфер прокси обычно работает как в режиме memory, пока не закончится память или пока самая старая запись не превысит заданный возраст. Если это происходит, буфер сбрасывается в базу данных и работает как в режиме disk, пока все данные не будут загружены, после чего снова начинает работать с памятью. При завершении работы буфер памяти сбрасывается в базу данных.

См. также: Буфер памяти прокси.

По умолчанию: `disk`<br> Значения: disk; memory; hybrid

ProxyConfigFrequency

Как часто Zabbix прокси будет забирать данные конфигурации с Zabbix сервера, в секундах.<br>Параметр активного прокси. Игнорируется для пассивных прокси (см. параметр ProxyMode).

По умолчанию: 10<br> Диапазон: 1-604800

ProxyLocalBuffer

Прокси будет локально хранить данные N часов, даже если данные уже синхронизированы с сервером.<br>Этот параметр можно использовать, если локальные данные используются сторонними приложениями.

По умолчанию: 0<br> Диапазон: 0-720

ProxyMemoryBufferAge

Максимальный возраст данных в буфере памяти прокси сервера, в секундах. Если включено (не ноль) и записи в буфере памяти прокси сервера более старые, то он заставляет буфер прокси сервера переключиться в режим базы данных до тех пор, пока все записи не будут выгружены на сервер. Этот параметр должен быть меньше или равен параметру ProxyOfflineBuffer.

По умолчанию: 0<br> Диапазон: 0;600-864000

ProxyMemoryBufferSize

Размер разделяемой памяти кэша для собранных данных истории, обнаружения и авторегистрации, в байтах. Если включено (не ноль), прокси будет хранить данные истории, обнаружения и авторегистрации в памяти, если только кэш не заполнен или сохранённые записи не старше, чем определено параметром ProxyMemoryBufferAge. Этот параметр нельзя использовать вместе с параметром ProxyLocalBuffer.

По умолчанию: 0<br> Диапазон: 0;128K-2G

ProxyMode

Режим работы прокси.<br>0 — прокси в активном режиме<br>1 — прокси в пассивном режиме<br>*Обратите внимание*, что при использовании активного прокси (чувствительные) данные конфигурации прокси могут стать доступны лицам, имеющим доступ к порту траппера Zabbix сервера. Такое возможно, потому что кто угодно может представиться активным прокси и запросить данные конфигурации; аутентификация не выполняется.

По умолчанию: 0<br> Диапазон: 0-1

ProxyOfflineBuffer

Прокси будет хранить собранные данные N часов в случае, если нет связи с сервером.<br> Более старые данные будут утеряны.

По умолчанию: 1<br> Диапазон: 1-720

Server

Если ProxyMode задан значением *активный режим*:<br>IP адрес или DNS имя Zabbix сервера (в формате адрес:порт) или кластер (в формате адрес:порт;адрес2:порт) для получения данных конфигурации с сервера и отправки данных на сервер.<br>Если порт не указан, используется порт по умолчанию.<br>Ноды кластера должны разделяться символом «точка с запятой».<br><br>Если ProxyMode задан значением *пассивный режим*:<br>Список (через запятую) IP адресов, опционально в CIDR нотации, или DNS имён Zabbix серверов. Входящие соединения будут приниматься только с адресов, указанных в этом списке. Если включена поддержка IPv6, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» обрабатываются одинаково.<br>«::/0» разрешает все IPv4 и IPv6 адреса. «0.0.0.0/0» можно использовать, чтобы разрешить любой IPv4 адрес.

Пример:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Обязательное: да

SNMPTrapperFile

Временный файл, используемый для передачи данных серверу от демона SNMP trap.<br>Должен быть таким же, как и в файле zabbix_trap_receiver.pl или файле конфигурации SNMPTT.

По умолчанию: `/tmp/zabbix_traps.tmp`

SocketDir

Папка для хранения IPC сокетов, которые используются внутренними службами Zabbix.

По умолчанию: `/tmp`

SourceIP

Локальный IP адрес для:

- исходящих подключений к Zabbix серверу;
- безагентных подключений (VMware, SSH, JMX, SNMP, Telnet и простые проверки);
- подключений HTTP агента;
- запросов HTTP из JavaScript для скриптовых элементов данных;
- запросов HTTP из JavaScript во время предобработки;
- подключений к Хранилищу.

SSHKeyLocation

Местоположение открытых и закрытых ключей для проверок и действий SSH.

SSLCertLocation

Местоположение файлов клиентских сертификатов SSL для аутентификации клиента.<br>Этот параметр используется только в веб-мониторинге.

SSLKeyLocation

Местоположение файлов закрытых ключей SSL для аутентификации клиента.<br>Этот параметр используется только в веб-мониторинге.

SSLCALocation

Местоположение файлов удостоверяющих центров (certificate authority, CA) для верификации SSL сертификатов сервера.<br>Обратите внимание, что значение этого параметра задаст опцию libcurl CURLOPT_CAPATH. Для libcurl версий до 7.42.0 эта опция имеет эффект, только если libcurl скомпилирован для использования OpenSSL. Для получения более подробной информации смотрите веб-страницу cURL [en].<br>Этот параметр используется в веб-мониторинге и при аутентификации SMTP.

StartAgentPollers

Количество экземпляров запускаемых при старте асинхронных процессов поллеров Zabbix агентов (agent poller). Смотрите MaxConcurrentChecksPerPoller.

По умолчанию: 1<br> Диапазон: 0-1000

StartBrowserPollers

Количество экземпляров запускаемых при старте процессов поллеров браузерных элементов данных (browser poller).

По умолчанию: 1<br> Диапазон: 0-1000

StartDBSyncers

Количество экземпляров запускаемых при старте процессов записи истории в базу данных (history syncer).<br>*Примечание*: Будьте осторожны при изменении этого значения, его увеличение может принести больше вреда, чем пользы.

По умолчанию: 4<br> Диапазон: 1-100

StartDiscoverers

Количество экземпляров запускаемых при старте процессов сетевого обнаружения (discoverer).

По умолчанию: 5<br> Диапазон: 0-1000

StartHTTPAgentPollers

Количество экземпляров запускаемых при старте асинхронных процессов поллеров HTTP агента (http agent poller). Смотрите MaxConcurrentChecksPerPoller.

По умолчанию: 1<br> Диапазон: 0-1000

StartHTTPPollers

Количество экземпляров запускаемых при старте процессов поллеров HTTP (HTTP pollers).

По умолчанию: 1<br> Диапазон: 0-1000

StartIPMIPollers

Количество экземпляров запускаемых при старте процессов поллеров IPMI (ipmi poller)

По умолчанию: 0<br> Диапазон: 0-1000

StartJavaPollers

Количество экземпляров запускаемых при старте процессов поллеров Java (java poller).

По умолчанию: 0<br> Диапазон: 0-1000

StartODBCPollers

Количество экземпляров запускаемых при старте процессов поллеров ODBC (odbc poller).

По умолчанию: 1<br> Диапазон: 0-1000

StartPingers

Количество экземпляров запускаемых при старте процессов пингеров ICMP (icmp pinger).

По умолчанию: 1<br> Диапазон: 0-1000

StartPollersUnreachable

Количество экземпляров запускаемых при старте процессов поллеров недоступных устройств (unreachable poller) (в том числе IPMI и Java). По крайней мере один поллер недоступных устройств должен быть запущен, если запускаются обычные поллеры, IPMI или Java.

По умолчанию: 1<br> Диапазон: 0-1000

StartPollers

Количество экземпляров запускаемых при старте процессов поллеров (poller).

По умолчанию: 5<br> Диапазон: 0-1000

Предварительно запущенные препроцессоры

Количество заранее запущенных экземпляров потоков предварительной обработки worker следует задавать не меньше количества доступных ядер CPU. Следует увеличить число worker, если предварительная обработка не ограничена CPU и выполняет много сетевых запросов.

По умолчанию: 16<br> Диапазон: 1-1000

StartSNMPPollers

Количество экземпляров запускаемых при старте асинхронных процессов поллеров SNMP (snmp poller). Смотрите MaxConcurrentChecksPerPoller.

По умолчанию: 1<br> Диапазон: 0-1000

StartSNMPTrapper

Если выставлено в 1, процесс траппера SNMP (snmp trapper) будет запущен.

По умолчанию: 0<br> Диапазон: 0-1

StartTrappers

Количество экземпляров запускаемых при старте процессов трапперов (trapper).<br>Трапперы принимают входящие соединения от Zabbix sender'а и активных агентов.

По умолчанию: 5<br> Диапазон: 0-1000

StartVMwareCollectors

Количество экземпляров запускаемых при старте процессов сборщиков данных VMware (vmware collector).

По умолчанию: 0<br> Диапазон: 0-250

StatsAllowedIP

Список (через запятую) IP адресов, возможно в нотации CIDR, или DNS имён внешних экземпляров Zabbix. Запрос статистики будет приниматься только с перечисленных здесь адресов. Если параметр не задан, никакие запросы статистики приниматься не будут.<br>Если поддержка IPv6 включена, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» рассматриваются как одинаковые, а «::/0» разрешит любой адрес IPv4 или IPv6. Можно использовать «0.0.0.0/0» , чтобы разрешить любой адрес IPv4.

Пример:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Timeout

Указывает, как долго (в секундах) следует ожидать установления соединения и обмена данными с Zabbix сервером, агентом, веб-сервисом, а также для старых проверок SNMP (только OID-номер либо строка).

Этот параметр определяет продолжительность различных коммуникационных операций:

- Выполнение удалённой команды на Zabbix агенте.
- Выполнение команды SSH / Telnet.
- Перепланирование элементов данных, когда интерфейс IPMI становится недоступным.
- Отправка ответа Zabbix серверу при сбое обмена данными из-за проблем с правами или шифрованием.
- Крайний срок для асинхронных сокетов IPC и опций управления работой.
- Запросы асинхронного поллера DNS.
- Ответы на сообщения heartbeat активных проверок.
- Получение данных (значений) Zabbix агента от активных агентов.
- Получение данных от утилиты Zabbix sender.
- Отправка Zabbix агенту списка активных проверок.

Этот тайм-аут **не** будет использоваться для тех проверок, которые имеют параметры гибкого тайм-аута, настраиваемые в веб-интерфейсе (на глобальном уровне, на уровне прокси-сервера или на уровне отдельных элементов данных). Например, элементы данных SNMP `walk[OID]` и `get[OID]` используют настраиваемый тайм-аут из веб-интерфейса; а старые проверки SNMP по-прежнему используют значение тайм-аута сервера.

По умолчанию: 3<br> Диапазон: 1-30

TLSAccept

Какие входящие подключения принимаются от Zabbix сервера. Используется пассивным прокси, игнорируется активным прокси. Можно указывать несколько значений через запятую:<br>*unencrypted* — принимать подключения без шифрования (по умолчанию)<br>*psk* — принимать подключения с TLS и общим ключом (pre-shared key, PSK)<br>*cert* — принимать подключения с TLS и сертификатом

Обязательное: да для пассивных прокси, если заданы параметры TLS сертификат или PSK (даже для *незашифрованных* подключений), в противном случае — нет

TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Абсолютный путь к файлу, который содержит сертификат или цепочку сертификатов прокси, используется для зашифрованных соединений между компонентами Zabbix.

TLSCipherAll

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов и общего ключа (PSK).

Пример:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов и общего ключа (PSK).

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Пример для OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов.

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Пример для OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов.

TLSCipherPSK

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе общего ключа (PSK).

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Пример для OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе общего ключа (PSK).

Пример:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Как прокси должен подключаться к Zabbix серверу. Используется активным прокси, игнорируется пассивным прокси. Можно указать только одно значение:<br>*unencrypted* — подключаться без шифрования (по умолчанию)<br>*psk* — подключаться, используя TLS и общий ключ (pre-shared key, PSK)<br>*cert* — подключаться, используя TLS и сертификат

Обязательное: да для активных прокси, если заданы параметры TLS сертификат или PSK (даже для *незашифрованных* подключений), в противном случае — нет

TLSCRLFile

Абсолютный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ прокси сервера (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSListen

Управляет TLS на сокете trapper.

Поддерживаемые значения:

- required - принимать только TLS-соединения

TLSPSKFile

Абсолютный путь к файлу, содержащему общий ключ (pre-shared key) прокси сервера, используется для зашифрованных коммуникаций с сервером Zabbix.

TLSPSKIdentity

Строка идентификатора PSK, используется для зашифрованных соединений с Zabbix сервером.

TLSServerCertIssuer

Разрешённый эмитент сертификата сервера.

TLSServerCertSubject

Разрешённый субъект сертификата сервера.

TmpDir

Папка с временными данными.

По умолчанию: `/tmp`

TrapperTimeout

Указывает время ожидания в секундах для:

- получения данных конфигурации от Zabbix сервера;
- выполнения глобального скрипта или выполнения удалённой команды.

По умолчанию: 300<br> Диапазон: 1-300

UnavailableDelay

Как часто узел сети будет проверяться на доступность в период его недоступности, в секундах.

По умолчанию: 60<br> Диапазон: 1-3600

UnreachableDelay

Как часто узел сети будет проверяться на доступность в период его недостижимости, в секундах.

По умолчанию: 15<br> Диапазон: 1-3600

UnreachablePeriod

Через сколько секунд недостижимости узел сети считается недоступным.

По умолчанию: 45<br> Диапазон: 1-3600

Пользователь

Использование привилегий указанного, существующего пользователя системы.<br>Имеет эффект, только если запускается под «root» и AllowRoot отключён.

По умолчанию: zabbix

Vault

Указывает поставщика хранилища:<br>*HashiCorp* — HashiCorp KV Secrets Engine version 2<br>*CyberArk* — CyberArk Central Credential Provider<br>Должен совпадать с поставщиком хранилища, указанном в веб-интерфейсе.

По умолчанию: HashiCorp

VaultDBPath

Путь или запрос в Vault, в зависимости от Vault, откуда по ключам будут получены учетные данные для базы данных.

Ключи, используемые для HashiCorp, — 'password' и 'username'.

Пример пути с VaultPrefix=/v1/secret/data/zabbix/:

```
database
```

Пример пути без VaultPrefix:

```
secret/zabbix/database
```

Ключи, используемые для CyberArk, — 'Content' и 'UserName'.

Пример:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_proxy_database
```

Этот параметр можно использовать только если DBUser и DBPassword не указаны.

VaultPrefix

Пользовательский префикс для пути или запроса Vault в зависимости от Vault. Если не указано, будут использованы наиболее подходящие значения по умолчанию.<br>Обратите внимание, что 'data' автоматически добавляется после точки монтирования для HashiCorp, если VaultPrefix не указан. <br>Обратите внимание, что 'data' автоматически добавляется после точки монтирования для HashiCorp, если VaultPrefix не указан.

Пример префикса для Hashicorp:

```
v1/secret/data/zabbix/
```

Пример префикса для Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Имя файла SSL сертификата, используемого для аутентификации клиента. Файл сертификата должен быть в формате PEM1.<br>Если файл сертификата также содержит закрытый ключ, оставьте поле файла ключа SSL пустым.<br>Папка, содержащая этот файл, указывается параметром конфигурации SSLCertLocation.<br>Эту опцию можно опустить, но она рекомендуется для хранилища CyberArkCCP.

VaultTLSKeyFile

Имя файла закрытого ключа SSL, используемого для аутентификации клиента. Файл закрытого ключа должен быть в формате PEM1.<br>Папка, содержащая этот файл, указывается параметром конфигурации SSLKeyLocation. <br>Эту опцию можно опустить, но она рекомендуется для хранилища CyberArkCCP.

VaultToken

Токен аутентификации Хранилища HashiCorp, который должен быть сгенерирован исключительно для Zabbix прокси с разрешениями только на чтение на пути, указанные в необязательном параметре конфигурации VaultDBPath.<br>Если одновременно заданы и параметр VaultToken, и переменная окружения VAULT_TOKEN, то это считается ошибкой.

Обязательное: Да, если Vault задан в значение *HashiCorp*; в противном случае — нет

VaultURL

HTTP[S] URL сервера Хранилища. Если SSLCALocation не указан, будет использоваться общесистемная папка сертификатов удостоверяющих центров (certificate authority, CA).

По умолчанию: `https://127.0.0.1:8200`

VMwareCacheSize

Размер разделяемой памяти для хранения данных VMware.<br>Для мониторинга использования кэша VMware можно использовать внутренние проверки VMware zabbix[vmware,buffer,...] (смотрите раздел о внутренних проверках).<br>Обратите внимание, что разделяемая память не выделяется, если не настроено на запуск ни одного экземпляра коллекторов VMware.

По умолчанию: 8M<br> Диапазон: 256K-2G

VMwareFrequency

Задержка в секундах между сбором данных с одной службы VMware.<br>Эту задержку необходимо установить в наименьшее из значений интервала обновления у любого элемента данных мониторинга VMware.

По умолчанию: 60<br> Диапазон: 10-86400

VMwarePerfFrequency

Задержка в секундах между получением статистики по счётчикам производительности с одной службы VMware.<br>Эту задержку необходимо установить в наименьшее из значений интервала обновления у любого элемента данных мониторинга VMware, который использует счётчики производительности VMware.

По умолчанию: 60<br> Диапазон: 10-86400

VMwareTimeout

Максимальное количество секунд, которое коллектор vmware потратит на ожидание ответа от службы VMware (vCenter или ESX гипервизора).

По умолчанию: 10<br> Диапазон: 1-300

WebDriverURL

HTTP[S] URL интерфейса WebDriver'а.

Пример (используется с автономным сервером Selenium WebDriver):

```
WebDriverURL=http://localhost:4444
```

Сноски

[1] DBSocket и DBPort взаимоисключают друг друга в конфигурации прокси. Укажите только один из них или оставьте оба не заданными.

### 3. Zabbix агент (UNIX)

Обзор

Параметры, поддерживаемые файлом конфигурации агента Zabbix (zabbix_agentd.conf), перечислены в этом разделе.

Параметры перечислены без дополнительной информации. Щелкните параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| Alias | Задает псевдоним для ключа элемента данных. |
| AllowKey | Разрешает выполнение тех ключей элементов данных, которые соответствуют шаблону. |
| AllowRoot | Разрешает агенту запускаться от имени 'root'. |
| BufferSend | Не хранить данные в буфере дольше N секунд. |
| BufferSize | Максимальное количество значений в буфере памяти. |
| DebugLevel | Уровень отладки. |
| DenyKey | Запрещает выполнение тех ключей элементов данных, которые соответствуют шаблону. |
| EnableRemoteCommands | Разрешены ли удаленные команды от сервера Zabbix. |
| HeartbeatFrequency | Частота сообщений heartbeat в секундах. |
| HostInterface | Необязательный параметр, определяющий интерфейс узла сети. |
| HostInterfaceItem | Необязательный параметр, определяющий элемент данных, используемый для получения интерфейса узла сети. |
| HostMetadata | Необязательный параметр, определяющий метаданные узла сети. |

| Parameter | Description |
|---|---|
| HostMetadataItem | Необязательный параметр, определяющий элемент данных агента Zabbix, используемый для получения метаданных узла сети. |
| Hostname | Необязательный параметр, определяющий имя узла сети. |
| HostnameItem | Необязательный параметр, определяющий элемент данных агента Zabbix, используемый для получения имени узла сети. |
| Include | В файл конфигурации можно включать отдельные файлы или все файлы в каталоге. |
| ListenBacklog | Максимальное количество ожидающих соединений в очереди TCP. |
| ListenIP | Список IP-адресов, разделенных запятыми, на которых должен слушать агент. |
| ListenPort | Агент будет слушать на этом порту подключения от сервера. |
| LoadModule | Модуль, который нужно загрузить при запуске агента. |
| LoadModulePath | Полный путь к расположению модулей агента. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogRemoteCommands | Включить запись выполненных команд shell как предупреждений. |
| LogType | Тип вывода журнала. |
| MaxLinesPerSecond | Максимальное количество новых строк, которое агент будет отправлять в секунду на сервер Zabbix или прокси при обработке активных проверок ’log’ и ’logrt’. |
| PidFile | Имя файла PID. |
| RefreshActiveChecks | Как часто обновляется список активных проверок. |
| Server | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен серверов Zabbix и прокси Zabbix. |
| ServerActive | Адрес сервера Zabbix/прокси или конфигурация кластера, откуда получать активные проверки. |
| SourceIP | Исходный IP-адрес. |
| StartAgents | Количество предварительно порожденных экземпляров zabbix_agentd, обрабатывающих пассивные проверки. |
| Timeout | Указывает, сколько времени ожидать (в секундах) установления соединения и обмена данными с прокси Zabbix или сервером. |
| TLSAccept | Какие входящие соединения принимать. |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA(ов), используемые для проверки сертификата узла при зашифрованных соединениях между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов, используемые для зашифрованных соединений между компонентами Zabbix. |
| TLSCipherAll | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherAll13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherCert | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherCert13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherPSK | Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSCipherPSK13 | Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSConnect | Как агент должен подключаться к серверу Zabbix или прокси. |
| TLSCRLFile | Полный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных соединений между компонентами Zabbix. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ агента, используемый для зашифрованных соединений между компонентами Zabbix. |
| TLSPSKFile | Полный путь к файлу, содержащему предварительно общий ключ агента, используемый для зашифрованных соединений с сервером Zabbix. |
| TLSPSKIdentity | Строка идентификатора предварительно общего ключа, используемая для зашифрованных соединений с сервером Zabbix. |
| TLSServerCertIssuer | Разрешенный издатель сертификата сервера (прокси). |
| TLSServerCertSubject | Разрешенный субъект сертификата сервера (прокси). |
| UnsafeUserParameters | Разрешить передачу всех символов в аргументах пользовательских параметров. |
| User | Снизить привилегии до конкретного существующего пользователя в системе. |
| UserParameter | Пользовательский параметр для мониторинга. |

| Parameter | Description |
|---|---|
| UserParameterDir | Путь поиска по умолчанию для команд UserParameter. |

Все параметры являются необязательными, если явно не указано, что параметр обязателен.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию демона, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды.
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Подробности параметра

Alias

Задаёт алиас для ключа элемента данных. Его можно использовать для замены длинных и сложных ключей элементов данных на более короткие и простые.<br> Можно добавлять несколько параметров *Alias*. Несколько параметров с одинаковым ключом *Alias* не разрешены.<br> Различные ключи параметра *Alias* могут ссылаться на тот же ключ элемента данных.<br> Алиасы можно использовать в параметре *HostMetadataItem*, но нельзя в параметре *HostnameItem*.

Пример 1: Получение числового идентификатора пользователя «zabbix».

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Теперь для получения данных можно использовать сокращённый ключ **zabbix.userid**.

Пример 2: Получение утилизации CPU с параметрами по умолчанию и с пользовательскими параметрами.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Такая запись позволяет использовать ключ **cpu.util** для получения утилизации CPU в процентах с параметрами по умолчанию, а также использовать **cpu.util[all, idle, avg15]** для получения конкретных данных об утилизации CPU.

Пример 3: Выполнение нескольких правил низкоуровневого обнаружения, которые обрабатывают одни и те же элементы данных обнаружения.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Теперь имеется возможность настроить несколько правил обнаружения, используя **vfs.fs.discovery** с разными параметрами для каждого правила, например, **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** и т.д.

AllowKey

Разрешение выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с DenyKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

AllowRoot

Разрешение агенту запускаться под «root». Если отключено, а агент запускается из-под «root», агент вместо этого будет пытаться переключиться на пользователя «zabbix». Игнорируется, если агент запускается под обычным пользователем.

По умолчанию: 0<br> Значения: 0 — не разрешено; 1 — разрешено

BufferSend

Не хранить данные в буфере дольше N секунд.

По умолчанию: 5<br> Диапазон: 1-3600

BufferSize

Максимальное количество значений в буфере памяти. При заполнении буфера агент будет отправлять все собранные данные Zabbix серверу или прокси.

По умолчанию: 100<br> Диапазон: 2-65535

DebugLevel

Укажите уровень отладки:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критическая информация;<br>*2* — информация об ошибках;<br>*3* — предупреждения; <br>*4* — для отладки (выдает много информации);<br>*5* — расширенная отладка (выдает еще больше информации).

По умолчанию: 3<br> Диапазон: 0-5

DenyKey

Запрет выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с AllowKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

EnableRemoteCommands

Разрешены ли удалённые команды с Zabbix сервера. Этот параметр **устарел**, используйте вместо него AllowKey=system.run[*] или DenyKey=system.run[*]. <br>Это внутренний алиас для параметров AllowKey/DenyKey, зависящий от значения:<br>0 — DenyKey=system.run[*]<br>1 — AllowKey=system.run[*]

По умолчанию: 0<br> Значения: 0 — не разрешены; 1 — разрешены

HeartbeatFrequency

Частота контрольных сообщений (heartbeat) в секундах. Используется для контроля доступности активных проверок.<br>0 — контрольные сообщения отключены.

По умолчанию: 60<br> Диапазон: 0-3600

HostInterface

Опциональный параметр, который задаёт интерфейс узла сети (IP адрес или DNS имя), используемый в процессе авторегистрации узла сети. Это значение будет использовано для заполнения интерфейса у свежесоздаваемого узла сети, можно явно настроить либо IP, либо DNS адрес. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Если не задано, значение будет получено из HostInterfaceItem.

Агент выдаст ошибку и не запустится, если значение превышает ограничение в 255 символов.

Диапазон: 0–255 символов.

HostInterfaceItem

Опциональный параметр, который задаёт элемент данных, используемый для получения интерфейса узла сети (IP адрес либо DNS имя) в процессе авторегистрации узла сети. Это значение используется, только если HostInterface не определён. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Во время запроса на авторегистрацию агент будет выводить в журнал предупреждение, если значение, возвращаемое указанным элементом данных, превышает ограничение в 255 символов.

Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

HostMetadata

Опциональный параметр, который задаёт метаданные узла сети, используемые, чтобы идентифицировать или отличить узел сети в процессе авторегистрации узлов сети (активный агент). HostMetadata позволяет различать узлы сети не только по имени хоста.

Если не определено, то значение берётся от HostMetadataItem.

Агент выдаст ошибку и не запустится, если указанное значение превышает ограничение 2034 байта или не является UTF-8 строкой. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет усечён по первому переводу строки.

Диапазон: 0–2034 байта.

HostMetadataItem

Опциональный параметр, который задаёт элемент данных Zabbix агента, используемый для получения метаданных узла сети. Этот параметр используется, только если HostMetadata не определён.

Значение HostMetadataItem извлекается на каждую попытку авторегистрации и используется только в процессе автоматической регистрации узлов сети (активный агент). HostMetadata позволяет различать узлы сети не только по имени хоста.

Поддерживаются UserParameters и алиасы. Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

При запросе на авторегистрацию агент запишет в журнал предупреждение, если полученное от указанного элемента данных значение превышает ограничение в 65535 кодовых позиций (code points) UTF-8. Значение, возвращаемое указанным элементом данных, должно являться UTF-8 строкой, в противном случае оно будет игнорироваться. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет усечён по первому переводу строки.

Hostname

Список (через запятую) уникальных, регистрозависимых имён узла сети. Требуется для активных проверок и должно совпадать с именами узла сети, указанными на сервере. Если не задано, значение будет браться из HostnameItem.<br>Допустимые символы: алфавитно-цифровые, «.», « », «_» и «-». Максимальная длина: 128 символов на имя узла сети, 2048 символов для всей строки.

По умолчанию: Задаётся параметром HostnameItem

HostnameItem

Опциональный параметр, который задаёт элемент данных Zabbix агента, используемый для получения имени узла сети. Этот параметр используется, только если Hostname не определён. Не поддерживает UserParameters или алиасы, но поддерживает system.run[], независимо от значений AllowKey/DenyKey.

По умолчанию: `system.hostname`

Include

Вы можете включить отдельные файлы или все файлы в каталоге в файл конфигурации. Чтобы включить в указанный каталог только соответствующие файлы, для сопоставления с шаблоном поддерживается подстановочный знак звёздочки.<br>Смотрите специальные заметки по поводу ограничений.

Пример:

```
Include=/      /  / /   /   /*.conf
```

ListenBacklog

Максимальное количество соединений в состоянии ожидания в очереди TCP.<br>Значение по умолчанию является зашитой в коде константой, которая зависит от системы.<br>Максимально поддерживаемое значение зависит от системы, слишком большие значения могу быть молча урезаны до «указанного в реализации максимума» (’implementation-specified maximum’).

По умолчанию: SOMAXCONN<br> Диапазон: 0 – INT_MAX

ListenIP

Список (через запятую) IP адресов, которые должен слушать агент.

По умолчанию: `0.0.0.0`

ListenPort

Агент будет слушать этот порт на предмет подключений с сервера.

По умолчанию: 10050<br> Диапазон: 1024-32767

LoadModule

Модуль, загружаемый при запуске агента. Модули используются для расширения функциональности агента. Либо модуль должен находиться в папке, указанной параметром LoadModulePath, либо путь должен быть указан перед именем модуля. Если указанный путь является абсолютным (начинается с «/»), то параметр LoadModulePath игнорируется.<br>Форматы:<br>LoadModule=<module.so><br>LoadModule=<путь/module.so><br>LoadModule= </абс_путь/modul
добавлять несколько параметров LoadModule.

LoadModulePath

Абсолютный путь к папке с модулями агента. Значение по умолчанию зависит от опций компиляции.

LogFile

Имя файла журнала.

Обязательное: Да, если LogType выставлен в *file*; в противном случае — нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключение автоматической ротации журнала.<br>*Примечание*: Если лимит размера файла достигнут, а ротация по каким-либо причинам не удалась, существующий файл журнала усекается и начинается заново.

По умолчанию: 1<br> Диапазон: 0-1024<br>

LogRemoteCommands

Разрешение журналировать выполняемые команды оболочки как предупреждения. Команды будут записаны в журнал, только если они выполняются удалённо. Записи в журнале не создаются, если system.run[] запускается локально параметрами HostMetadataItem, HostInterfaceItem или HostnameItem.

По умолчанию: 0<br> Значения: 0 — запрещено, 1 — разрешено

LogType

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*system* — запись журнала в syslog;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: `file`

MaxLinesPerSecond

Максимальное количество новых строк в секунду, которые агент будет отправлять Zabbix серверу или прокси при обработке активных проверок «log» или «logrt». Указаное значение будет переопределено параметром «maxlines», указанным в ключах элементов данных «log» или «logrt».<br>*Примечание*: при поиске нужной строки в элементах данных журналов Zabbix будет обрабатывать в 10 раз больше новых строк, чем указано в параметре *MaxLinesPerSecond*.

По умолчанию: 20<br> Диапазон: 1-1000

PidFile

Имя PID файла.

По умолчанию: `/tmp/zabbix_agent.pid`

RefreshActiveChecks

Как часто обновлять список активных проверок, в секундах. Обратите внимание, что после неуспешного обновления активных проверок, следующая попытка будет предпринята через 60 секунд.

По умолчанию: 5<br> Диапазон: 1-86400

Server

Список (через запятую) IP адресов, опционально в CIDR нотации, или DNS имён хостов Zabbix серверов и Zabbix прокси. Входящие соединения будут приниматься только с хостов, указанных в этом списке. Если включена поддержка IPv6, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» обрабатываются одинаково, а «::/0» разрешает все IPv4 и IPv6 адреса. «0.0.0.0/0» можно использовать, чтобы разрешить любой IPv4 адрес. Обратите внимание, что «IPv4-совместимые IPv6 адреса» (префикс 0000::/96) поддерживаются, но являются устаревшими согласно [RFC4291 [en]] (https://tools.ietf.org/html/rfc4291#section-2.5.5). Пробелы разрешены.

Пример:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Обязательное: да, если для StartAgents явно не установлено значение 0.

ServerActive

Адрес Zabbix сервера/прокси или конфигурация кластера для получения активных проверок. Адрес сервера/прокси — это IP адрес или DNS имя и опционально порт, разделённые двоеточием.<br>Конфигурация кластера — один или более адресов серверов либо членов группы прокси, разделённые символом «точка с запятой». Можно указать несколько серверов/кластеров/прокси Zabbix, разделяя их запятой. От каждого сервера/кластера Zabbix должно быть указано не более одного Zabbix прокси, если только не используются группы прокси. Если указан Zabbix прокси, то сервер/кластер Zabbix для этого прокси не должны быть указаны.<br>Несколько адресов через запятую можно указать для использования нескольких независимых серверов Zabbix параллельно. Пробелы разрешены.<br>Если порт не указан, используется порт по умолчанию.<br>Адреса IPv6 должны быть заключены в квадратные скобки, если для этого хоста указан порт. Если порт не указан, квадратные скобки для адресов IPv6 опциональны.<br>Если этот параметр не указан, активные проверки отключены.

Пример для Zabbix прокси:

```
ServerActive=127.0.0.1:10051
```

Пример для группы Zabbix прокси:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Пример для нескольких серверов:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Пример для кластера высокой доступности:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Пример для конфигурации высокой доступности с двумя кластерами и одним сервером:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Локальный IP адрес для:

- исходящих подключений к Zabbix серверу или прокси;
- создания подключений при выполнении некоторых элементов данных (web.page.get, net.tcp.port и т.п.).

StartAgents

Количество экземпляров запускаемых при старте процессов zabbix_agentd, обрабатывающих пассивные проверки. Если выставлено в 0, отключает пассивные проверки и агент не будет слушать никакие TCP порты.

По умолчанию: 10<br> Диапазон: 0–100

Timeout

Указывает, как долго (в секундах) следует ожидать установления соединения и обмена данными с Zabbix сервером или прокси.<br>

Этот параметр определяет продолжительность различных коммуникационных операций, в том числе:

- ожидание ответа от Zabbix сервера;
- отправку запросов на Zabbix сервер, в том числе запросы конфигурации элементов данных и значения элементов данных в активных проверках;
- получение данных журнала посредством мониторинга файла журнала;
- отправка сообщений heartbeat;
- максимальная длительность для проверок `vfs.*`;
- используется модулями агента Zabbix;
- служит в качестве запасного варианта в сценариях, где сервер или прокси версии старше 7.0 отправляет проверки без тайм-аутов.

Этот тайм-аут **не** будет использоваться для тех проверок агента, которые имеют настраиваемые параметры тайм-аута в веб-интерфейсе (на глобальном уровне, на уровне прокси-сервера или на уровне отдельных элементов данных).

По умолчанию: 3<br> Диапазон: 1-30

TLSAccept

Какие входящие соединения принимать. Используется для пассивных проверок. Можно указать несколько значений через запятую:<br>*unencrypted* — принимать соединения без шифрования (по умолчанию)<br>*psk* — принимать подключения с TLS и общим ключом (pre-shared key, PSK)<br>*cert* — принимать подключения с TLS и сертификатом

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже для *незашифрованных* подключений), в противном случае — нет

TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Абсолютный путь к файлу, который содержит сертификат агента или цепочку сертификатов, используется для зашифрованных соединений между компонентами Zabbix.

TLSCipherAll

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов и общего ключа (PSK).

Пример:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов и общего ключа (PSK).

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Пример для OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов.

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Пример для OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе сертификатов.

TLSCipherPSK

Строка приоритетов GnuTLS или строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе общего ключа (PSK).

Пример для GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Пример для OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSipherPSK13

Строка шифров для OpenSSL 1.1.1 или новее в TLS 1.3. Переопределяет критерии выбора алгоритмов шифрования по умолчанию для шифрования на основе общего ключа (PSK).

Пример:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Как агент должен подключаться к Zabbix серверу или прокси. Используется для активных проверок. Можно указать только одно значение:<br>*unencrypted* — подключаться без шифрования (по умолчанию)<br>*psk* — подключаться, используя TLS и общий ключ (pre-shared key, PSK)<br>*cert* — подключаться, используя TLS и сертификат

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже при *незашифрованном* соединении), в противном случае — нет

TLSCRLFile

Абсолютный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ агента (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSPSKFile

Абсолютный путь к файлу, содержащему общий ключ (pre-shared key) агента. Используется для зашифрованных коммуникаций с сервером Zabbix.

TLSPSKIdentity

Строка идентификатора PSK, используется для зашифрованных соединений с Zabbix сервером.

TLSServerCertIssuer

Разрешённый эмитент сертификата сервера (прокси).

TLSServerCertSubject

Разрешённый субъект сертификата сервера (прокси).

UnsafeUserParameters

Разрешить передавать аргументами в пользовательские параметры все символы. Не разрешены следующие символы: \ ' " ` * ? [ ] { } ~ $ ! & ; ( ) < > | # @<br>Кроме того, не разрешены символы новой строки.

По умолчанию: 0<br> Значения: 0 — не разрешать, 1 — разрешить

User

Использование привилегий указанного, существующего пользователя системы.<br> Имеет эффект только если запускается под «root» и AllowRoot отключён.

По умолчанию: `zabbix`

UserParameter

Пользовательский параметр для мониторинга. Можно указать несколько пользовательских параметров.<br>Формат: UserParameter=<ключ>,<команда оболочки><br>Обратите внимание, что команда не должна возвращать только пустую строку или EOL. Команды оболочки могут иметь относительные пути, если указан параметр UserParameterDir.

Пример:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Путь поиска по умолчанию для команд UserParameter. Если используется, агент перед выполнением команды сменит свою рабочую директорию на указанную здесь. Таким образом, команды UserParameter могут иметь относительный префикс `./` вместо полного пути.<br>Допустимо только одно значение.

Пример:

```
UserParameterDir=/opt/myscripts
```

Смотрите также

1. Различия в конфигурациях Zabbix агента для активных и пассивных проверок, начиная с версии 2.0.0 [en]

#### 4. Zabbix агент 2 (UNIX)

Обзор

Zabbix agent 2 — это агент Zabbix нового поколения, который может использоваться вместо Zabbix agent.

Параметры, поддерживаемые файлом конфигурации Zabbix agent 2 (zabbix_agent2.conf), перечислены в этом разделе.

Параметры приведены без дополнительной информации. Нажмите на параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| Alias | Задает псевдоним для ключа элемента данных. |
| AllowKey | Разрешает выполнение тех ключей элементов данных, которые соответствуют шаблону. |
| BufferSend | Не хранить данные в буфере дольше N секунд. |
| BufferSize | Максимальное количество значений в буфере памяти. |
| ControlSocket | Управляющий сокет, используемый для отправки команд во время выполнения с помощью опции '-R'. |
| DebugLevel | Уровень отладки. |
| DenyKey | Запрещает выполнение тех ключей элементов данных, которые соответствуют шаблону. |

| Parameter | Description |
|---|---|
| EnablePersistentBuffer | Включает использование локального постоянного хранилища для активных элементов данных. |
| ForceActiveChecksOnStart | Выполнять активные проверки сразу после перезапуска для первой полученной конфигурации. |
| HeartbeatFrequency | Частота сообщений heartbeat в секундах. |
| HostInterface | Необязательный параметр, определяющий интерфейс узла сети. |
| HostInterfaceItem | Необязательный параметр, определяющий элемент данных, используемый для получения интерфейса узла сети. |
| HostMetadata | Необязательный параметр, определяющий метаданные узла сети. |
| HostMetadataItem | Необязательный параметр, определяющий элемент данных агента Zabbix, используемый для получения метаданных узла сети. |
| Hostname | Необязательный параметр, определяющий имя узла сети. |
| HostnameItem | Необязательный параметр, определяющий элемент данных агента Zabbix, используемый для получения имени узла сети. |
| Include | В файл конфигурации можно включать отдельные файлы или все файлы в каталоге. |
| ListenIP | Список IP-адресов, разделенных запятыми, на которых агент должен прослушивать подключения. |
| ListenPort | Агент будет прослушивать этот порт для подключений от сервера. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogType | Тип вывода журнала. |
| PersistentBufferFile | Файл, в котором Zabbix agent 2 должен хранить базу данных SQLite. |
| PersistentBufferPeriod | Период времени, в течение которого данные должны храниться при отсутствии соединения с сервером или прокси. |
| PidFile | Имя PID-файла. |
| Plugins.<PluginName>.System.Capacity | Ограничение на количество проверок на плагин, которые могут выполняться одновременно. |
| Plugins.Log.MaxLinesPerSecond | Максимальное количество новых строк, которое агент будет отправлять в секунду на сервер Zabbix или прокси при обработке активных проверок ’log’ и ’logrt’. |
| Plugins.SystemRun.LogRemoteCommands | Включает запись выполненных команд оболочки в журнал как предупреждений. |
| PluginSocket | Путь к UNIX-сокету для взаимодействия с загружаемыми плагинами. |
| PluginTimeout | Тайм-аут для соединений с загружаемыми плагинами, в секундах. |
| RefreshActiveChecks | Как часто обновляется список активных проверок. |
| Server | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен серверов Zabbix и прокси Zabbix. |
| ServerActive | Адрес сервера Zabbix/прокси или конфигурация кластера, откуда получать активные проверки. |
| SourceIP | Исходный IP-адрес. |
| StatusPort | Если задан, агент будет прослушивать этот порт для HTTP-запросов состояния (http://localhost:<port>/status). |
| Timeout | Определяет, сколько времени ждать (в секундах) установления соединения и обмена данными с прокси или сервером Zabbix. |
| TLSAccept | Какие входящие подключения принимать. |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA, используемые для проверки сертификата узла, для зашифрованного обмена данными между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов, используемые для зашифрованного обмена данными между компонентами Zabbix. |
| TLSCipherAll | Строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherAll13 | Строка шифров OpenSSL (TLS 1.3) для OpenSSL 1.1.1 или новее. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherCert | Строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherCert13 | Строка шифров OpenSSL (TLS 1.3) для OpenSSL 1.1.1 или новее. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherPSK | Строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSCipherPSK13 | Строка шифров OpenSSL (TLS 1.3) для OpenSSL 1.1.1 или новее. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSConnect | Способ подключения агента к серверу или прокси Zabbix. |

| Parameter | Description |
|---|---|
| TLSCRLFile | Полный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованного обмена данными между компонентами Zabbix. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ агента, используемый для зашифрованного обмена данными между компонентами Zabbix. |
| TLSPSKFile | Полный путь к файлу, содержащему предварительно общий ключ агента, используемый для зашифрованного обмена данными с сервером Zabbix. |
| TLSPSKIdentity | Строка идентификатора предварительно общего ключа, используемая для зашифрованного обмена данными с сервером Zabbix. |
| TLSServerCertIssuer | Разрешенный издатель сертификата сервера (прокси). |
| TLSServerCertSubject | Разрешенный субъект сертификата сервера (прокси). |
| UnsafeUserParameters | Разрешает передачу всех символов в аргументах пользовательских параметров. |
| UserParameter | Пользовательский параметр для мониторинга. |
| UserParameterDir | Путь поиска по умолчанию для команд UserParameter. |

Все параметры являются необязательными, если явно не указано, что параметр является обязательным.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды.
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Подробности параметра

Alias

Задаёт алиас для ключа элемента данных. Его можно использовать для замены длинных и сложных ключей элементов данных на более короткие и простые.<br> Можно добавлять несколько параметров *Alias*. Несколько параметров с одинаковым ключом *Alias* не разрешены.<br> Различные ключи параметра *Alias* могут ссылаться на тот же ключ элемента данных.<br> Алиасы можно использовать в параметре *HostMetadataItem*, но нельзя в параметре *HostnameItem*.

Пример 1: Получение числового идентификатора пользователя «zabbix».

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Теперь для получения данных можно использовать сокращённый ключ **zabbix.userid**.

Пример 2: Получение утилизации CPU с параметрами по умолчанию и с пользовательскими параметрами.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Такая запись позволяет использовать ключ **cpu.util** для получения утилизации CPU в процентах с параметрами по умолчанию, а также использовать **cpu.util[all, idle, avg15]** для получения конкретных данных об утилизации CPU.

Пример 3: Выполнение нескольких правил низкоуровневого обнаружения, которые обрабатывают одни и те же элементы данных обнаружения.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Теперь имеется возможность настроить несколько правил обнаружения, используя **vfs.fs.discovery** с разными параметрами для каждого правила, например, **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** и т.д.

AllowKey

Разрешение выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с DenyKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

BufferSend

Интервал времени в секундах, определяющий, как часто значения из буфера отсылаются на Zabbix сервер. Обратите внимание, что если буфер заполнен, данные будут отосланы раньше.

По умолчанию: 5<br> Диапазон: 1-3600

BufferSize

Максимальное количество значений в буфере памяти. При заполнении буфера агент будет отправлять все собранные данные Zabbix серверу или прокси. Этот параметр должен использоваться только если постоянный буфер отключён (*EnablePersistentBuffer=0*).

По умолчанию: 1000<br> Диапазон: 2-65535

ControlSocket

Сокет управления, используется при отправке команд управления при помощи опции «-R».

По умолчанию: `/tmp/agent.sock`

DebugLevel

Укажите уровень отладки:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критическая информация;<br>*2* — информация об ошибках;<br>*3* — предупреждения; <br>*4* — для отладки (выдает много информации);<br>*5* — расширенная отладка (выдает еще больше информации).

По умолчанию: 3<br> Диапазон: 0-5

DenyKey

Запрет выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с AllowKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

EnablePersistentBuffer

Включить использование локального постоянного хранилища для активных элементов данных. Если постоянное хранилище отключено, будет использован буфер памяти.

По умолчанию: 0<br> Значения: 0 — отключено, 1 — включено

ForceActiveChecksOnStart

Выполнить активные проверки сразу после перезапуска для первой полученной конфигурации. Также доступно как параметр конфигурации плагина, например: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

По умолчанию: 0<br> Значения: 0 — отключено, 1 — включено

Частота сердцебиения

Частота контрольных сообщений (heartbeat) в секундах. Используется для контроля доступности активных проверок.<br>0 — контрольные сообщения отключены.

По умолчанию: 60<br> Диапазон: 0-3600

HostInterface

Опциональный параметр, который задаёт интерфейс узла сети (IP адрес или DNS имя), используемый в процессе авторегистрации узла сети. Это значение будет использовано для заполнения интерфейса у свежесоздаваемого узла сети, можно явно настроить либо IP, либо DNS адрес. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Если не задано, значение будет получено из HostInterfaceItem.

Агент выдаст ошибку и не запустится, если значение превышает ограничение в 255 символов.

Диапазон: 0–255 символов.

HostInterfaceItem

Опциональный параметр, который задаёт элемент данных, используемый для получения интерфейса узла сети (IP адрес либо DNS имя) в процессе авторегистрации узла сети. Это значение используется, только если HostInterface не определён. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Во время запроса на авторегистрацию агент будет выводить в журнал предупреждение, если значение, возвращаемое указанным элементом данных, превышает ограничение в 255 символов.

Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

HostMetadata

Опциональный параметр, который задаёт метаданные узла сети, используемые, чтобы идентифицировать или отличить узел сети в процессе авторегистрации узлов сети. HostMetadata позволяет различать узлы сети не только по имени хоста.

Если не определено, то значение берётся от HostMetadataItem.

Агент выдаст ошибку и не запустится, если указанное значение превышает ограничение 2034 байта или не является UTF-8 строкой. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет усечён по первому переводу строки.

Диапазон: 0–2034 байта.

HostMetadataItem

Опциональный параметр, который задаёт элемент данных, используемый для получения метаданных узла сети. Этот параметр используется только если HostMetadata не определён.

Значение HostMetadataItem извлекается на каждую попытку авторегистрации и используется только в процессе автоматической регистрации узлов сети. HostMetadata позволяет различать узлы сети не только по имени хоста.

Поддерживаются UserParameters и алиасы. Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

При запросе на авторегистрацию агент запишет в журнал предупреждение, если полученное от указанного элемента данных значение превышает ограничение в 65535 кодовых позиций (code points) UTF-8. Значение, возвращаемое указанным элементом данных, должно являться UTF-8 строкой, в противном случае оно будет игнорироваться. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет усечён по первому переводу строки.

Hostname

Список (через запятую) уникальных, регистрозависимых имён узла сети. Требуется для активных проверок и должно совпадать с именами узла сети, указанными на сервере. Если не задано, значение будет браться из HostnameItem.<br>Допустимые символы: алфавитно-цифровые, «.», « », «_» и «-». Максимальная длина: 128 символов на имя узла сети, 2048 символов для всей строки.

По умолчанию: Задаётся параметром HostnameItem

HostnameItem

Опциональный параметр, который задаёт элемент данных Zabbix агента, используемый для получения имени узла сети. Этот параметр используется, только если Hostname не определён. Не поддерживает UserParameters или алиасы, но поддерживает system.run[], независимо от значений AllowKey/DenyKey.

По умолчанию: `system.hostname`

Include

Вы можете включить отдельные файлы или все файлы в каталоге в файл конфигурации. В процессе инсталляции Zabbix создаст директорию для включения файлов в /usr/local/etc, если только она не была изменена во время компиляции. Путь может быть относительным по отношению к местоположению файла *zabbix_agent2.conf*.<br>Чтобы включить в указанный каталог только соответствующие файлы, для сопоставления с шаблоном поддерживается подстановочный знак звёздочки.<br>Смотрите специальные заметки по поводу ограничений.

Пример:

```
Include=/      /   / /    /    /*.conf
```

ListenIP

Список (через запятую) IP адресов, которые должен слушать агент. Первый IP адрес отсылается на Zabbix сервер при подключении к нему для получения списка активных проверок.

По умолчанию: `0.0.0.0`

ListenPort

Агент будет слушать этот порт на предмет подключений с сервера.

По умолчанию: 10050<br> Диапазон: 1024-32767

LogFile

Имя файла журнала.

По умолчанию: `/tmp/zabbix_agent2.log`<br> Обязательное: Да, если LogType выставлен в *file*; в противном случае — нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключение автоматической ротации журнала.<br>*Примечание*: Если лимит размера файла достигнут, а ротация по каким-либо причинам не удалась, существующий файл журнала усекается и начинается заново.

По умолчанию: 1<br> Диапазон: 0-1024<br>

Тип журнала

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*system* — запись журнала в syslog;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: `file`

PersistentBufferFile

Файл, в котором Zabbix агент 2 должен хранить базу данных SQLite. Имя файла должно быть указано с абсолютным путём. Этот параметр используется, только если включён постоянный буфер (*EnablePersistentBuffer=1*).

PersistentBufferPeriod

Период времени, в течение которого должны храниться данные, когда нет соединения с сервером или прокси. Более старые данные будут потеряны. Журнальные данные будут сохранены. Этот параметр используется, только если включён постоянный буфер (*EnablePersistentBuffer=1*).

По умолчанию: 1h<br> Диапазон: 1m-365d

PidFile

Имя PID файла.

По умолчанию: `/tmp/zabbix_agent2.pid`

Plugins.<ИмяПлагина>.System.Capacity

Ограничение количества проверок на плагин < >, которые могут выполняться одновременно.

По умолчанию: 1000 Диапазон: 1-1000

Plugins.Log.MaxLinesPerSecond

Максимальное количество новых строк в секунду, которые агент будет отправлять Zabbix серверу или прокси при обработке активных проверок «log» и «logrt». Указаное значение будет переопределено параметром «maxlines», указанным в ключах элементов данных «log» и «logrt».<br>*Примечание*: при поиске нужной строки в элементах данных журналов Zabbix будет обрабатывать в 10 раз больше новых строк, чем указано в параметре *MaxLinesPerSecond*.

По умолчанию: 20<br> Диапазон: 1-1000

Plugins.SystemRun.LogRemoteCommands

Разрешение журналировать выполняемые команды оболочки как предупреждения. Команды будут записаны в журнал, только если они выполняются удалённо. Записи в журнале не создаются, если system.run[] запускается локально параметрами HostMetadataItem, HostInterfaceItem или HostnameItem.

По умолчанию: 0<br> Значения: 0 — запрещено, 1 — разрешено

PluginSocket

Путь к файлу Unix сокета для коммуникаций с подгружаемыми плагинами.

По умолчанию: `/tmp/agent.plugin.sock`

PluginTimeout

Тайм-аут для соединений с подгружаемыми плагинами, в секундах.

По умолчанию: `Timeout`<br> Диапазон: 1-30

RefreshActiveChecks

Как часто обновлять список активных проверок, в секундах. Обратите внимание, что после неуспешного обновления активных проверок, следующая попытка будет предпринята через 60 секунд.

По умолчанию: 5<br> Диапазон: 1-86400

Server

Список (через запятую) IP адресов, опционально в CIDR нотации, или DNS имён хостов Zabbix серверов или Zabbix прокси. Входящие соединения будут приниматься только с хостов, указанных в этом списке. Если включена поддержка IPv6, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» обрабатываются одинаково, а «::/0» разрешает все IPv4 и IPv6 адреса.

«0.0.0.0/0» можно использовать, чтобы разрешить любой IPv4 адрес. Пробелы разрешены. Если этот параметр не указан, пассивные проверки отключаются, и агент не будет прослушивать ни один порт TCP.

Пример:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Адрес Zabbix сервера/прокси или конфигурация кластера для получения активных проверок. Адрес сервера/прокси — это IP адрес или DNS имя и опционально порт, разделённые двоеточием.<br>Конфигурация кластера — один или более адресов серверов либо члено группы прокси, разделённые символом «точка с запятой». Можно указать несколько серверов/кластеров/прокси Zabbix, разделяя их запятой. От каждого сервера/кластера Zabbix должно быть указано не более одного Zabbix прокси, если только не используются группы прокси. Если указан Zabbix прокси, то сервер/кластер Zabbix для этого прокси не должны быть указаны.<br>Несколько адресов через запятую можно указать для использования нескольких независимых серверов Zabbix параллельно. Пробелы разрешены.<br>Если порт не указан, используется порт по умолчанию.<br>Адреса IPv6 должны быть заключены в квадратные скобки, если для этого хоста указан порт. Если порт не указан, квадратные скобки для адресов IPv6 опциональны.<br>Если этот параметр не указан, активные проверки отключены.

Пример для Zabbix прокси:

```
ServerActive=127.0.0.1:10051
```

Пример для группы Zabbix прокси:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Пример для нескольких серверов:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Пример для кластера высокой доступности:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Пример для конфигурации высокой доступности с двумя кластерами и одним сервером:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

IP-адрес источника

Локальный IP адрес для:

- исходящих подключений к Zabbix серверу или прокси;
- создания подключений при выполнении некоторых элементов данных (web.page.get, net.tcp.port и т.п.).

StatusPort

Если задан, агент будет слушать указанный порт для HTTP запросов состояния (http://localhost:<порт>/status).

Диапазон: 1024-32767

Timeout

Указывает, как долго (в секундах) следует ожидать установления соединения и обмена данными с Zabbix сервером или прокси.<br>

Этот параметр определяет продолжительность различных коммуникационных операций, в том числе:

- ожидание ответа от Zabbix сервера;
- отправку запросов на Zabbix сервер, в том числе запросы конфигурации элементов данных и значения элементов данных в активных проверках;
- получение данных журнала посредством мониторинга файла журнала;
- отправка сообщений heartbeat;
- максимальная длительность для проверок `vfs.*`;
- служит в качестве запасного варианта в сценариях, где сервер или прокси версии старше 7.0 отправляет проверки без тайм-аутов.

Этот тайм-аут **не** будет использоваться для тех проверок агента, которые имеют настраиваемые параметры тайм-аута в веб-интерфейсе (на глобальном уровне, на уровне прокси-сервера или на уровне отдельных элементов данных).

По умолчанию: 3<br> Диапазон: 1-30

TLSAccept

Какие входящие соединения принимать. Используется для пассивных проверок. Можно указать несколько значений через запятую:<br>*unencrypted* — принимать соединения без шифрования (по умолчанию)<br>*psk* — принимать подключения с TLS и общим ключом (pre-shared key, PSK)<br>*cert* — принимать подключения с TLS и сертификатом

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже для *незашифрованных* подключений), в противном случае — нет

TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Абсолютный путь к файлу, который содержит сертификат агента или цепочку сертификатов, используется для зашифрованных соединений между компонентами Zabbix.

TLSCipherAll

Строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK.

Пример:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherAll13

Строка шифров OpenSSL (TLS 1.3) для OpenSSL 1.1.1 или более новой версии. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK.

Пример:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert

Строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов.

Пример:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

Обратите внимание, что этот параметр нельзя использовать вместе с `TLSAccept=cert,psk`; для подключения по сертификату (`TLSConnect=cert`) вместо него используйте TLSCipherAll.

TLSCipherCert13

Строка шифров OpenSSL (TLS 1.3) для OpenSSL 1.1.1 или новее. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов.

Обратите внимание, что этот параметр нельзя использовать вместе с `TLSAccept=cert,psk`; для подключения по сертификату (`TLSConnect=cert`) вместо него используйте TLSCipherAll13.

TLSCipherPSK

Строка шифров OpenSSL (TLS 1.2). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK.

Пример:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Строка шифров OpenSSL (TLS 1.3) для OpenSSL 1.1.1 или более новой версии. Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK.

Пример:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Как агент должен подключаться к Zabbix серверу или прокси. Используется для активных проверок. Можно указать только одно значение:<br>*unencrypted* — подключаться без шифрования (по умолчанию)<br>*psk* — подключаться, используя TLS и общий ключ (pre-shared key, PSK)<br>*cert* — подключаться, используя TLS и сертификат

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже при *незашифрованном* соединении), в противном случае — нет

TLSCRLFile

Абсолютный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ агента (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSPSKFile

Абсолютный путь к файлу, содержащему общий ключ (pre-shared key) агента. Используется для зашифрованных коммуникаций с сервером Zabbix.

TLSPSKIdentity

Строка идентификатора PSK, используется для зашифрованных соединений с Zabbix сервером.

TLSServerCertIssuer

Разрешённый эмитент сертификата сервера (прокси).

TLSServerCertSubject

Разрешённый субъект сертификата сервера (прокси).

UnsafeUserParameters

Разрешить передавать аргументами в пользовательские параметры все символы. Не разрешены следующие символы: \ ' " ` * ? [ ] { } ~ $ ! & ; ( ) < > | # @<br>Кроме того, не разрешены символы новой строки.

По умолчанию: 0<br> Значения: 0 — не разрешать, 1 — разрешить

UserParameter

Пользовательский параметр для мониторинга. Можно указать несколько пользовательских параметров.<br>Формат: UserParameter=<ключ>,<команда оболочки><br>Обратите внимание, что команда не должна возвращать только пустую строку или EOL. Команды оболочки могут иметь относительные пути, если указан параметр UserParameterDir.

Пример:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Путь поиска по умолчанию для команд UserParameter. Если используется, агент перед выполнением команды сменит свою рабочую директорию на указанную здесь. Таким образом, команды UserParameter могут иметь относительный префикс `./` вместо полного пути.<br>Допустимо только одно значение.

Пример:

```
UserParameterDir=/opt/myscripts
```

### 5. Zabbix агент (Windows)

Обзор

Параметры, поддерживаемые файлом конфигурации агента Zabbix для Windows (zabbix_agentd.conf), перечислены в этом разделе.

Параметры перечислены без дополнительной информации. Щелкните параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| Alias | Задает псевдоним для ключа элемента данных. |
| AllowKey | Разрешает выполнение тех ключей элементов данных, которые соответствуют шаблону. |
| BufferSend | Не хранить данные в буфере дольше N секунд. |
| BufferSize | Максимальное количество значений в буфере памяти. |
| DebugLevel | Уровень отладки. |
| DenyKey | Запрещает выполнение тех ключей элементов данных, которые соответствуют шаблону. |

| Parameter | Description |
|---|---|
| EnableRemoteCommands | Разрешены ли удаленные команды от сервера Zabbix. |
| HeartbeatFrequency | Частота сообщений heartbeat в секундах. |
| HostInterface | Необязательный параметр, который определяет интерфейс узла сети. |
| HostInterfaceItem | Необязательный параметр, который определяет элемент данных, используемый для получения интерфейса узла сети. |
| HostMetadata | Необязательный параметр, который определяет метаданные узла сети. |
| HostMetadataItem | Необязательный параметр, который определяет элемент данных агента Zabbix, используемый для получения метаданных узла сети. |
| Hostname | Необязательный параметр, который определяет имя узла сети. |
| HostnameItem | Необязательный параметр, который определяет элемент данных агента Zabbix, используемый для получения имени узла сети. |
| Include | В файл конфигурации можно включать отдельные файлы или все файлы в каталоге. |
| ListenBacklog | Максимальное количество ожидающих соединений в очереди TCP. |
| ListenIP | Список IP-адресов, разделенных запятыми, на которых должен прослушивать агент. |
| ListenPort | Агент будет прослушивать этот порт для соединений от сервера. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogRemoteCommands | Включить запись выполненных команд оболочки как предупреждений. |
| LogType | Тип вывода журнала. |
| MaxLinesPerSecond | Максимальное количество новых строк, которое агент будет отправлять в секунду на сервер Zabbix или прокси при обработке активных проверок ‘log’ и ‘logrt’. |
| PerfCounter | Определяет новый параметр <parameter_name>, который является средним значением системного счетчика производительности <perf_counter_path> за указанный период времени <period> (в секундах). |
| PerfCounterEn | Определяет новый параметр <parameter_name>, который является средним значением системного счетчика производительности <perf_counter_path> за указанный период времени <period> (в секундах). По сравнению с PerfCounter, пути perfcounter должны быть на английском языке. |
| RefreshActiveChecks | Как часто обновляется список активных проверок. |
| Server | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен серверов Zabbix и прокси Zabbix. |
| ServerActive | Адрес сервера Zabbix/прокси или конфигурация кластера, откуда получать активные проверки. |
| SourceIP | Исходный IP-адрес. |
| StartAgents | Количество предварительно созданных экземпляров zabbix_agentd, которые обрабатывают пассивные проверки. |
| Timeout | Указывает, сколько времени ожидать (в секундах) установления соединения и обмена данными с прокси Zabbix или сервером. |
| TLSAccept | Какие входящие соединения принимать. |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты CA верхнего уровня для проверки сертификата узла, используемые для зашифрованной связи между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов, используемые для зашифрованной связи между компонентами Zabbix. |
| TLSConnect | Как агент должен подключаться к серверу Zabbix или прокси. |
| TLSCRLFile | Полный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованной связи между компонентами Zabbix. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ агента, используемый для зашифрованной связи между компонентами Zabbix. |
| TLSPSKFile | Полный путь к файлу, содержащему предварительно общий ключ агента, используемый для зашифрованной связи с сервером Zabbix. |
| TLSPSKIdentity | Строка идентификатора предварительно общего ключа, используемая для зашифрованной связи с сервером Zabbix. |
| TLSServerCertIssuer | Разрешенный издатель сертификата сервера (прокси). |
| TLSServerCertSubject | Разрешенный субъект сертификата сервера (прокси). |
| UnsafeUserParameters | Разрешить передачу всех символов в аргументах пользовательских параметров. |
| UserParameter | Пользовательский параметр для мониторинга. |
| UserParameterDir | Путь поиска по умолчанию для команд UserParameter. |

Все параметры являются необязательными, если явно не указано, что параметр обязателен.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию демона, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды.
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с "#", поддерживаются только в начале строки.

Подробности параметра

Alias

Задаёт алиас для ключа элемента данных. Его можно использовать для замены длинных и сложных ключей элементов данных на более короткие и простые.<br> Можно добавлять несколько параметров *Alias*. Несколько параметров с одинаковым ключом *Alias* не разрешены.<br> Различные ключи параметра *Alias* могут ссылаться на тот же ключ элемента данных.<br> Алиасы можно использовать в параметре *HostMetadataItem*, но нельзя в параметре *HostnameItem* или *PerfCounter*.

Пример 1: Получение с сервера утилизации файла подкачки в процентах.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Теперь для получения данных можно использовать сокращённый ключ **pg_usage**.

Пример 2: Получение утилизации CPU с параметрами по умолчанию и с пользовательскими параметрами.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Такая запись позволяет использовать ключ **cpu.load** для получения утилизации CPU в процентах с параметрами по умолчанию, а также использовать **cpu.load[percpu,avg15]** для получения конкретных данных об утилизации CPU.

Пример 3: Выполнение нескольких правил низкоуровневого обнаружения, которые обрабатывают одни и те же элементы данных обнаружения.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Теперь имеется возможность настроить несколько правил обнаружения, используя **vfs.fs.discovery** с разными параметрами для каждого правила, например, **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** и т.д.

AllowKey

Разрешение выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с DenyKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

BufferSend

Не хранить данные в буфере дольше N секунд.

По умолчанию: 5<br> Диапазон: 1-3600

BufferSize

Максимальное количество значений в буфере памяти. При заполнении буфера агент будет отправлять все собранные данные Zabbix серверу или прокси.

По умолчанию: 100<br> Диапазон: 2-65535

DebugLevel

Укажите уровень отладки:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критическая информация;<br>*2* — информация об ошибках;<br>*3* — предупреждения; <br>*4* — для отладки (выдает много информации);<br>*5* — расширенная отладка (выдает еще больше информации).

По умолчанию: 3<br> Диапазон: 0-5

DenyKey

Запрет выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с AllowKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

EnableRemoteCommands

Разрешены ли удалённые команды с Zabbix сервера. Этот параметр устарел, используйте вместо него AllowKey=system.run[*] или DenyKey=system.run[*]. <br>Это внутренний алиас для параметров AllowKey/DenyKey, зависящий от значения:<br>0 — DenyKey=system.run[*]<br>1 — AllowKey=system.run[*]

По умолчанию: 0<br> Значения: 0 — не разрешены; 1 — разрешены

HeartbeatFrequency

Частота контрольных сообщений (heartbeat) в секундах. Используется для контроля доступности активных проверок.<br>0 — контрольные сообщения отключены.

По умолчанию: 60<br> Диапазон: 0-3600

HostInterface

Опциональный параметр, который задаёт интерфейс узла сети (IP адрес либо DNS имя), используемый в процессе авторегистрации узла сети. Это значение будет использовано для заполнения интерфейса у свежесоздаваемого узла сети, можно явно настроить либо IP, либо DNS адрес. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Если не задано, значение будет получено из HostInterfaceItem.

Агент выдаст ошибку и не запустится, если значение превышает ограничение в 255 символов.

Диапазон: 0–255 символов.

HostInterfaceItem

Опциональный параметр, который задаёт элемент данных, используемый для получения интерфейса узла сети (IP адрес либо DNS имя) в процессе авторегистрации узла сети. Это значение используется, только если HostInterface не определён. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Во время запроса на авторегистрацию агент будет выводить в журнал предупреждение, если значение, возвращаемое указанным элементом данных, превышает ограничение в 255 символов.

Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

HostMetadata

Опциональный параметр, который задаёт метаданные узла сети, используемые, чтобы идентифицировать или отличить узел сети в процессе авторегистрации узлов сети (активный агент). HostMetadata позволяет различать узлы сети не только по имени хоста.

Если не определено, то значение берётся от HostMetadataItem.

Агент выдаст ошибку и не запустится, если указанное значение превышает ограничение 2034 байта или не является UTF-8 строкой. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет обрезан по первому переводу строки.

Диапазон: 0–2034 байта.

HostMetadataItem

Опциональный параметр, который задаёт элемент данных Zabbix агента, используемый для получения метаданных узла сети. Этот параметр используется, только если HostMetadata не определён.

Значение HostMetadataItem извлекается на каждую попытку авторегистрации и используется только в процессе автоматической регистрации узлов сети (активный агент). HostMetadata позволяет различать узлы сети не только по имени хоста.

Поддерживаются UserParameters, счётчики производительности (performance counters) и алиасы. Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

При запросе на авторегистрацию агент запишет в журнал предупреждение, если полученное от указанного элемента данных значение превышает ограничение в 65535 кодовых позиций (code points) UTF-8. Значение, возвращаемое указанным элементом данных, должно являться UTF-8 строкой, в противном случае оно будет игнорироваться. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет обрезан по первому переводу строки.

Hostname

Список (через запятую) уникальных, регистрозависимых имён узла сети. Требуется для активных проверок и должно совпадать с именами узла сети, указанными на сервере. Если не задано, значение будет браться из HostnameItem.<br>Допустимые символы: алфавитно-цифровые, «.», « », «_» и «-». Максимальная длина: 128 символов на имя узла сети, 2048 символов для всей строки.

По умолчанию: Задаётся параметром HostnameItem

HostnameItem

Опциональный параметр, который задаёт элемент данных Zabbix агента, используемый для получения имени узла сети. Этот параметр используется, только если Hostname не определён. Не поддерживает UserParameters, счётчики производительности (performance counters) или алиасы, но поддерживает system.run[], независимо от значений AllowKey/DenyKey.<br>Смотрите также более подробное описание.

По умолчанию: `system.hostname`

Include

Вы можете включить отдельные файлы или все файлы в каталоге в файл конфигурации (расположен по умолчанию в папке `C:\Program Files\Zabbix Agent` , если Zabbix агент устанавливается с использованием установщика пакетов Windows MSI; расположен в папке, указанной во время установки, если Zabbix агент устанавливается как zip-архив). Все включаемые файлы должны иметь корректный синтаксис, в противном случае агент не запустится.<br>Для включения только необходимых файлов из указанной папки, поддерживается символ звёздочки для поиска совпадения по маске.<br>Смотрите специальные заметки по поводу ограничений.

Пример:

```
Include=C:\Program Files\Zabbix Agent\zabbix_agentd.d\*.conf
```

ListenBacklog

Максимальное количество соединений в состоянии ожидания в очереди TCP.<br>Значение по умолчанию является зашитой в коде константой, которая зависит от системы.<br>Максимально поддерживаемое значение зависит от системы, слишком большие значения могу быть молча урезаны до «указанного в реализации максимума» (’implementation-specified maximum’).

По умолчанию: `SOMAXCONN`<br> Диапазон: 0 – INT_MAX

ListenIP

Список (через запятую) IP адресов, которые должен слушать агент.

По умолчанию: `0.0.0.0`

ListenPort

Агент будет слушать этот порт на предмет подключений с сервера.

По умолчанию: 10050<br> Диапазон: 1024-32767

LogFile

Имя файла журнала агента.

По умолчанию: `C:\zabbix_agentd.log`<br> Обязательное: Да, если для LogType задан как *file*; иначе — нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключение автоматической ротации журнала.<br>*Примечание*: Если лимит размера файла достигнут, а ротация по каким-либо причинам не удалась, существующий файл журнала усекается и начинается заново.

По умолчанию: 1<br> Диапазон: 0-1024<br>

LogRemoteCommands

Разрешение журналировать выполняемые команды оболочки как предупреждения. Команды будут записаны в журнал, только если они выполняются удалённо. Записи в журнале не создаются, если system.run[] запускается локально параметрами HostMetadataItem, HostInterfaceItem или HostnameItem.

По умолчанию: 0<br> Значения: 0 — запрещено, 1 — разрешено

LogType

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*system* — запись в журнал событий Windows;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: `file`

MaxLinesPerSecond

Максимальное количество новых строк в секунду, которые агент будет отправлять Zabbix серверу или прокси при обработке активных проверок «log», «logrt» и «eventlog». Указаное значение будет переопределено параметром «maxlines», указанным в ключах элементов данных «log», «logrt» и «eventlog».<br>*Примечание*: при поиске нужной строки в элементах данных журналов Zabbix будет обрабатывать в 10 раз больше новых строк, чем указано в параметре *MaxLinesPerSecond*.

По умолчанию: 20<br> Диапазон: 1-1000

PerfCounter

Определяет новый параметр <имя_параметра>, который который является усреднённым значением системного счётчика производительности <путь_к_счётчику_производительности> за указанный период времени <период> (в секундах).<br>Синтаксис: <имя_параметра>,"<путь_к_счётчику_производительности>",<период>

Например, если хотите получать среднее количество прерываний процессора в секунду за последнюю минуту, вы можете задать новый параметр «interrupts» следующим образом:<br>

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Пожалуйста, обратите внимание на двойные кавычки вокруг пути счётчика производительности. Имя параметра (interrupts) используется как ключ элемента данных при создании элемента данных. Выборки для вычисления среднего значения будут браться каждую секунду.<br>Вы можете выполнить «typeperf -qx», чтобы получить список всех счётчиков производительности, доступных в Windows.

PerfCounterEn

Определяет новый параметр <имя_параметра>, который является усреднённым значением системного счётчика производительности <путь_к_счётчику_производительности> за указанный период времени <период> (в секундах). По сравнению с PerfCounter, пути к счётчику производительности должны быть записаны по-английски. Поддерживается только на **Windows Server 2008/Vista** и выше.<br>Синтаксис: <имя_параметра>,"<путь_к_счётчику_производительности>",<пер

Например, если хотите получать среднее количество прерываний процессора в секунду за последнюю минуту, вы можете задать новый параметр «interrupts» следующим образом:<br>

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Пожалуйста, обратите внимание на двойные кавычки вокруг пути счётчика производительности. Имя параметра (interrupts) используется как ключ элемента данных при создании элемента данных. Выборки для расчёта среднего значения будут браться каждую секунду.<br>Вы можете найти список английских строк путём просмотра следующего ключа реестра: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

RefreshActiveChecks

Как часто обновлять список активных проверок, в секундах. Обратите внимание, что после неуспешного обновления активных проверок, следующая попытка будет предпринята через 60 секунд.

По умолчанию: 5<br> Диапазон: 1-86400

Server

Список (через запятую) IP адресов, опционально в CIDR нотации, или DNS имён хостов Zabbix серверов и Zabbix прокси. Входящие соединения будут приниматься только с хостов, указанных в этом списке. Если включена поддержка IPv6, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» обрабатываются одинаково, а «::/0» разрешает все IPv4 и IPv6 адреса. «0.0.0.0/0» можно использовать, чтобы разрешить любой IPv4 адрес. Обратите внимание, что «IPv4-совместимые IPv6 адреса» (префикс 0000::/96) поддерживаются, но являются устаревшими согласно [RFC4291 [en]] (https://tools.ietf.org/html/rfc4291#section-2.5.5). Пробелы разрешены.

Пример:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Обязательное: да, если для StartAgents явно не установлено значение 0.

ServerActive

Адрес Zabbix сервера/прокси или конфигурация кластера для получения активных проверок. Адрес сервера/прокси — это IP адрес или DNS имя и опционально порт, разделённые двоеточием.<br>Конфигурация кластера — один или более адресов серверов или членов группы прокси, разделённые символом «точка с запятой». Можно указать несколько серверов/кластеров/прокси Zabbix, разделяя их запятой. От каждого сервера/кластера Zabbix должно быть указано не более одного Zabbix прокси, если только не используются группы прокси. Если указан Zabbix прокси, то

сервер/кластер Zabbix для этого прокси не должны быть указаны.<br>Несколько адресов через запятую можно указать для использования нескольких независимых серверов Zabbix параллельно. Пробелы разрешены.<br>Если порт не указан, используется порт по умолчанию.<br>Адреса IPv6 должны быть заключены в квадратные скобки, если для этого хоста указан порт. Если порт не указан, квадратные скобки для адресов IPv6 опциональны.<br>Если этот параметр не указан, активные проверки отключены.

Пример для Zabbix прокси:

```
ServerActive=127.0.0.1:10051
```

Пример для группы Zabbix прокси:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Пример для нескольких серверов:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Пример для кластера высокой доступности:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Пример для конфигурации высокой доступности с двумя кластерами и одним сервером:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

Диапазон: (*)

SourceIP

Локальный IP адрес для:

- исходящих подключений к Zabbix серверу или прокси;
- создания подключений при выполнении некоторых элементов данных (web.page.get, net.tcp.port и т.п.).

StartAgents

Количество экземпляров запускаемых при старте процессов zabbix_agentd, обрабатывающих пассивные проверки. Если выставлено в 0, отключает пассивные проверки и агент не будет слушать никакие TCP порты.

По умолчанию: 10<br> Диапазон: 0–100 (*)

Timeout

Указывает, как долго (в секундах) следует ожидать установления соединения и обмена данными с Zabbix сервером или прокси.<br>

Этот параметр определяет продолжительность различных коммуникационных операций, в том числе:

- ожидание ответа от Zabbix сервера;
- отправку запросов на Zabbix сервер, в том числе запросы конфигурации элементов данных и значения элементов данных в активных проверках;
- получение данных журнала посредством мониторинга файла журнала или журнала событий Windows;
- отправка сообщений heartbeat;
- максимальная длительность для проверок **vfs.**;
- используется модулями агента Zabbix;
- служит в качестве запасного варианта в сценариях, где сервер или прокси версии старше 7.0 отправляет проверки без тайм-аутов.

Этот тайм-аут **не** будет использоваться для тех проверок агента, которые имеют настраиваемые параметры тайм-аута в веб-интерфейсе (на глобальном уровне, на уровне прокси-сервера или на уровне отдельных элементов данных).

По умолчанию: 3<br> Диапазон: 1-30

TLSAccept

Какие входящие соединения принимать. Используется для пассивных проверок. Можно указать несколько значений через запятую:<br>*unencrypted* — принимать соединения без шифрования (по умолчанию)<br>*psk* — принимать подключения с TLS и общим ключом (pre-shared key, PSK)<br>*cert* — принимать подключения с TLS и сертификатом

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже для *незашифрованных* подключений), в противном случае — нет

TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Абсолютный путь к файлу, который содержит сертификат агента или цепочку сертификатов, используется для зашифрованных соединений между компонентами Zabbix.

TLSConnect

Как агент должен подключаться к Zabbix серверу или прокси. Используется для активных проверок. Можно указать только одно значение:<br>*unencrypted* — подключаться без шифрования (по умолчанию)<br>*psk* — подключаться, используя TLS и общий ключ (pre-shared key, PSK)<br>*cert* — подключаться, используя TLS и сертификат

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже при *незашифрованном* соединении), в противном случае — нет

TLSCRLFile

Абсолютный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ агента (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSPSKFile

Абсолютный путь к файлу, содержащему общий ключ (pre-shared key) агента. Используется для зашифрованных коммуникаций с сервером Zabbix.

TLSPSKIdentity

Строка идентификатора PSK, используется для зашифрованных соединений с Zabbix сервером.

TLSServerCertIssuer

Разрешённый эмитент сертификата сервера (прокси).

TLSServerCertSubject

Разрешённый субъект сертификата сервера (прокси).

UnsafeUserParameters

Разрешить передавать аргументами в пользовательские параметры все символы. Не разрешены следующие символы: \ ' " ` * ? [ ] { } ~ $ ! & ; ( ) < > | # @<br>Кроме того, не разрешены символы новой строки.

По умолчанию: 0<br> Значения: 0 — не разрешать, 1 — разрешить

UserParameter

Пользовательский параметр для мониторинга. Можно указать несколько пользовательских параметров.<br>Формат: UserParameter=<ключ>,<команда оболочки><br>Обратите внимание, что команда не должна возвращать только пустую строку или EOL. Команды оболочки могут иметь относительные пути, если указан параметр UserParameterDir.

Пример:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Путь поиска по умолчанию для команд UserParameter. Если используется, агент перед выполнением команды сменит свою рабочую директорию на указанную здесь. Таким образом, команды UserParameter могут иметь относительный префикс `./` вместо полного пути. Допустимо только одно значение.

Пример:

```
UserParameterDir=C:\opt\myscripts
```

**Note:**
(*) Количество активных серверов, перечисленных в параметре ServerActive, плюс количество запускаемых при старте экземпляров для пассивных проверок, указанное в StartAgents, должно быть меньше 64.

## 6. Zabbix агент 2 (Windows)

Обзор

Zabbix agent 2 — это агент Zabbix нового поколения, который может использоваться вместо Zabbix agent.

Параметры, поддерживаемые файлом конфигурации Zabbix agent 2 для Windows (zabbix_agent2.conf), перечислены в этом разделе.

Параметры перечислены без дополнительной информации. Нажмите на параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| Alias | Задает псевдоним для ключа элемента данных. |
| AllowKey | Разрешает выполнение тех ключей элементов данных, которые соответствуют шаблону. |
| BufferSend | Не хранить данные в буфере дольше N секунд. |
| BufferSize | Максимальное количество значений в буфере памяти. |
| ControlSocket | Управляющий сокет, используемый для отправки команд во время выполнения с помощью опции '-R'. |
| DebugLevel | Уровень отладки. |
| DenyKey | Запрещает выполнение тех ключей элементов данных, которые соответствуют шаблону. |
| EnablePersistentBuffer | Включает использование локального постоянного хранилища для активных элементов данных. |
| ForceActiveChecksOnStart | Выполняет активные проверки сразу после перезапуска для первой полученной конфигурации. |
| HeartbeatFrequency | Частота сообщений heartbeat в секундах. |
| HostInterface | Необязательный параметр, определяющий интерфейс узла сети. |
| HostInterfaceItem | Необязательный параметр, определяющий элемент данных, используемый для получения интерфейса узла сети. |
| HostMetadata | Необязательный параметр, определяющий метаданные узла сети. |
| HostMetadataItem | Необязательный параметр, определяющий элемент данных агента Zabbix, используемый для получения метаданных узла сети. |
| Hostname | Необязательный параметр, определяющий имя узла сети. |
| HostnameItem | Необязательный параметр, определяющий элемент данных агента Zabbix, используемый для получения имени узла сети. |
| Include | В файл конфигурации можно включать отдельные файлы или все файлы из каталога. |
| ListenIP | Список IP-адресов, разделенных запятыми, на которых агент должен прослушивать подключения. |
| ListenPort | Агент будет прослушивать этот порт для подключений от сервера. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogType | Тип вывода журнала. |
| PersistentBufferFile | Файл, в котором Zabbix agent 2 должен хранить базу данных SQLite. |
| PersistentBufferPeriod | Период времени, в течение которого данные должны храниться при отсутствии соединения с сервером или прокси. |
| Plugins.<PluginName>.System.Capacity | Ограничение на количество проверок на плагин, которые могут выполняться одновременно. |
| Plugins.Log.MaxLinesPerSecond | Максимальное количество новых строк, которое агент будет отправлять в секунду на сервер Zabbix или прокси при обработке активных проверок 'log' и 'logrt'. |
| Plugins.SystemRun.LogRemoteCommands | Включает ведение журнала выполненных команд оболочки как предупреждений. |
| PluginSocket | Путь к именованному каналу Windows для взаимодействия загружаемых плагинов. |
| PluginTimeout | Тайм-аут для подключений к загружаемым плагинам, в секундах. |
| PerfCounter | Определяет новый параметр <parameter_name>, который является средним значением счетчика производительности системы <perf_counter_path> за указанный период времени <period> (в секундах). |
| PerfCounterEn | Определяет новый параметр <parameter_name>, который является средним значением счетчика производительности системы <perf_counter_path> за указанный период времени <period> (в секундах). По сравнению с PerfCounter, пути perfcounter должны быть на английском языке. |
| RefreshActiveChecks | Как часто обновляется список активных проверок. |
| Server | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен серверов Zabbix и прокси Zabbix. |
| ServerActive | Адрес сервера Zabbix/прокси или конфигурация кластера, откуда получать активные проверки. |

| Parameter | Description |
|---|---|
| SourceIP | Исходный IP-адрес. |
| StatusPort | Если задан, агент будет прослушивать этот порт для HTTP-запросов состояния (http://localhost:<port>/status). |
| Timeout | Указывает, сколько времени ожидать (в секундах) установления соединения и обмена данными с прокси или сервером Zabbix. |
| TLSAccept | Какие входящие подключения принимать. |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты корневых CA для проверки сертификата узла, используемые для зашифрованной связи между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов, используемые для зашифрованной связи между компонентами Zabbix. |
| TLSCipherAll13 | Строка шифров OpenSSL (TLS 1.3). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK. |
| TLSCipherCert13 | Строка шифров OpenSSL (TLS 1.3). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов. |
| TLSCipherPSK13 | Строка шифров OpenSSL (TLS 1.3). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK. |
| TLSConnect | Способ подключения агента к серверу Zabbix или прокси. |
| TLSCRLFile | Полный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованной связи между компонентами Zabbix. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ агента, используемый для зашифрованной связи между компонентами Zabbix. |
| TLSPSKFile | Полный путь к файлу, содержащему предварительно общий ключ агента, используемый для зашифрованной связи с сервером Zabbix. |
| TLSPSKIdentity | Строка идентификатора предварительно общего ключа, используемая для зашифрованной связи с сервером Zabbix. |
| TLSServerCertIssuer | Разрешенный издатель сертификата сервера (прокси). |
| TLSServerCertSubject | Разрешенный субъект сертификата сервера (прокси). |
| UnsafeUserParameters | Разрешает передавать все символы в аргументах пользовательских параметров. |
| UserParameter | Пользовательский параметр для мониторинга. |
| UserParameterDir | Путь поиска по умолчанию для команд UserParameter. |

Все параметры являются необязательными, если явно не указано, что параметр обязателен.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Подробности параметра

Alias

Задаёт алиас для ключа элемента данных. Его можно использовать для замены длинных и сложных ключей элементов данных на более короткие и простые.<br> Можно добавлять несколько параметров *Alias*. Несколько параметров с одинаковым ключом *Alias* не разрешены.<br> Различные ключи параметра *Alias* могут ссылаться на тот же ключ элемента данных.<br> Алиасы можно использовать в параметре *HostMetadataItem*, но нельзя в параметре *HostnameItem*.

Пример 1: Получение с сервера утилизации файла подкачки в процентах.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Теперь для получения данных можно использовать сокращённый ключ **pg_usage**.

Пример 2: Получение утилизации CPU с параметрами по умолчанию и с пользовательскими параметрами.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Такая запись позволяет использовать ключ **cpu.load** для получения утилизации CPU в процентах с параметрами по умолчанию, а также использовать **cpu.load[percpu,avg15]** для получения конкретных данных об утилизации CPU.

Пример 3: Выполнение нескольких правил низкоуровневого обнаружения, которые обрабатывают одни и те же элементы данных обнаружения.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Теперь имеется возможность настроить несколько правил обнаружения, используя **vfs.fs.discovery** с разными параметрами для каждого правила, например, **vfs.fs.discovery[foo]**, **vfs.fs.discovery[bar]** и т.д.

AllowKey

Разрешение выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с DenyKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

BufferSend

Интервал времени в секундах, определяющий, как часто значения из буфера отсылаются на Zabbix сервер.<br> Обратите внимание, что если буфер заполнен, данные будут отосланы раньше.

По умолчанию: 5<br> Диапазон: 1-3600

BufferSize

Максимальное количество значений в буфере памяти. При заполнении буфера агент будет отправлять все собранные данные Zabbix серверу или прокси. <br>Этот параметр должен использоваться, только если постоянный буфер отключён (*EnablePersistentBuffer=0*).

По умолчанию: 1000<br> Диапазон: 2-65535

ControlSocket

Сокет управления, используется при отправке команд управления при помощи опции «-R».

По умолчанию: `\\.\pipe\agent.sock`

DebugLevel

Укажите уровень отладки:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критическая информация;<br>*2* — информация об ошибках;<br>*3* — предупреждения; <br>*4* — для отладки (выдает много информации);<br>*5* — расширенная отладка (выдает еще больше информации).

По умолчанию: 3<br> Диапазон: 0-5

DenyKey

Запрет выполнения тех ключей элементов данных, которые соответствуют шаблону. Шаблон ключа — это выражение с подстановочным знаком, которое поддерживает символ «*» для соответствия с любым количеством любых символов.<br>Можно задать несколько правил соответствия ключей в сочетании с AllowKey. Параметры обрабатываются по очереди в соответствии с порядком их появления. Смотрите также: Ограничение проверок агента.

EnablePersistentBuffer

Включить использование локального постоянного хранилища для активных элементов данных. Если постоянное хранилище отключено, будет использован буфер памяти.

По умолчанию: 0<br> Значения: 0 — отключено, 1 — включено

ForceActiveChecksOnStart

Выполнить активные проверки сразу после перезапуска для первой полученной конфигурации. Также доступно как параметр конфигурации плагина, например: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

По умолчанию: 0<br> Значения: 0 — отключено, 1 — включено

HeartbeatFrequency

Частота контрольных сообщений (heartbeat) в секундах. Используется для контроля доступности активных проверок.<br>0 — контрольные сообщения отключены.

По умолчанию: 60<br> Диапазон: 0-3600

HostInterface

Опциональный параметр, который задаёт интерфейс узла сети (IP адрес или DNS имя), используемый в процессе авторегистрации узла сети. Это значение будет использовано для заполнения интерфейса у свежесоздаваемого узла сети, можно явно настроить либо IP, либо DNS адрес. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Если не задано, значение будет получено из HostInterfaceItem.

Агент выдаст ошибку и не запустится, если значение превышает ограничение в 255 символов.

Диапазон: 0–255 символов.

HostInterfaceItem

Опциональный параметр, который задаёт элемент данных, используемый для получения интерфейса узла сети (IP адрес либо DNS имя) в процессе авторегистрации узла сети. Это значение используется, только если HostInterface не определён. Для дополнительной информации смотрите Использование DNS в качестве интерфейса по умолчанию.

Во время запроса на авторегистрацию агент будет выводить в журнал предупреждение, если значение, возвращаемое указанным элементом данных, превышает ограничение в 255 символов.

Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

HostMetadata

Опциональный параметр, который задаёт метаданные узла сети, используемые, чтобы идентифицировать или отличить узел сети в процессе авторегистрации узлов сети (активный агент). HostMetadata позволяет различать узлы сети не только по имени хоста.

Если не определено, то значение берётся от HostMetadataItem.

Агент выдаст ошибку и не запустится, если указанное значение превышает ограничение 2034 байта или не является UTF-8 строкой. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет обрезан по первому переводу строки.

Диапазон: 0–2034 байта.

HostMetadataItem

Опциональный параметр, который задаёт элемент данных Zabbix агента, используемый для получения метаданных узла сети. Этот параметр используется, только если HostMetadata не определён.

Значение HostMetadataItem извлекается на каждую попытку авторегистрации и используется только в процессе автоматической регистрации узлов сети (активный агент). HostMetadata позволяет различать узлы сети не только по имени хоста.

Поддерживаются UserParameters и алиасы. Элемент данных system.run[] поддерживается независимо от значений параметров AllowKey/DenyKey.

При запросе на авторегистрацию агент запишет в журнал предупреждение, если полученное от указанного элемента данных значение превышает ограничение в 65535 кодовых позиций (code points) UTF-8. Значение, возвращаемое указанным элементом данных, должно являться UTF-8 строкой, в противном случае оно будет игнорироваться. Если параметр ожидает IP адрес или DNS имя, то значения, которые являются допустимыми значениями UTF-8, но не являются допустимыми IP адресами или DNS именами, также будут отклонены и сообщены как недействительные.

Многострочные метаданные не поддерживаются — вывод будет обрезан по первому переводу строки.

Hostname

Список (через запятую) уникальных, регистрозависимых имён узла сети. Требуется для активных проверок и должно совпадать с именами узла сети, указанными на сервере. Если не задано, значение будет браться из HostnameItem.<br>Допустимые символы: алфавитно-цифровые, «.», « », «_» и «-». Максимальная длина: 128 символов на имя узла сети, 2048 символов для всей строки.

По умолчанию: Задаётся параметром HostnameItem

HostnameItem

Опциональный параметр, который задаёт элемент данных, используемый для получения имени узла сети. Этот параметр используется, только если Hostname не определён. Не поддерживает UserParameters или алиасы, но поддерживает system.run[], независимо от значений AllowKey/DenyKey.

По умолчанию: `system.hostname`

Include

Вы можете включить отдельные файлы или все файлы в каталоге в файл конфигурации (по умолчанию расположен в папке `C:\Program Files\Zabbix Agent 2`, если Zabbix агент устанавливается с использованием установщика пакетов Windows MSI; расположен в папке, указанной во время установки, если Zabbix агент устанавливается как zip-архив). Все включаемые файлы должны иметь корректный синтаксис, в противном случае агент не запустится. Путь может быть относительным к местоположению файла *zabbix_agent2.conf* (например,`Include=.\zabbix_agent2.d\plugins.d\*.conf`).<br>Д

включения только необходимых файлов из указанной папки, поддерживается символ звёздочки для поиска совпадения по маске.<br>Смотрите специальные заметки по поводу ограничений.

Пример:

```
Include=C:\Program Files\Zabbix Agent2\zabbix_agent2.d\*.conf
```

ListenIP

Список (через запятую) IP адресов, которые должен слушать агент. Первый IP адрес отсылается на Zabbix сервер при подключении к нему для получения списка активных проверок.

По умолчанию: `0.0.0.0`

ListenPort

Агент будет слушать этот порт на предмет подключений с сервера.

По умолчанию: 10050<br> Диапазон: 1024-32767

LogFile

Имя файла журнала агента.

По умолчанию: `c:\zabbix_agent2.log`<br> Обязательное: Да, если для LogType задан как *file*; иначе — нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключение автоматической ротации журнала.<br>*Примечание*: Если лимит размера файла достигнут, а ротация по каким-либо причинам не удалась, существующий файл журнала усекается и начинается заново.

По умолчанию: 1<br> Диапазон: 0-1024<br>

LogType

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: `file`

PersistentBufferFile

Файл, в котором Zabbix агент 2 должен хранить базу данных SQLite. Имя файла должно быть указано с абсолютным путём. Этот параметр используется, только если включён постоянный буфер (*EnablePersistentBuffer=1*).

PersistentBufferPeriod

Период времени, в течение которого должны храниться данные, когда нет соединения с сервером или прокси. Более старые данные будут потеряны. Журнальные данные будут сохранены. Этот параметр используется, только если включён постоянный буфер (*EnablePersistentBuffer=1*).

По умолчанию: `1h`<br> Диапазон: 1m-365d

Plugins.<ИмяПлагина>.System.Capacity

Ограничение количества проверок на плагин < >, которые могут выполняться одновременно.

По умолчанию: 1000 Диапазон: 1-1000

Plugins.Log.MaxLinesPerSecond

Максимальное количество новых строк в секунду, которые агент будет отправлять Zabbix серверу или прокси при обработке активных проверок «log», «logrt» и «eventlog». Указаное значение будет переопределено параметром «max-lines», указанным в ключах элементов данных «log», «logrt» и «eventlog».<br>*Примечание*: при поиске нужной строки в элементах данных журналов Zabbix будет обрабатывать в 10 раз больше новых строк, чем указано в параметре *MaxLinesPerSecond*.

По умолчанию: 20<br> Диапазон: 1-1000

Plugins.SystemRun.LogRemoteCommands

Разрешение журналировать выполняемые команды оболочки как предупреждения. Команды будут записаны в журнал, только если они выполняются удалённо. Записи в журнале не создаются, если system.run[] запускается локально параметрами HostMetadataItem, HostInterfaceItem или HostnameItem.

По умолчанию: 0<br> Значения: 0 — запрещено, 1 — разрешено

PluginSocket

Путь к именованному каналу Windows для коммуникаций с подгружаемыми плагинами.

По умолчанию: `\\.\pipe\agent.plugin.sock`

PluginTimeout

Тайм-аут для соединений с подгружаемыми плагинами, в секундах.

По умолчанию: `Timeout`<br> Диапазон: 1-30

PerfCounter

Определяет новый параметр <parameter_name>, который является средним значением счетчика производительности системы <perf_counter_path> за указанный период времени <period> (в секундах).<br>Синтаксис: <parameter_name>,”<perf_counte

Например, если вы хотите получать среднее число прерываний процессора в секунду за последнюю минуту, можно определить новый параметр ”interrupts” следующим образом:<br>

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Обратите внимание на двойные кавычки вокруг пути к счетчику производительности. Имя параметра (interrupts) следует использовать как ключ элемента данных при создании элемента данных. Образцы для вычисления среднего значения будут сниматься каждую секунду.<br>Вы можете выполнить ”typeperf -qx”, чтобы получить список всех счетчиков производительности, доступных в Windows.

PerfCounterEn

Определяет новый параметр <parameter_name>, который является средним значением счетчика производительности системы <perf_counter_path> за указанный период времени <period> (в секундах). По сравнению с PerfCounter, пути perfcounter должны быть на английском языке. Поддерживается только в **Windows Server 2008/Vista** и более поздних версиях.<br>Синтаксис: <parameter_name>,”<perf_counter_path>”,<period>

Например, если вы хотите получать среднее количество прерываний процессора в секунду за последнюю минуту, вы можете определить новый параметр ”interrupts” следующим образом:<br>

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Обратите внимание на двойные кавычки вокруг пути к счетчику производительности. Имя параметра (interrupts) будет использоваться как ключ элемента данных при создании элемента данных. Образцы для вычисления среднего значения будут сниматься каждую секунду.<br>Список английских строк можно найти, просмотрев следующий раздел реестра: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

RefreshActiveChecks

Как часто обновлять список активных проверок, в секундах. Обратите внимание, что после неуспешного обновления активных проверок, следующая попытка будет предпринята через 60 секунд.

По умолчанию: 5<br> Диапазон: 1-86400

Server

Список (через запятую) IP адресов, опционально в CIDR нотации, или DNS имён хостов Zabbix серверов и Zabbix прокси. Входящие соединения будут приниматься только с хостов, указанных в этом списке. Если включена поддержка IPv6, то «127.0.0.1», «::127.0.0.1», «::ffff:127.0.0.1» обрабатываются одинаково, а «::/0» разрешает все IPv4 и IPv6 адреса. «0.0.0.0/0» можно использовать, чтобы разрешить любой IPv4 адрес. Пробелы разрешены. Если этот параметр не указан, пассивные проверки отключаются, и агент не будет прослушивать ни один порт TCP.

Пример:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Адрес Zabbix сервера/прокси или конфигурация кластера для получения активных проверок. Адрес сервера/прокси — это IP адрес или DNS имя и опционально порт, разделённые двоеточием.<br>Конфигурация кластера — один или более адресов серверов либо членов группы прокси, разделённые символом «точка с запятой». Можно указать несколько серверов/кластеров/прокси Zabbix, разделяя их запятой. От каждого сервера/кластера Zabbix должно быть указано не более одного Zabbix прокси, если только не используются группы прокси. Если указан Zabbix прокси, то сервер/кластер Zabbix для этого прокси не должны быть указаны.<br>Несколько адресов через запятую можно указать для использования нескольких независимых серверов Zabbix параллельно. Пробелы разрешены.<br>Если порт не указан, используется порт по умолчанию.<br>Адреса IPv6 должны быть заключены в квадратные скобки, если для этого хоста указан порт. Если порт не указан, квадратные скобки для адресов IPv6 опциональны.<br>Если этот параметр не указан, активные проверки отключены.

Пример для Zabbix прокси:

```
ServerActive=127.0.0.1:10051
```

Пример для группы Zabbix прокси:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Пример для нескольких серверов:

```
ServerActive=127.0.0.1:20051,zabbix.domain,\[::1\]:30051,::1,\[12fc::1\]
```

Пример для кластера высокой доступности:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Пример для конфигурации высокой доступности с двумя кластерами и одним сервером:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Локальный IP адрес для:

- исходящих подключений к Zabbix серверу или прокси;
- создания подключений при выполнении некоторых элементов данных (web.page.get, net.tcp.port и т.п.).

StatusPort

Если задан, агент будет слушать указанный порт для HTTP запросов состояния (http://localhost:<порт>/status).

Диапазон: 1024-32767

Timeout

Указывает, как долго (в секундах) следует ожидать установления соединения и обмена данными с Zabbix сервером или прокси.<br>

Этот параметр определяет продолжительность различных коммуникационных операций, в том числе:

- ожидание ответа от Zabbix сервера;
- отправку запросов на Zabbix сервер, в том числе запросы конфигурации элементов данных и значения элементов данных в активных проверках;
- получение данных журнала посредством мониторинга файла журнала или журнала событий Windows;
- отправка сообщений heartbeat;
- максимальная длительность для проверок `vfs.*`;
- используется в качестве запасного варианта в сценариях, где сервер или прокси версии старше 7.0 отправляет проверки без тайм-аутов.

Этот тайм-аут **не** будет использоваться для тех проверок агента, которые имеют настраиваемые параметры тайм-аута в веб-интерфейсе (на глобальном уровне, на уровне прокси-сервера или на уровне отдельных элементов данных).

По умолчанию: 3<br> Диапазон: 1-30

TLSAccept

Какие входящие соединения принимать. Используется для пассивных проверок. Можно указать несколько значений через запятую:<br>*unencrypted* — принимать соединения без шифрования (по умолчанию)<br>*psk* — принимать подключения с TLS и общим ключом (pre-shared key, PSK)<br>*cert* — принимать подключения с TLS и сертификатом

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже для *незашифрованных* подключений), в противном случае — нет

TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Абсолютный путь к файлу, который содержит сертификат агента или цепочку сертификатов, используется для зашифрованных соединений между компонентами Zabbix.

TLSCipherAll13

Строка шифров OpenSSL (TLS 1.3). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов и PSK.

Пример:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert13

Строка шифров OpenSSL (TLS 1.3). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе сертификатов.

Обратите внимание, что этот параметр нельзя использовать вместе с `TLSAccept=cert,psk`; для подключения по сертификату (`TLSConnect=cert`) вместо него используйте TLSCipherAll13.

TLSCipherPSK13

Строка шифров OpenSSL (TLS 1.3). Переопределяет критерии выбора набора шифров по умолчанию для шифрования на основе PSK.

Пример:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Как агент должен подключаться к Zabbix серверу или прокси. Используется для активных проверок. Можно указать только одно значение:<br>*unencrypted* — подключаться без шифрования (по умолчанию)<br>*psk* — подключаться, используя TLS и общий ключ (pre-shared key, PSK)<br>*cert* — подключаться, используя TLS и сертификат

Обязательное: да, если заданы TLS сертификат или параметры PSK (даже при *незашифрованном* соединении), в противном случае — нет

TLSCRLFile

Абсолютный путь к файлу, содержащему отозванные сертификаты. Этот параметр используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ агента (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

TLSPSKFile

Абсолютный путь к файлу, содержащему общий ключ (pre-shared key) агента. Используется для зашифрованных коммуникаций с сервером Zabbix.

TLSPSKIdentity

Строка идентификатора PSK, используется для зашифрованных соединений с Zabbix сервером.

TLSServerCertIssuer

Разрешённый эмитент сертификата сервера (прокси).

TLSServerCertSubject

Разрешённый субъект сертификата сервера (прокси).

UnsafeUserParameters

Разрешить передавать аргументами в пользовательские параметры все символы. Не разрешены следующие символы: \ ' " ` * ? [ ] { } ~ $ ! & ; ( ) < > | # @<br>Кроме того, не разрешены символы новой строки.

По умолчанию: 0<br> Значения: 0 — не разрешать, 1 — разрешить

UserParameter

Пользовательский параметр для мониторинга. Можно указать несколько пользовательских параметров.<br>Формат: UserParameter=<ключ>,<команда оболочки><br>Обратите внимание, что команда не должна возвращать только пустую строку или EOL. Команды оболочки могут иметь относительные пути, если указан параметр UserParameterDir.

Пример:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Путь поиска по умолчанию для команд UserParameter. Если используется, агент перед выполнением команды сменит свою рабочую директорию на указанную здесь. Таким образом, команды UserParameter могут иметь относительный префикс `.\` вместо полного пути.<br>Допустимо только одно значение.

Пример:

```
UserParameterDir=C:\opt\myscripts
```

### 7. Плагины Zabbix агента 2

Обзор

В этом разделе содержатся описания параметров файла конфигурации для плагинов Zabbix агента 2. Пожалуйста, используйте боковую панель для доступа к информации о конкретном плагине.

### 1. Плагин Ceph

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Ceph для агента 2 Zabbix (ceph.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с "#", поддерживаются только в начале строки.

Параметры

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.Ceph.Default.ApiKey | | | | Ключ API по умолчанию для подключения к Ceph; используется, если значение не указано в ключе элемента данных или в именованной сессии. |
| Plugins.Ceph.Default.User | | | | Имя пользователя по умолчанию для подключения к Ceph; используется, если значение не указано в ключе элемента данных или в именованной сессии. |
| Plugins.Ceph.Default.Uri | | | https://localhost:8003 | URI по умолчанию для подключения к Ceph; используется, если значение не указано в ключе элемента данных или в именованной сессии.<br><br>URL не должен содержать встроенные креденшиалы (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживается только схема `https`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию 8003).<br>Примеры: `https://127.0.0.1:8003`<br>`localhost` |
| Plugins.Ceph.InsecureSkipVerify | | false / true | false | Определяет, должен ли http клиент верифицировать серверные цепочку сертификатов и имя хоста.<br>Если *true*, TLS принимает любой предъявленный сервером сертификат и любое имя хоста в этом сертификате. В этом режиме TLS уязвим к атакам «человек посередине» (man-in-the-middle) (следует использовать только для тестирования). |
| Plugins.Ceph.KeepAlive | | 60-900 | 300 | Максимальное время ожидания (в секундах), прежде чем неиспользуемые плагином соединения закрываются. |
| Plugins.Ceph.Sessions.<ИмяСессии>.ApiKey | | | | Ключ API именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.Ceph.Sessions.<ИмяСессии>.User | | | | Имя пользователя именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.Ceph.Sessions.<ИмяСессии>.Uri | | | | Строка подключения именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных.<br><br>Не должна содержать встроенные креденшиалы (они будут проигнорированы).<br>Должна соответствовать формату URI.<br>Поддерживается только схема `https`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию 8003).<br>Примеры: `https://127.0.0.1:8003`<br>`localhost` |
| Plugins.Ceph.Timeout | нет | 1-30 | глобальный тайм-аут | Тайм-аут выполнения запроса (продолжительность в секундах ожидания завершения запроса, прежде чем он будет завершён принудительно). |

Смотрите также:

- Описание общих параметров конфигурации Zabbix агента 2: Zabbix агент 2 (UNIX) / Zabbix агент 2 (Windows)
- Инструкции по настройке плагинов

**2. Плагин Docker**

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Docker для агента 2 Zabbix (docker.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.Docker.Endpoint | | | unix:///var/run/docker.sock | Местоположение unix-сокета демона Docker.<br>Должно содержать схему (поддерживается только `unix://`). |
| Plugins.Docker.Timeout | нет | 1-30 | глобальный тайм-аут | Тайм-аут выполнения запроса (продолжительность в секундах ожидания завершения запроса, прежде чем он будет завершён принудительно). |

Смотрите также:

- Описание общих параметров конфигурации Zabbix агента 2: Zabbix агент 2 (UNIX) / Zabbix агент 2 (Windows)
- Инструкции по настройке плагинов

**3. Плагин Ember+**

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Ember+ для агента 2 Zabbix (ember.conf).

Плагин Ember+ является загружаемым плагином и доступен, а также полностью описан в репозитории плагина Ember+.

В настоящее время этот плагин доступен только для сборки из исходного кода (как для Unix, так и для Windows).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Опции

| Параметр | Описание |
|---|---|
| -V --version | Вывести версию плагина и информацию о лицензии. |
| -h --help | Распечатать справочную информацию (сокращённо). |

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.EmberPlus.Default.Uri | | | tcp://localhost:9998 | URI по умолчанию для подключения. Единственная поддерживаемая схема — `tcp://`. Схему можно опустить. Встроенные учетные данные будут проигнорированы. |
| Plugins.EmberPlus.KeepAlive | | 60-900 | 300 | Максимальное время ожидания (в секундах) до закрытия неиспользуемых подключений плагина. |
| Plugins.EmberPlus.Sessions.<SessionName>.Url | | | tcp://localhost:9998 | URI для подключения для указанной сессии. Единственная поддерживаемая схема — `tcp://`. Схему можно опустить. Встроенные учетные данные будут проигнорированы.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.EmberPlus.System.Path | | | | Путь к исполняемому файлу плагина Ember+.<br>Пример использования:<br>`Plugins.EmberPlus.System.Path=/usr/libexec/zabbix/zabbix` |
| Plugins.EmberPlus.Timeout | | 1-30 | global timeout | Продолжительность ожидания ответа от сервера в секундах при первом подключении и при последующих операциях в сессии. |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

#### 4. Плагин Memcached

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Memcached для агента 2 Zabbix (memcached.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.Memcached.Default.Password | нет | | | Пароль по умолчанию для подключения к Memcached; используется, если значение не указано ни в ключе элемента данных, ни в именованной сессии. |
| Plugins.Memcached.Default.Uri | нет | | tcp://localhost:11211 | URI по умолчанию для подключения к Memcached; используется, если значение не указано ни в ключе элемента данных, ни в именованной сессии.<br><br>Не должно содержать встроенных креденшиалов (они будут проигнорированы).<br>Должно соответствовать формату URI.<br>Поддерживаются схемы: `tcp`, `unix`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию 11211).<br>Примеры: `tcp://localhost:11211`<br>`localhost`<br>`unix:/var/run/memcached.sock` |
| Plugins.Memcached.Default.User | нет | | | Имя пользователя по умолчанию для подключения к Memcached; используется, если значение не указано ни в ключе элемента данных, ни в именованной сессии. |
| Plugins.Memcached.KeepAlive | нет | 60-900 | 300 | Максимальное время ожидания (в секундах), прежде чем неиспользуемые соединения плагина будут закрыты. |
| Plugins.Memcached.Sessions.<ИмяСессии>.Password | нет | | | Пароль именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.Memcached.Sessions.<ИмяСессии>.Uri | нет | | | Строка подключения именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных.<br><br>Не должна содержать встроенных креденшиалов (они будут проигнорированы).<br>Должна соответствовать формату URI.<br>Поддерживаются схемы: `tcp`, `unix`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию 11211).<br>Примеры: `tcp://localhost:11211`<br>`localhost`<br>`unix:/var/run/memcached.sock` |
| Plugins.Memcached.Sessions.<ИмяСессии>.User | нет | | | Имя пользователя именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.Memcached.Timeout | нет | 1-30 | глобальный тайм-аут | Тайм-аут выполнения запроса (продолжительность в секундах ожидания завершения запроса, прежде чем он будет завершён принудительно). |

Смотрите также:

- Описание общих параметров конфигурации Zabbix агента 2: Zabbix агент 2 (UNIX) / Zabbix агент 2 (Windows)
- Инструкции по настройке плагинов

#### 5. Плагин Modbus

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Modbus для агента 2 Zabbix (modbus.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;

- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.Modbus.Sessions.<ИмяСессии>.Endpoint | | | | Точка входа — строка подключения, состоящая из схемы протокола, адреса хоста и порта или имени последовательного порта и атрибутов.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.Modbus.Sessions.<ИмяСессии>.SlaveID | | | | Идентификатор ведомого именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных.<br>Пример: `Plugins.Modbus.Sessions.MB1.SlaveID=20`<br>*Обратите внимание*, что этот параметр именованной сессии проверяется только если значение id ведомого в параметре ключа элемента данных пустое. |
| Plugins.Modbus.Sessions.<ИмяСессии>.Timeout | | | | Тайм-аут именованной сессии в секундах.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных.<br>Пример: `Plugins.Modbus.Sessions.MB1.Timeout=2` |

Если вам нужно задать тайм-аут выполнения запроса (как долго ждать завершения запроса, прежде чем он будет завершён принудительно), используйте диалог настроек элемента данных.

Смотрите также:

- Описание общих параметров конфигурации Zabbix агента 2: Zabbix агент 2 (UNIX) / Zabbix агент 2 (Windows)
- Инструкции по настройке плагинов

#### 6. Плагин MongoDB

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина MongoDB для агента 2 Zabbix (mongo.conf).

Плагин MongoDB является загружаемым плагином и доступен, а также полностью описан в репозитории плагина MongoDB.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Опции

| Параметр | Описание |
|---|---|
| -V --version | Вывести версию плагина и информацию о лицензии. |
| -h --help | Вывести справочную информацию (сокращённо). |

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.MongoDB.Default.Password | | | | Пароль по умолчанию для подключения к MongoDB; используется, если в ключе элемента данных или именованной сессии не указано значение. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.MongoDB.Default.Uri | | | | URI по умолчанию для подключения к MongoDB; используется, если в ключе элемента данных или именованной сессии не указано значение.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживается только схема `tcp`; схему можно опустить.<br>Порт можно опустить (по умолчанию=27017).<br>Примеры: `tcp://127.0.0.1:27017`, `tcp:localhost`, `localhost` |
| Plugins.MongoDB.Default.User | | | | Имя пользователя по умолчанию для подключения к MongoDB; используется, если в ключе элемента данных или именованной сессии не указано значение. |
| Plugins.MongoDB.KeepAlive | | 60-900 | 300 | Максимальное время ожидания (в секундах) до закрытия неиспользуемых подключений плагина. |
| Plugins.MongoDB.Sessions.<SessionName>.Password | | | | Пароль именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile | (yes, if Plugins.MongoDB.Sessions.<SessionName>.TLSConnect is set to one of: verify_ca, verify_full) | | | Полный путь к файлу, содержащему сертификаты корневого CA(ов) для проверки сертификата узла, используемые для зашифрованных соединений между агентом Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile | (yes, if Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile is specified | | | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов, используемые для зашифрованных соединений между агентом Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.MongoDB.Sessions.<SessionName>.TLSConnect | | | | Тип шифрования для соединений между агентом Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Поддерживаемые значения:<br>*required* - требуется TLS-соединение;<br>*verify_ca* - проверка сертификатов;<br>*verify_full* - проверка сертификатов и IP-адреса.<br><br>Поддерживается начиная с версии плагина 1.2.1 |
| Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile | (yes, if Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile is specified | | | Полный путь к файлу, содержащему закрытый ключ базы данных, используемый для зашифрованных соединений между агентом Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.MongoDB.Sessions.<SessionName>.Uri | | | | Строка подключения именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживается только схема `tcp`; схему можно опустить.<br>Порт можно опустить (по умолчанию=27017).<br>Примеры: `tcp://127.0.0.1:27017`, `tcp:localhost`, `localhost` |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.MongoDB.Sessions.<SessionName>.User | | | | Имя пользователя именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.MongoDB.System.Path | | | | Путь к исполняемому файлу плагина MongoDB.<br>Пример использования:<br>`Plugins.MongoDB.System.Path=/usr/libexec/zabbix/zabbix-ag` |
| Plugins.MongoDB.Timeout | | 1-30 | global timeout | Тайм-аут выполнения запроса (время ожидания, в секундах, до завершения запроса перед его остановкой). |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

**7. Плагин MQTT**

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина MQTT для агента 2 Zabbix (mqtt.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.MQTT.Default.Password | | | | Пароль по умолчанию для подключения к MQTT; используется, если значение не указано в ключе элемента данных или в именованной сессии. |
| Plugins.MQTT.Default.TLSCAFile | | | | Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла для зашифрованных соединений между Zabbix агентом 2 и брокером MQTT; используется, если значение не указано в именованной сессии. |
| Plugins.MQTT.Default.TLSCertFile | | | | Абсолютный путь к файлу, который содержит сертификат агента или цепочку сертификатов для зашифрованных соединений между Zabbix агентом 2 и брокером MQTT; используется, если значение не указано в именованной сессии. |
| Plugins.MQTT.Default.TLSKeyFile | | | | Абсолютный путь к файлу, содержащему закрытый ключ (private key) MQTT для зашифрованных коммуникаций между Zabbix агентом 2 и брокером MQTT; используется, если значение не указано в именованной сессии. |
| Plugins.MQTT.Default.Topic | | | | Тема по умолчанию для подписки MQTT; используется, если значение не указано в ключе элемента данных или в именованной сессии.<br><br>Тема может содержать подстановочные символы (”+”,”#”)<br>Примеры: `path/to/file`<br>`path/to/#`<br>`path/+/topic` |

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.MQTT.Default.Url | | | tcp://localhost:1883 | Строка подключения к брокеру MQTT по умолчанию; используется, если значение не указано в ключе элемента данных или в именованной сессии.<br><br>Не должна включать параметры запроса.<br>Должна соответствовать формату URL.<br>Поддерживаются схемы: `tcp` (по умолчанию), `ws`, `tls`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию 1883).<br>Примеры: `tcp://host:1883`<br>`localhost`<br>`ws://host:8080` |
| Plugins.MQTT.Default.User | | | | Имя пользователя по умолчанию для подключения к MQTT; используется, если значение не указано в ключе элемента данных или в именованной сессии. |
| Plugins.MQTT.Sessions.<ИмяСессии>.Password | | | | Пароль именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.MQTT.Sessions.<ИмяСессии>.TLSCAFile | | | | Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между Zabbix агентом 2 и брокером MQTT.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.MQTT.Sessions.<ИмяСессии>.TLSCertFile | | | | Абсолютный путь к файлу, который содержит сертификат агента или цепочку сертификатов, которые используются для зашифрованных соединений между Zabbix агентом 2 и брокером MQTT.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.MQTT.Sessions.<ИмяСессии>.TLSKeyFile | | | | Абсолютный путь к файлу, содержащему закрытый ключ (private key) MQTT, который используется для зашифрованных коммуникаций между Zabbix агентом 2 и брокером MQTT.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |
| Plugins.MQTT.Sessions.<ИмяСессии>.Topic | | | | Тема именованной сессии для подписки MQTT.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных.<br><br>Тема может содержать подстановочные символы (”+”,”#”)<br>Примеры: `path/to/file`<br>`path/to/#`<br>`path/+/topic` |
| Plugins.MQTT.Sessions.<ИмяСессии>.Url | | | | Строка подключения именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных.<br><br>Не должна включать параметры запроса.<br>Должна соответствовать формату URL.<br>Поддерживаются темы: `tcp` (по умолчанию), `ws`, `tls`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию 1883).<br>Примеры: `tcp://host:1883`<br>`localhost`<br>`ws://host:8080` |

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.MQTT.Sessions.<ИмяСессии>.User | | | | Имя пользователя именованной сессии.<br>**<ИмяСессии>** — определяет имя сессии, используемое в ключах элементов данных. |

Если вам нужно задать тайм-аут выполнения запроса (продолжительность в секундах ожидания завершения запроса, прежде чем он будет завершён принудительно),используйте диалог настройки элемента данных.

Смотрите также:

- Описание общих параметров конфигурации Zabbix агента 2: Zabbix агент 2 (UNIX) / Zabbix агент 2 (Windows)
- Инструкции по настройке плагинов

#### 8. Плагин MSSQL

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина MSSQL для агента 2 Zabbix (mssql.conf).

Плагин MSSQL является загружаемым плагином и доступен, а также полностью описан в репозитории плагина MSSQL.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с "#", поддерживаются только в начале строки.

Опции

| Параметр | Описание |
|---|---|
| -V --version | Вывести версию плагина и информацию о лицензии. |
| -h --help | Распечатать справочную информацию (сокращённо). |

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.MSSQL.CustomQueriesDir | | | `/usr/local/share/zabbix/mssql_custom_queries` для систем Unix<br>`*:\Program Files\Zabbix Agent 2\Custom Queries\MSSQL` для систем Windows, где * — имя диска из переменной среды `ProgramFiles` | Указывает путь к каталогу, содержащему пользовательские файлы .sql с пользовательскими запросами, которые плагин может выполнять. Плагин загружает все доступные файлы .sql в настроенном каталоге при запуске. Это означает, что любые изменения в файлах пользовательских запросов не будут отражены до перезапуска плагина. Плагин запускается и останавливается вместе с Zabbix agent 2. |
| Plugins.MSSQL.CustomQueriesEnabled | | | false | Если задано, включает выполнение ключа элемента данных `mssql.custom.query`. Если отключено, из пути каталога пользовательских запросов не будут загружаться никакие запросы. |
| Plugins.MSSQL.Default.CACertPath | | | | Путь к файлу с публичным ключом сертификата центра сертификации (CA), выдавшего сертификат сервера MSSQL. Сертификат должен быть в формате PEM. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.MSSQL.Default.Database | | | | Имя базы данных, к которой подключаться по умолчанию. |
| Plugins.MSSQL.Default.Encrypt | | | | Указывает тип шифрования соединения по умолчанию. Возможные значения:<br>*true* - передача данных между плагином и сервером шифруется;<br>*false* - передача данных между плагином и сервером не шифруется, кроме пакета входа в систему;<br>*strict* - передача данных между плагином и сервером шифруется end-to-end с использованием TDS8;<br>*disable* - передача данных между плагином и сервером не шифруется. |
| Plugins.MSSQL.Default.HostNameInCertificate | | | | Общее имя (CN) сертификата сервера MSSQL по умолчанию. |
| Plugins.MSSQL.Default.Password | | | | Пароль, который по умолчанию будет отправляться на защищенный сервер MSSQL. |
| Plugins.MSSQL.Default.TLSMinVersion | | | | Минимальная версия TLS, используемая по умолчанию. Возможные значения: `1.0`, `1.1`, `1.2`, `1.3`. |
| Plugins.MSSQL.Default.TrustServerCertificate | | | | Следует ли плагину доверять сертификату сервера без его проверки по умолчанию. Возможные значения: `true`, `false`. |
| Plugins.MSSQL.Default.Uri | | | sqlserver://localhost:1433 | URI подключения по умолчанию. Единственная поддерживаемая схема — `sqlserver://`. Схему можно опустить. Встроенные учетные данные будут проигнорированы. |
| Plugins.MSSQL.Default.User | | | | Имя пользователя, которое по умолчанию будет отправляться на защищенный сервер MSSQL. |
| Plugins.MSSQL.KeepAlive | | 60-900 | 300 | Максимальное время ожидания (в секундах) до закрытия неиспользуемых соединений плагина. |
| Plugins.MSSQL.Sessions.<SessionName>.CACertPath | | | | Путь к файлу с публичным ключом сертификата центра сертификации (CA), выдавшего сертификат сервера MSSQL, для именованной сессии. Сертификат должен быть в формате PEM.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.Database | | | | Имя базы данных, к которой нужно подключаться для именованной сессии.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.Encrypt | | | | Указывает тип шифрования соединения для именованной сессии. Возможные значения:<br>*true* - передача данных между плагином и сервером шифруется;<br>*false* - передача данных между плагином и сервером не шифруется, кроме пакета входа в систему;<br>*strict* - передача данных между плагином и сервером шифруется end-to-end с использованием TDS8;<br>*disable* - передача данных между плагином и сервером не шифруется.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate | | | | Общее имя (CN) сертификата сервера MSSQL для именованной сессии.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.Password | | | | Пароль, который будет отправляться на защищенный сервер MSSQL для именованной сессии.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion | | | | Минимальная версия TLS, используемая для именованной сессии. Возможные значения: `1.0`, `1.1`, `1.2`, `1.3`.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate | | | | Следует ли плагину доверять сертификату сервера без его проверки для именованной сессии. Возможные значения: `true`, `false`.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.Uri | | | sqlserver://localhost:1433 | URI подключения для именованной сессии. Единственная поддерживаемая схема — `sqlserver://`. Схему можно опустить. Встроенные учетные данные будут проигнорированы.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.Sessions.<SessionName>.User | | | | Имя пользователя, которое будет отправляться на защищенный сервер MSSQL для именованной сессии.<br>**<SessionName>** - задайте имя сессии для использования в ключах элементов данных. |
| Plugins.MSSQL.System.Path | | | | Путь к исполняемому файлу плагина MSSQL. Глобальная настройка для плагина MSSQL. Применяется ко всем соединениям.<br>Пример использования: `Plugins.MSSQL.System.Path=/usr/libexec/zabbix/zabbix-agen` |
| Plugins.MSSQL.Timeout | | 1-30 | global timeout | Продолжительность ожидания ответа сервера в секундах при первом подключении и при последующих операциях в сессии. |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

#### 9. Плагин MySQL

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина MySQL для агента 2 Zabbix (mysql.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Mysql.CallTimeout | | 1-30 | global timeout | Максимальное время ожидания в секундах до завершения запроса. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Mysql.CustomQueriesPath | | | /usr/local/share/zabbix/custom-queries/mysql для систем Unix<br><br>*:\Program Files\Zabbix Agent 2\Custom Queries\Mysql для систем Windows, где * — имя диска из переменной среды ProgramFiles | Полный путь к каталогу, содержащему файлы .sql с пользовательскими запросами. |
| Plugins.Mysql.CustomQueriesEnabled | | | false | Если задано, включает выполнение ключа элемента данных mysql.custom.query. Если отключено, запросы не будут загружаться из пути к каталогу пользовательских запросов. |
| Plugins.Mysql.Default.Password | | | | Пароль по умолчанию для подключения к MySQL; используется, если в ключе элемента данных или именованной сессии не указано значение. |
| Plugins.Mysql.Default.TLSCAFile | (yes, if Plugins.Mysql.Default.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему сертификаты корневого(ых) CA для проверки сертификата узла при зашифрованных соединениях между Zabbix agent 2 и контролируемыми базами данных; используется, если в именованной сессии не указано значение. |
| Plugins.Mysql.Default.TLSCertFile | (yes, if Plugins.Mysql.Default.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов для зашифрованных соединений между Zabbix agent 2 и контролируемыми базами данных; используется, если в именованной сессии не указано значение. |
| Plugins.Mysql.Default.TLSConnect | | | | Тип шифрования для соединений между Zabbix agent 2 и контролируемыми базами данных; используется, если в именованной сессии не указано значение.<br><br>Поддерживаемые значения:<br>*required* - требуется TLS-соединение;<br>*verify_ca* - проверка сертификатов;<br>*verify_full* - проверка сертификатов и IP-адреса. |
| Plugins.Mysql.Default.TLSKeyFile | (yes, if Plugins.Mysql.Default.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему закрытый ключ базы данных для зашифрованных соединений между Zabbix agent 2 и контролируемыми базами данных; используется, если в именованной сессии не указано значение. |
| Plugins.Mysql.Default.Uri | | | tcp://localhost:3306 | URI по умолчанию для подключения к MySQL; используется, если в ключе элемента данных или именованной сессии не указано значение.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживаемые схемы: tcp, unix; схему можно опустить.<br>Порт можно опустить (по умолчанию=3306).<br>Примеры: tcp://localhost:3306<br>localhost<br>unix:/var/run/mysql.sock |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Mysql.Default.User | no | | | Имя пользователя по умолчанию для подключения к MySQL; используется, если в ключе элемента данных или именованной сессии не указано значение. |
| Plugins.Mysql.KeepAlive | no | 60-900 | 300 | Максимальное время ожидания (в секундах) до закрытия неиспользуемых подключений плагина. |
| Plugins.Mysql.Sessions.<SessionName>.Password | no | | | Пароль именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Mysql.Sessions.<SessionName>.TLSCAFile | no (yes, if Plugins.Mysql.Sessions.<SessionName>.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему сертификаты корневого(ых) CA для проверки сертификата узла, используемый для зашифрованных соединений между Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Mysql.Sessions.<SessionName>.TLSCertFile | yes, if Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile is specified | | | Полный путь к файлу, содержащему сертификат агента или цепочку сертификатов, используемый для зашифрованных соединений между Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Mysql.Sessions.<SessionName>.TLSConnect | no | | | Тип шифрования для соединений между Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Поддерживаемые значения:<br>*required* - требуется TLS-соединение;<br>*verify_ca* - проверка сертификатов;<br>*verify_full* - проверка сертификатов и IP-адреса. |
| Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile | yes, if Plugins.Mysql.Sessions.<SessionName>.TLSCertFile is specified | | | Полный путь к файлу, содержащему закрытый ключ базы данных, используемый для зашифрованных соединений между Zabbix agent 2 и контролируемыми базами данных.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Mysql.Sessions.<SessionName>.Uri | no | | | Строка подключения именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживаемые схемы: `tcp`, `unix`; схему можно опустить.<br>Порт можно опустить (по умолчанию=3306).<br>Примеры: `tcp://localhost:3306`<br>`localhost`<br>`unix:/var/run/mysql.sock` |
| Plugins.Mysql.Sessions.<SessionName>.User | no | | | Имя пользователя именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Mysql.Timeout | no | 1-30 | global timeout | Максимальное время ожидания в секундах при установлении соединения. |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

**10 Плагин NVIDIA GPU**

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина NVIDIA GPU для агента 2 Zabbix (nvidia.conf).

Плагин NVIDIA GPU является загружаемым плагином и доступен, а также полностью описан в репозитории плагина NVIDIA GPU.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.NVIDIA.System.Path | | | | Путь к исполняемому файлу плагина NVIDIA GPU.<br>Пример использования:<br>`Plugins.NVIDIA.System.Path=/path/to/executable/nvidia` |
| Plugins.NVIDIA.Timeout | | 1-30 | global timeout | Указывает максимальное время (в секундах) ожидания ответа сервера при попытках подключения и последующих операциях в сеансе. Глобальный тайм-аут для элемента данных или индивидуальный тайм-аут элемента данных переопределит это значение, если оно больше. |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

**11 Плагин Oracle**

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Oracle Zabbix agent 2 (oracle.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Oracle.CallTimeout | | 1-30 | global timeout | Максимальное время ожидания в секундах до завершения запроса. |
| Plugins.Oracle.ConnectTimeout | | 1-30 | global timeout | Максимальное время ожидания в секундах до установления соединения. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Oracle.CustomQueriesPath | | | /usr/local/share/zabbix/custom-queries/oracle для систем Unix<br><br>*:\Program Files\Zabbix Agent 2\Custom Queries\Oracle для систем Windows, где * - имя диска из переменной среды ProgramFiles | Полный путь к каталогу, содержащему файлы .sql с пользовательскими запросами.<br>Пример: /etc/zabbix/oracle/sql |
| Plugins.Oracle.CustomQueriesEnabled | | | false | Если задано, включает выполнение ключа элемента данных oracle.custom.query. Если отключено, запросы не будут загружаться из пути каталога пользовательских запросов. |
| Plugins.Oracle.Default.Password | | | | Пароль по умолчанию для подключения к Oracle; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.Oracle.Default.Service | | | | Имя службы по умолчанию для подключения к Oracle (SID не поддерживается); используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.Oracle.Default.Uri | | | tcp://localhost:1521 | URI по умолчанию для подключения к Oracle; используется, если значение не указано в ключе элемента данных или именованной сессии.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживается только схема tcp; схему можно опустить.<br>Порт можно опустить (по умолчанию=1521).<br>Начиная с **Zabbix 7.4.3** также можно указывать ключ TNS или значение TNS в качестве строки подключения.<br>Значение TNS должно быть составлено без пробелов.<br>Примеры: tcp://127.0.0.1:1521<br>localhost<br>zbx_tns_example (ключ TNS)<br>(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(POR<br>(значение TNS) |
| Plugins.Oracle.Default.User | | | | Имя пользователя по умолчанию для подключения к Oracle; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.Oracle.KeepAlive | | 60-900 | 300 | Максимальное время ожидания (в секундах) до закрытия неиспользуемых подключений плагина. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Oracle.ResolveTNS | no | true / false | false | Параметр определяет, как интерпретировать строку подключения (ConnString) в ключе метрики `commonParams` для подключения к серверу Oracle.<br>Если установлено значение false, к ConnString будут добавлены схема по умолчанию и порт (если они отсутствуют). Если параметр установлен в true, схема по умолчанию и порт будут опущены (если только они явно не указаны в ConnString), а ConnString будет передан клиенту Oracle как есть. Если клиент Oracle найдет ConnString в файле tnsnames.ora, для подключения к серверу Oracle будет использовано найденное описание подключения.<br>Этот параметр поддерживается начиная с Zabbix 7.4.3. |
| Plugins.Oracle.Sessions.<SessionName>.Password | no | | | Пароль именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Oracle.Sessions.<SessionName>.Service | no | | | Имя службы именованной сессии, используемое для подключения (SID не поддерживается).<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.Oracle.Sessions.<SessionName>.Uri | no | | | Строка подключения именованной сессии для Oracle.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Не должна содержать встроенные учетные данные (они будут проигнорированы).<br>Должна соответствовать формату URI.<br>Поддерживается только схема `tcp`; схему можно опустить.<br>Порт можно опустить (по умолчанию=1521).<br>Начиная с **Zabbix 7.4.3** также можно указывать ключ TNS или значение TNS в качестве строки подключения.<br>Значение TNS должно быть составлено без пробелов.<br>Примеры: `tcp://127.0.0.1:1521`<br>`localhost`<br>`zbx_tns_example` (ключ TNS)<br>`(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(POR` (значение TNS) |
| Plugins.Oracle.Sessions.<SessionName>.User | no | | | Имя пользователя именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

### 12 Плагин PostgreSQL

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина PostgreSQL для агента 2 Zabbix (postgresql.conf).

Плагин PostgreSQL является загружаемым плагином и доступен, а также полностью описан в репозитории плагина PostgreSQL.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды;

- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Опции

| Параметр | Описание |
|---|---|
| -V --version | Вывести версию плагина и информацию о лицензии. |
| -h --help | Распечатать справочную информацию (сокращённо). |

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.PostgreSQL.Default.CacheMode | | | prepare | Режим кэширования для подключения PostgreSQL.<br>Поддерживаемые значения:<br>*prepare* (по умолчанию) - создаст подготовленные операторы на сервере PostgreSQL;<br>*describe* - будет использовать анонимный подготовленный оператор для описания оператора без создания оператора на сервере.<br>Обратите внимание, что ”describe” в первую очередь полезен, когда среда не позволяет использовать подготовленные операторы, например при работе через пул соединений, такой как PgBouncer. |
| Plugins.PostgreSQL.CallTimeout | | 1-30 | global timeout | Максимальное время ожидания (в секундах) завершения запроса. |
| Plugins.PostgreSQL.CustomQueriesPath | | | `/usr/local/share/zabbix/custom-queries/postgresql` для Unix-систем<br><br>`*:\Program Files\Zabbix Agent 2\Custom Queries\PostgreSQL` для Windows-систем, где `*` - имя диска из переменной среды `ProgramFiles` | Полный путь к каталогу, содержащему файлы `.sql` с пользовательскими запросами. |
| Plugins.PostgreSQL.CustomQueriesEnabled | | | false | Если задано, включает выполнение ключа элемента данных `postgresql.custom.query`. Если отключено, запросы не будут загружаться из пути каталога пользовательских запросов. |
| Plugins.PostgreSQL.Default.Database | | | | База данных по умолчанию для подключения к PostgreSQL; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.PostgreSQL.Default.Password | | | | Пароль по умолчанию для подключения к PostgreSQL; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.PostgreSQL.Default.TLSCAFile | (yes, if Plugins.PostgreSQL.Default.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему сертификат(ы) CA верхнего уровня для проверки сертификата узла при зашифрованной связи между Zabbix agent 2 и контролируемыми базами данных; используется, если значение не указано в именованной сессии. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.PostgreSQL.Default.TLSCertFile | No (yes, if Plugins.PostgreSQL.Default.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему сертификат PostgreSQL или цепочку сертификатов для зашифрованной связи между Zabbix agent 2 и контролируемыми базами данных; используется, если значение не указано в именованной сессии. |
| Plugins.PostgreSQL.Default.TLSConnect | No | | | Тип шифрования для связи между Zabbix agent 2 и контролируемыми базами данных; используется, если значение не указано в именованной сессии.<br>Поддерживаемые значения:<br>*required* - подключение с использованием TLS в качестве транспортного режима без проверки идентичности;<br>*verify_ca* - подключение с использованием TLS и проверкой сертификата;<br>*verify_full* - подключение с использованием TLS, проверкой сертификата и проверкой того, что идентификатор базы данных (CN), указанный DBHost, совпадает с ее сертификатом.<br>Неопределенный тип шифрования означает незашифрованное соединение. |
| Plugins.PostgreSQL.Default.TLSKeyFile | No (yes, if Plugins.PostgreSQL.Default.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему закрытый ключ PostgreSQL для зашифрованной связи между Zabbix agent 2 и контролируемыми базами данных; используется, если значение не указано в именованной сессии. |
| Plugins.PostgreSQL.Default.Uri | No | | | URI по умолчанию для подключения к PostgreSQL; используется, если значение не указано в ключе элемента данных или именованной сессии.<br><br>Не должен включать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживаемые схемы: `tcp`, `unix`.<br>Примеры: `tcp://127.0.0.1:5432`<br>`tcp://localhost`<br>`unix:/var/run/postgresql/.s.PGSQL.5432` |
| Plugins.PostgreSQL.Default.User | No | | | Имя пользователя по умолчанию для подключения к PostgreSQL; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.PostgreSQL.KeepAlive | No | 60-900 | 300 | Максимальное время ожидания (в секундах) перед закрытием неиспользуемых подключений плагина. |
| Plugins.PostgreSQL.Sessions.<SessionName>.CachePolicy | No | | prepare | Режим кэширования для подключения PostgreSQL.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br>Поддерживаемые значения:<br>*prepare* (по умолчанию) - создаст подготовленные операторы на сервере PostgreSQL;<br>*describe* - будет использовать анонимный подготовленный оператор для описания оператора без создания оператора на сервере.<br>Обратите внимание, что ”describe” в первую очередь полезен, когда среда не позволяет использовать подготовленные операторы, например при работе через пул соединений, такой как PgBouncer. |
| Plugins.PostgreSQL.Sessions.<SessionName>.Database | No | | | База данных для подключения сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.PostgreSQL.Sessions.<SessionName>.Password | No | Must match the password format. | | Пароль для подключения сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile | no (yes, if Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect is set to *verify_ca* or *verify_full*) | | | Полный путь к файлу, содержащему сертификат(ы) CA верхнего уровня для проверки сертификата узла.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile | yes, if Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile is specified | | | Полный путь к файлу, содержащему сертификат PostgreSQL или цепочку сертификатов.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect | no | | | Тип шифрования для подключения PostgreSQL.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Поддерживаемые значения:<br>*required* - подключение с использованием TLS в качестве транспортного режима без проверки идентичности;<br>*verify_ca* - подключение с использованием TLS и проверкой сертификата;<br>*verify_full* - подключение с использованием TLS, проверкой сертификата и проверкой того, что идентификатор базы данных (CN), указанный DBHost, совпадает с ее сертификатом.<br>Неопределенный тип шифрования означает незашифрованное соединение. |
| Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile | yes, if Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile is specified | | | Полный путь к файлу, содержащему закрытый ключ PostgreSQL.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.PostgreSQL.Sessions.<SessionName>.Uri | no | | | Строка подключения именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных.<br><br>Не должен включать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживаемые схемы: `tcp`, `unix`.<br>Примеры: `tcp://127.0.0.1:5432`<br>`tcp://localhost`<br>`unix:/var/run/postgresql/.s.PGSQL.5432` |
| Plugins.PostgreSQL.Sessions.<SessionName>.User | no | | | Имя пользователя именованной сессии.<br>**<SessionName>** - задает имя сессии для использования в ключах элементов данных. |
| Plugins.PostgreSQL.System.Path | yes | | | Путь к исполняемому файлу плагина PostgreSQL.<br>Пример использования:<br>`Plugins.PostgreSQL.System.Path=/usr/libexec/zabbix/zabbi` |
| Plugins.PostgreSQL.Timeout | no | 1-30 | global timeout | Таймаут выполнения запроса (время ожидания, в секундах, по истечении которого запрос будет остановлен). |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

### 13 Плагин Redis

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина Redis для Zabbix agent 2 (redis.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации.
- Значения поддерживают переменные среды.
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM.
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Redis.Default.Password | no | | | Пароль по умолчанию для подключения к Redis; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.Redis.Default.TLSConnect | no | | | Требование к проверке TLS для клиента (Zabbix agent 2).<br><br>Поддерживаемые значения:<br>`required` - шифрованное, без проверки (только для тестирования)<br>`verify_ca` - шифрованное, проверка сертификата сервера<br>`verify_full` - шифрованное, проверка сертификата сервера и имени сервера с помощью SAN<br>Пустое или не заданное значение считается отсутствием TLS. |
| Plugins.Redis.Default.TLSCAFile | no | | | Полный путь к файлу, содержащему сертификаты корневого CA. |
| Plugins.Redis.Default.TLSCertFile | no | | | Полный путь к файлу, содержащему сертификат агента. |
| Plugins.Redis.Default.TLSKeyFile | no | | | Полный путь к файлу, содержащему закрытый ключ агента. |
| Plugins.Redis.Default.Uri | no | | tcp://localhost:6379 | URI по умолчанию для подключения к Redis; используется, если значение не указано в ключе элемента данных или именованной сессии.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживаемые схемы: `tcp`, `unix`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию=6379).<br>Примеры: `tcp://localhost:6379`<br>`localhost`<br>`unix:/var/run/redis.sock` |
| Plugins.Redis.Default.User | no | | default | Пользователь по умолчанию для отправки на защищенный сервер Redis; используется, если значение не указано в ключе элемента данных или именованной сессии. |
| Plugins.Redis.KeepAlive | no | 60-900 | 300 | Максимальное время ожидания (в секундах) до закрытия неиспользуемых подключений плагина. |
| Plugins.Redis.Sessions.<SessionName>.Password | no | | | Пароль именованной сессии.<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных. |

| Parameter | Mandatory | Range | Default | Description |
|---|---|---|---|---|
| Plugins.Redis.Sessions.<SessionName>.TLSConnect | | | | Требование к проверке TLS для клиента (Zabbix agent 2).<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных.<br><br>Поддерживаемые значения:<br>`required` - шифрованное, без проверки (только для тестирования)<br>`verify_ca` - шифрованное, проверка сертификата сервера<br>`verify_full` - шифрованное, проверка сертификата сервера и имени сервера с помощью SAN<br>Пустое или не заданное значение считается отсутствием TLS. |
| Plugins.Redis.Sessions.<SessionName>.TLSCAFile | (yes, if TLSConnect is `verify_ca` or `verify_full`) | | | Полный путь к файлу, содержащему сертификаты корневого CA для проверки сертификата сервера Redis.<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных. |
| Plugins.Redis.Sessions.<SessionName>.TLSCertFile | | | | Полный путь к файлу, содержащему сертификат агента для аутентификации клиента.<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных. |
| Plugins.Redis.Sessions.<SessionName>.TLSKeyFile | | | | Полный путь к файлу, содержащему закрытый ключ агента для аутентификации клиента.<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных. |
| Plugins.Redis.Sessions.<SessionName>.Uri | | | localhost:6379 | Строка подключения именованной сессии.<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных.<br><br>Не должен содержать встроенные учетные данные (они будут проигнорированы).<br>Должен соответствовать формату URI.<br>Поддерживаемые схемы: `tcp`, `unix`; схема может быть опущена.<br>Порт может быть опущен (по умолчанию=6379).<br>Примеры: `tcp://localhost:6379`<br>`localhost`<br>`unix:/var/run/redis.sock` |
| Plugins.Redis.Sessions.<SessionName>.User | | | default | Пользователь для отправки на защищенный сервер Redis.<br>**<SessionName>** - задает имя сессии, которое будет использоваться в ключах элементов данных. |
| Plugins.Redis.Timeout | | 1-30 | global timeout | Тайм-аут выполнения запроса (время ожидания завершения запроса в секундах до его остановки). |

См. также:

- Описание общих параметров конфигурации Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Инструкции по настройке плагинов

**14 SMART plugin**

Обзор

В этом разделе перечислены параметры, поддерживаемые в файле конфигурации плагина SMART для агента 2 Zabbix (smart.conf).

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Путь к исполняемому файлу `smartctl` должен быть задан либо путем добавления его каталога в системную переменную среды `PATH`, либо путем настройки `Plugins.Smart.Path`; это относится как к Linux, так и к Windows;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Параметры

| Параметр | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|
| Plugins.Smart.Path | нет | | smartctl | Путь к исполняемому файлу smartctl. |
| Plugins.Smart.Timeout | нет | 1-30 | глобальный тайм-аут | Тайм-аут выполнения запроса (продолжительность в секундах ожидания завершения запроса, прежде чем он будет завершён принудительно). |

Смотрите также:

- Описание общих параметров конфигурации Zabbix агента 2: Zabbix агент 2 (UNIX) / Zabbix агент 2 (Windows)
- Инструкции по настройке плагинов

### 8. Zabbix Java gateway

Если вы используете скрипты `startup.sh` и `shutdown.sh` для запуска Zabbix Java gateway, вы можете указать необходимые параметры конфигурации в файле`settings.sh`. Скрипты запуска и остановки считывают содержимое файла настроек и заботятся о конвертации переменных оболочки (shell variables, приведены в первом столбце) в свойства Java (Java properties, приведены во втором столбце).

Если вы запускаете Zabbix Java gateway вручную, непосредственно с помощью `java`, тогда вы указываете соответствующие свойства Java в командной строке.

| Переменная | Свойство | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|---|
| LISTEN_IP | zabbix.listenIP | нет | | 0.0.0.0 | IP адрес, который слушает Zabbix Java Gateway. |
| LISTEN_PORT | zabbix.listenPort | нет | 1024-32767 | 10052 | Порт, который слушает Zabbix Java Gateway. |
| PID_FILE | zabbix.pidFile | нет | | /tmp/zabbix_java.pid | Имя PID файла. Если не задано, Zabbix Java Gateway запустится как консольное приложение. |
| PROPERTIES_FILE | zabbix.propertiesFile | нет | | | Имя файла свойств. Может использоваться, чтобы задать дополнительные свойства, используя формат ключ-значение, таким образом, что они не будут видимы в командной строке или перезапишут существующие свойства. Например: ”javax.net.ssl.trustStorePassword=<п |

| Переменная | Свойство | Обязательный | Диапазон | По умолчанию | Описание |
|---|---|---|---|---|---|
| START_POLLERS | zabbix.startPollers | нет | 1-1000 | 5 | Количество запускаемых рабочих потоков. |
| TIMEOUT | zabbix.timeout | нет | 1-30 | 3 | Как долго ждать завершения сетевых операций (в секундах). |

**Warning:**
Порт 10052 не зарегистрирован в IANA.

### 9. Zabbix веб-сервис

Обзор

Веб-служба Zabbix — это процесс, который используется для взаимодействия с внешними веб-службами.

Параметры, поддерживаемые файлом конфигурации веб-службы Zabbix (zabbix_web_service.conf), перечислены в этом разделе.

Параметры перечислены без дополнительной информации. Щелкните параметр, чтобы увидеть полные сведения.

| Parameter | Description |
|---|---|
| AllowedIP | Список IP-адресов, разделенных запятыми, при необходимости в нотации CIDR, или DNS-имен серверов Zabbix и прокси Zabbix. |
| DebugLevel | Уровень отладки. |
| IgnoreURLCertErrors | Указывает обработку ошибок проверки TLS-сертификата при доступе к URL веб-интерфейса. |
| Include | Вы можете включить отдельные файлы или все файлы в каталоге в файл конфигурации. |
| ListenPort | Служба будет прослушивать этот порт для подключений от сервера. |
| LogFile | Имя файла журнала. |
| LogFileSize | Максимальный размер файла журнала. |
| LogType | Тип вывода журнала. |
| Timeout | Максимальное время (в секундах), затрачиваемое на формирование PDF отчета панели мониторинга. |
| TLSAccept | Какие входящие подключения принимать. |
| TLSCAFile | Полный путь к файлу, содержащему сертификаты корневого CA(ов), используемые для проверки сертификата узла, для зашифрованного обмена данными между компонентами Zabbix. |
| TLSCertFile | Полный путь к файлу, содержащему сертификат службы или цепочку сертификатов, используемые для зашифрованного обмена данными между компонентами Zabbix. |
| TLSKeyFile | Полный путь к файлу, содержащему закрытый ключ службы, используемый для зашифрованного обмена данными между компонентами Zabbix. |

Все параметры являются необязательными, если явно не указано, что параметр является обязательным.

Обратите внимание:

- Значения по умолчанию отражают значения по умолчанию процесса, а не значения в поставляемых файлах конфигурации;
- Значения поддерживают переменные среды;
- Zabbix поддерживает файлы конфигурации только в кодировке UTF-8 без BOM;
- Комментарии, начинающиеся с ”#”, поддерживаются только в начале строки.

Подробности параметра

AllowedIP

Список (через запятую) IP адресов, опционально в CIDR нотации, или DNS имён Zabbix серверов и Zabbix прокси. Входящие соединения будут приниматься только с хостов, указанных в этом списке.<br>Если включена поддержка IPv6, то `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` обрабатываются одинаково, а `::/0` разрешает все IPv4 и IPv6 адреса. `0.0.0.0/0` можно использовать, чтобы разрешить любой IPv4 адрес.

Пример:

`127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com`

Обязательное: да

DebugLevel

Укажите уровень отладки:<br>*0* — основная информация о запуске и остановке процессов Zabbix<br>*1* — критическая информация;<br>*2* — информация об ошибках;<br>*3* — предупреждения; <br>*4* — для отладки (выдает много информации);<br>*5* — расширенная отладка (выдает еще больше информации).

По умолчанию: 3<br> Диапазон: 0-5

IgnoreURLCertErrors

Определяет обработку ошибок проверки TLS-сертификата при доступе к URL веб-интерфейса:<br>0 - не игнорировать ошибки сертификата;<br>1 - игнорировать ошибки сертификата.<br>

По умолчанию: 0<br> Диапазон: 0-1

Включение

Вы можете включать отдельные файлы или все файлы в каталоге в файл конфигурации. Во время установки Zabbix создаст каталог include в /usr/local/etc, если он не был изменен во время компиляции. Путь может быть относительным к расположению файла *zabbix_web_service.conf*.<br>Чтобы включать только нужные файлы в указанном каталоге, для сопоставления шаблонов поддерживается подстановочный символ звездочка.<br>См. special notes о ограничениях.

Пример:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenPort

Веб-сервис будет прослушивать этот порт на предмет соединений с сервером.

По умолчанию: 10053<br> Диапазон: 1024-32767

LogFile

Имя файла журнала.

Пример:

`/tmp/zabbix_web_service.log`

Обязательное: Да, если LogType выставлен в *file*; в противном случае — нет

LogFileSize

Максимальный размер файла журнала в МБ.<br>0 — отключить автоматическую ротацию журналов.<br>*Примечание*: если достигнут предел размера файла журнала и ротация файла по какой-либо причине не удалась, существующий файл журнала усекается и запускается заново.

По умолчанию: 1<br> Диапазон: 0–1024

Тип журнала

Тип вывода журнала:<br>*file* — запись журнала в файл, указанный в параметре LogFile;<br>*system* — запись журнала в syslog;<br>*console* — вывод журнала в стандартный вывод.

По умолчанию: `file`

Timeout

Максимальное время (в секундах), которое можно потратить на форматирование PDF отчёта панели

По умолчанию: 10<br> Диапазон: 1-30

TLSAccept

Какие входящие подключения принимать:<br>*unencrypted* — принимать подключения без шифрования (по умолчанию)<br>*cert* — принимать подключения с TLS и сертификатом

По умолчанию: `unencrypted`

######TLSCAFile

Абсолютный путь к файлу, который содержит сертификаты удостоверяющих центров (certificate authority, CA) верхнего уровня для проверки сертификата узла, используется для зашифрованных соединений между компонентами Zabbix.

TLSCertFile

Полный путь к файлу, содержащему сертификат сервиса или цепочку сертификатов, используемый для зашифрованной связи с компонентами Zabbix.

TLSKeyFile

Абсолютный путь к файлу, содержащему закрытый ключ сервиса (private key), используется для зашифрованных коммуникаций между компонентами Zabbix.

### 10 Переменные среды

Обзор

Переменные среды позволяют настраивать компоненты Zabbix без жесткого задания значений в файлах конфигурации. Это упрощает управление конфигурациями в динамических средах, таких как Docker, где переменные можно передавать во время выполнения, чтобы адаптировать систему к разным сценариям.

В простейшем случае, задав значение параметра конфигурации DebugLevel для сервера Zabbix через переменную среды, вы можете использовать ее для настройки сервера при запуске:

```
####                Zabbix:
DebugLevel=${NEW_DEBUG_LEVEL}

####          Zabbix:
NEW_DEBUG_LEVEL=5 /usr/sbin/zabbix_server
```

Переменные среды поддерживаются следующими компонентами Zabbix:

- Сервер
- Прокси
- Агент (UNIX или Windows)
- Агент 2 (UNIX или Windows), включая плагины
- Веб-сервис
- Zabbix sender (при использовании опции -c, --config)

Важные примечания

- Если для параметра конфигурации задана переменная среды, которая не указана при запуске компонента, используется значение параметра по умолчанию.
- При использовании команд во время выполнения (например, для увеличения уровня журнала агента) необходимо указывать все ранее использовавшиеся переменные среды. Это связано с тем, что компоненты Zabbix используют свой файл конфигурации для выполнения команд во время выполнения; если переменные среды опущены, будут использованы значения параметров конфигурации по умолчанию. См. Примеры.
- Команда во время выполнения `userparameter_reload` runtime command не поддерживает повторную загрузку переменных среды. Во время перезагрузки переменные игнорируются, и перезагружаются только параметры с обычными значениями.
- Текущие переменные среды процесса, которые использовались в файлах конфигурации, очищаются после запуска компонента Zabbix. Это гарантирует, что дочерние процессы (например, удаленные скрипты, выполняемые Zabbix) не смогут получить доступ к этим переменным. Однако обратите внимание, что начальные переменные процесса по-прежнему можно получить (например, через файл `/proc/<PID>/environ`).

Синтаксис

Переменные среды должны использовать следующий синтаксис: ${буквенно-цифровые_символы/подчеркивания}.

Имя переменной может содержать только буквы (a-z, A-Z), символы подчеркивания (_), и цифры (0-9), и не должно начинаться с цифры.

Переменные, которые не соответствуют требуемому синтаксису или объединены с обычным значением, будут рассматриваться как обычные значения, что может привести к ошибкам.

Правильный синтаксис переменной:

```
DebugLevel=${NEW_DEBUG_LEVEL}
Hostname=${ZBX_HOSTNAME}
LogFile=${LogFile_001}
```

Неправильный синтаксис переменной:

```
DebugLevel=${5_DebugLevel}
Hostname=${ZBX.HOSTNAME 1}
LogFile=/${HOME}/zabbix/zabbix_server.log
```

> **Note:**
> В Windows имена переменных среды не чувствительны к регистру.

Примеры

Следующие примеры показывают, как настроить и использовать переменные среды с компонентами Zabbix.

Пример 1: Настройка и тестирование агента Zabbix

1. Задайте переменные среды в файле конфигурации агента:

```
Hostname=${ZBX_HOSTNAME}
ServerActive=${ServerActive}
```

2. Проверьте файл конфигурации:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

3. Запустите агент с переменными среды:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

При использовании команд во время выполнения (например, для увеличения уровня логирования агента) необходимо указывать все ранее использованные переменные среды:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Это связано с тем, что агент использует свой файл конфигурации для выполнения команд во время выполнения; если переменные среды не указаны, будут использованы значения параметров конфигурации по умолчанию.

Кроме того, после задания переменных среды в файле конфигурации агента вы можете сделать их доступными для процессов (например, с помощью команды `export`). Это снижает риск неожиданного поведения из-за отсутствующих или неверно заданных переменных.

```
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
```

Пример 2: Настройка агента Zabbix для контейнера

Если вы создаете и настраиваете собственный пользовательский образ для компонентов Zabbix (например, агента Zabbix), вы можете задать параметры конфигурации с помощью переменных среды, а затем запустить контейнер с этими переменными.

1. При подготовке образа задайте переменные среды в файле конфигурации агента:

```
Hostname=${ZBX_HOSTNAME}
BufferSize=${BUFSZ}
ListenPort=${LISTENPORT}
UserParameter=${_UsrPar01}
UserParameter=${_UsrPar02}
```

2. После сборки образа контейнера запустите контейнер агента (например, Docker) с переменными среды:

```
docker run --name my-zabbix-agent -e ZBX_HOSTNAME="new-hostname" -e BUFSZ=1000 -e LISTENPORT=20050 -e _Usr
```

3. При использовании команд времени выполнения (например, для увеличения уровня журнала агента) откройте shell контейнера и выполните команду времени выполнения:

```
docker exec -it <containerid> sh
/usr/sbin/zabbix_agentd -R log_level_increase
```

**Attention:**
Команда времени выполнения `userparameter_reload` не поддерживает повторную загрузку переменных среды. Во время повторной загрузки переменные игнорируются, и перезагружаются только параметры с обычными значениями.

#### 11 Включение

Обзор

Используя параметр `Include`, можно к конфигурации сервера, прокси, агента и веб-сервиса добавлять дополнительные файлы или директории.

Примечания по включению

Если параметр `Include` используется для включения файла, файл должен быть доступен для чтения.

Если параметр `Include` используется для включения каталога:

- Все файлы в каталоге должны быть доступны для чтения.
- Не следует предполагать какой-либо определенный порядок включения (например, файлы не включаются в алфавитном порядке). Поэтому не определяйте один параметр в нескольких файлах "Include" (например, чтобы переопределить общую настройку более конкретной).
- Все файлы в каталоге включаются в конфигурацию.
- Остерегайтесь резервных копий файлов, автоматически создаваемых некоторыми текстовыми редакторами. Например, если при редактировании файла "include/my_specific.conf" создается резервная копия "include/my_specific_conf.BAK", то будут включены оба файла. Переместите "include/my_specific.conf.BAK" за пределы каталога "Include". В Linux содержимое каталога "Include" можно проверить командой "ls -al" на наличие ненужных файлов.

Если параметр `Include` используется для включения файлов по шаблону:

- Все файлы, соответствующие шаблону, должны быть доступны для чтения.
- Не следует предполагать какой-либо определенный порядок включения (например, файлы не включаются в алфавитном порядке). Поэтому не определяйте один параметр в нескольких файлах "Include" (например, чтобы переопределить общую настройку более конкретной).

### 3. Протоколы

В этом разделе описаны коммуникационные протоколы, используемые компонентами Zabbix, в том числе обмен данными между сервером и прокси-сервером, связь с агентами и форматы экспорта данных.

Для поиска необходимой спецификации протокола, пожалуйста, воспользуйтесь боковым меню.

#### 1. Протокол обмена данными сервер-прокси

Обзор

Обмен данными сервер — прокси основывается на формате JSON.

Сообщения запроса и ответа должны начинаться с заголовка и длины данных.

Пассивный прокси

Запрос конфигурации

Сервер сначала отправит пустой запрос `proxy config`. Этот запрос отправляется каждые `ProxyConfigFrequency` (параметр конфигурации сервера) секунд.

Прокси отвечает текущей версией прокси, токеном сеанса и ревизией конфигурации. Сервер отвечает данными конфигурации, которые необходимо обновить.

| name | value type | description |
|---|---|---|
| server→proxy: | | |
| **request** | *string* | 'proxy config' |
| proxy→server: | | |

| name | value type | description |
|---|---|---|
| **version** | *string* | Версия прокси (<major>.<minor>.<build>). |
| **session** | *string* | Токен сеанса конфигурации прокси. |
| **config_revision** | *number* | Ревизия конфигурации прокси. |
| server→proxy: | | |
| **full_sync** | *number* | 1 - если отправляются полные данные конфигурации; отсутствует - в противном случае (необязательно). |
| **data** | *array* | Объект табличных данных. Отсутствует, если конфигурация не была изменена (необязательно). |
| **<table>** | *object* | Один или несколько объектов с данными <table> (необязательно, в зависимости от изменений). |
| **fields** | *array* | Массив имен полей. |
| - | *string* | Имя поля. |
| **data** | *array* | Массив строк. |
| - | *array* | Массив столбцов. |
| - | *string,number* | Значение столбца с типом, зависящим от типа столбца в схеме базы данных. |
| **macro.secrets** | *object* | Информация о секретных макросах, отсутствует, если в макросах vault нет изменений (необязательно). |
| **config_revision** | *number* | Ревизия кэша конфигурации - отправляется вместе с данными конфигурации (необязательно). |
| **del_hostids** | *array* | Массив удаленных hostid (необязательно). |
| - | *number* | Идентификатор узла сети. |
| **del_macro_hostids** | *array* | Массив hostid, у которых удалены все макросы (необязательно). |
| - | *number* | Идентификатор узла сети. |
| proxy→server: | | |
| **response** | *string* | Информация об успешности запроса (’success’ или ’failed’). |
| **version** | *string* | Версия прокси (<major>.<minor>.<build>). |

Пример:

server→proxy:

```
{
  "request":"proxy config"
}
```

proxy→server:

```
{
  "version": "7.4.0",
  "session": "0033124949800811e5686dbfd9bcea98",
  "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, "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "5
]
},
"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
}
```


proxy→server:


```
{
  "response": "success",
  "version": "7.4.0"
}
```


Запрос данных

Запрос `proxy data` используется для получения данных о доступности интерфейсов узлов сети, исторических данных, данных обнаружения и данных авто-регистрации от прокси. Этот запрос отправляется каждые `ProxyDataFrequency` (параметр конфигурации сервера) секунд.

| name | value type | description |
|---|---|---|
| server→proxy: | | |
| **request** | *string* | 'proxy data' |
| proxy→server: | | |
| **session** | *string* | Токен сеанса данных. |
| **interface availability** | *array* | *(optional)* Массив объектов данных о доступности интерфейсов. |
| **interfaceid** | *number* | Идентификатор интерфейса. |
| **available** | *number* | Доступность интерфейса:<br><br>**0**, *INTERFACE_AVAILABLE_UNKNOWN* - неизвестно<br>**1**, *INTERFACE_AVAILABLE_TRUE* - доступен<br>**2**, *INTERFACE_AVAILABLE_FALSE* - недоступен |
| **error** | *string* | Сообщение об ошибке интерфейса или пустая строка. |
| **history data** | *array* | *(optional)* Массив объектов исторических данных. |
| **itemid** | *number* | Идентификатор элемента данных. |
| **clock** | *number* | Метка времени значения элемента данных (секунды). |
| **ns** | *number* | Метка времени значения элемента данных (наносекунды). |
| **value** | *string* | *(optional)* Значение элемента данных. |
| **id** | *number* | Идентификатор значения (возрастающий счетчик, уникальный в пределах одного сеанса данных). |
| **timestamp** | *number* | *(optional)* Метка времени для элементов данных типа log. |
| **source** | *string* | *(optional)* Значение источника элемента данных eventlog. |
| **severity** | *number* | *(optional)* Значение серьезности элемента данных eventlog. |
| **eventid** | *number* | *(optional)* Значение eventid элемента данных eventlog. |

| name | value type | description |
|---|---|---|
| **state** | *string* | *(optional)* Состояние элемента данных:<br>**0**, *ITEM_STATE_NORMAL*<br>**1**, *ITEM_STATE_NOTSUPPORTED* |
| **lastlogsize** | *number* | *(optional)* Последний размер журнала для элементов данных типа log. |
| **mtime** | *number* | *(optional)* Время изменения для элементов данных типа log. |
| **discovery data** | *array* | *(optional)* Массив объектов данных обнаружения. |
| **clock** | *number* | Метка времени данных обнаружения. |
| **druleid** | *number* | Идентификатор правила обнаружения. |
| **dcheckid** | *number* | Идентификатор проверки обнаружения или null для данных правила обнаружения. |
| **type** | *number* | Тип проверки обнаружения:<br><br>**-1** данные правила обнаружения<br>**0**, *SVC_SSH* - проверка службы SSH<br>**1**, *SVC_LDAP* - проверка службы LDAP<br>**2**, *SVC_SMTP* - проверка службы SMTP<br>**3**, *SVC_FTP* - проверка службы FTP<br>**4**, *SVC_HTTP* - проверка службы HTTP<br>**5**, *SVC_POP* - проверка службы POP<br>**6**, *SVC_NNTP* - проверка службы NNTP<br>**7**, *SVC_IMAP* - проверка службы IMAP<br>**8**, *SVC_TCP* - проверка доступности TCP-порта<br>**9**, *SVC_AGENT* - агент Zabbix<br>**10**, *SVC_SNMPv1* - агент SNMPv1<br>**11**, *SVC_SNMPv2* - агент SNMPv2<br>**12**, *SVC_ICMPPING* - ICMP ping<br>**13**, *SVC_SNMPv3* - агент SNMPv3<br>**14**, *SVC_HTTPS* - проверка службы HTTPS<br>**15**, *SVC_TELNET* - проверка доступности Telnet |
| **ip** | *string* | IP-адрес узла сети. |
| **dns** | *string* | DNS-имя узла сети. |
| **port** | *number* | *(optional)* Номер порта службы. |
| **key_** | *string* | *(optional)* Ключ элемента данных для проверки обнаружения типа **9** *SVC_AGENT* |
| **value** | *string* | *(optional)* Значение, полученное от службы, для большинства служб может быть пустым. |
| **status** | *number* | *(optional)* Состояние службы:<br><br>**0**, *DOBJECT_STATUS_UP* - служба доступна<br>**1**, *DOBJECT_STATUS_DOWN* - служба недоступна |
| **auto registration** | *array* | *(optional)* Массив объектов данных авто-регистрации. |
| **clock** | *number* | Метка времени данных авто-регистрации. |
| **host** | *string* | Имя узла сети. |
| **ip** | *string* | *(optional)* IP-адрес узла сети. |
| **dns** | *string* | *(optional)* Разрешенное DNS-имя по IP-адресу. |
| **port** | *string* | *(optional)* Порт узла сети. |
| **host_metadata** | *string* | *(optional)* Метаданные узла сети, отправленные агентом (на основе параметра конфигурации агента HostMetadata или HostMetadataItem). |
| **tasks** | *array* | *(optional)* Массив задач. |
| **type** | *number* | Тип задачи:<br><br>**0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - результат удаленной команды |

| name | value type | description |
|---|---|---|
| **status** | *number* | Состояние выполнения удаленной команды:<br><br>**0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - удаленная команда успешно выполнена<br>**1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - выполнение удаленной команды завершилось ошибкой |
| **error** | *string* | *(optional)* Сообщение об ошибке. |
| **parent_taskid** | *number* | ID родительской задачи. |
| **more** | *number* | *(optional)* 1 - осталось еще исторических данных для отправки. |
| **clock** | *number* | *(optional)* Метка времени передачи данных (секунды). |
| **ns** | *number* | *(optional)* Метка времени передачи данных (наносекунды). |
| **version** | *string* | Версия прокси (<major>.<minor>.<build>). |
| server→proxy: | | |
| **response** | *string* | Информация об успешности запроса (’success’ или ’failed’). |
| **tasks** | *array* | *(optional)* Массив задач. |
| **type** | *number* | Тип задачи:<br><br>**1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - удаленная команда |
| **clock** | *number* | Время создания задачи. |
| **ttl** | *number* | Время в секундах, после которого задача истекает. |
| **commandtype** | *number* | Тип удаленной команды:<br><br>**0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - использовать пользовательский скрипт<br>**1**, *ZBX_SCRIPT_TYPE_IPMI* - использовать IPMI<br>**2**, *ZBX_SCRIPT_TYPE_SSH* - использовать SSH<br>**3**, *ZBX_SCRIPT_TYPE_TELNET* - использовать Telnet<br>**4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - использовать глобальный скрипт (в настоящее время функционально эквивалентен пользовательскому скрипту) |
| **command** | *string* | Удаленная команда для выполнения. |
| **execute_on** | *number* | Целевой объект выполнения для пользовательских скриптов:<br><br>**0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - выполнить скрипт на агенте<br>**1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - выполнить скрипт на сервере<br>**2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - выполнить скрипт на прокси |
| **port** | *number* | *(optional)* Порт для команд Telnet и SSH. |
| **authtype** | *number* | *(optional)* Тип аутентификации для команд SSH. |
| **username** | *string* | *(optional)* Имя пользователя для команд Telnet и SSH. |
| **password** | *string* | *(optional)* Пароль для команд Telnet и SSH. |
| **publickey** | *string* | *(optional)* Открытый ключ для команд SSH. |
| **privatekey** | *string* | *(optional)* Закрытый ключ для команд SSH. |
| **parent_taskid** | *number* | ID родительской задачи. |
| **hostid** | *number* | ID целевого узла сети. |

Пример:

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":"7.4.0"
}
```

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
      }
  ]
}
```

Запрос задач

Обмен `proxy tasks` обеспечивает доставку и подтверждение задач удаленных команд между сервером и пассивным прокси. Когда сервер опрашивает прокси на предмет результатов задач, он отправляет пустой запрос `proxy tasks` с интервалом, определяемым параметром `ProxyDataFrequency`. Прокси отвечает своей текущей версией и любыми ожидающими результатами задач (включая временные метки). Затем сервер подтверждает получение, возвращая `response`, и может включить новые задачи для выполнения прокси.

| name | value type | description |
|---|---|---|
| server→proxy: | | |
| **request** | *string* | 'proxy tasks' |
| proxy→server: | | |
| **version** | *string* | Версия прокси (<major>.<minor>.<build>). |
| **clock** | *number* | *(optional)* Временная метка передачи данных (секунды). |
| **ns** | *number* | *(optional)* Временная метка передачи данных (наносекунды). |
| server→proxy: | | |
| **response** | *string* | Информация об успешности запроса ('success' or 'failed'). |

Пример:

server→proxy:

```
{
  "request":"proxy tasks"
}
```

proxy→server:

```
{
  "version":"7.0.0",
  "clock":1721059872,
  "ns":848141522
}
```

server→proxy:

```
{
  "response":"success"
}
```

Активный прокси

Запрос конфигурации

Запрос `proxy config` отправляется активным прокси для получения данных конфигурации прокси. Этот запрос отправляется каждые `ProxyConfigFrequency` секунд (параметр конфигурации прокси).

| name | value type | description |
|---|---|---|
| proxy→server: | | |
| **request** | *string* | 'proxy config' |
| **host** | *string<br>* | Имя прокси. |
| **version** | *string* | Версия прокси (<major>.<minor>.<build>). |
| **session** | *string* | Токен сеанса конфигурации прокси. |
| **config_revision** | *number* | Ревизия конфигурации прокси. |
| **hostmap_revision** | *number* | Текущая ревизия сопоставления узел сети-прокси. |
| server→proxy: | | |
| **full_sync** | *number* | 1 - если отправляются полные данные конфигурации, иначе отсутствует (необязательно). |
| **data** | *array* | Объект табличных данных. Отсутствует, если конфигурация не изменилась (необязательно). |
| **<table>** | *object* | Один или несколько объектов с данными <table> (необязательно, в зависимости от изменений). |
| **fields** | *array* | Массив имен полей. |
| - | *string* | Имя поля. |
| **data** | *array* | Массив строк. |
| - | *array* | Массив столбцов. |

| name | value type | description |
|---|---|---|
| - | *string,number* | Значение столбца с типом, зависящим от типа столбца в схеме базы данных. |
| **macro.secrets** | *object* | Информация о секретных макросах, отсутствует, если в макросах vault нет изменений (необязательно). |
| **proxy_group** | *string* | Имя группы прокси, к которой принадлежит прокси. |
| **config_revision** | *number* | Ревизия кэша конфигурации - отправляется вместе с данными конфигурации (необязательно). |
| **del_hostids** | *array* | Массив удаленных hostid (необязательно). |
| - | *number* | Идентификатор узла сети. |
| **del_macro_hostids** | *array* | Массив hostid, у которых удалены все макросы (необязательно). |
| - | *number* | Идентификатор узла сети. |

Пример:

proxy→server:

```
{
  "request": "proxy config",
  "host": "Zabbix proxy",
  "version":"7.4.0",
  "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, "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "", "5
]
},
"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
}
```


Запрос данных

Запрос `proxy data` отправляется прокси для передачи данных о доступности интерфейсов узлов сети, истории, обнаружении и авторегистрации. Этот запрос отправляется каждые `DataSenderFrequency` секунд (параметр конфигурации прокси). Обратите внимание, что активный прокси по-прежнему будет опрашивать сервер Zabbix каждую секунду на наличие задач удаленных команд (с пустым запросом `proxy data`).

| name | value type | description |
|---|---|---|
| proxy→server: | | |
| **request** | *string* | 'proxy data' |
| **host** | *string* | Имя прокси. |
| **session** | *string* | Токен сеанса данных. |
| **interface availability** | *array* | *(optional)* Массив объектов данных о доступности интерфейсов. |
| **interfaceid** | *number* | Идентификатор интерфейса. |
| **available** | *number* | Доступность интерфейса:<br><br>**0**, *INTERFACE_AVAILABLE_UNKNOWN* - неизвестно<br>**1**, *INTERFACE_AVAILABLE_TRUE* - доступен<br>**2**, *INTERFACE_AVAILABLE_FALSE* - недоступен |
| **error** | *string* | Сообщение об ошибке интерфейса или пустая строка. |
| **history data** | *array* | *(optional)* Массив объектов данных истории. |
| **itemid** | *number* | Идентификатор элемента данных. |
| **clock** | *number* | Метка времени значения элемента данных (секунды). |
| **ns** | *number* | Метка времени значения элемента данных (наносекунды). |
| **value** | *string* | *(optional)* Значение элемента данных. |
| **id** | *number* | Идентификатор значения (возрастающий счетчик, уникальный в рамках одного сеанса данных). |
| **timestamp** | *number* | *(optional)* Метка времени элементов данных типа log. |
| **source** | *string* | *(optional)* Значение source элемента данных eventlog. |
| **severity** | *number* | *(optional)* Значение severity элемента данных eventlog. |
| **eventid** | *number* | *(optional)* Значение eventid элемента данных eventlog. |
| **state** | *string* | *(optional)* Состояние элемента данных:<br>**0**, *ITEM_STATE_NORMAL*<br>**1**, *ITEM_STATE_NOTSUPPORTED* |
| **lastlogsize** | *number* | *(optional)* Последний размер журнала для элементов данных типа log. |
| **mtime** | *number* | *(optional)* Время изменения для элементов данных типа log. |
| **discovery data** | *array* | *(optional)* Массив объектов данных обнаружения. |
| **clock** | *number* | Метка времени данных обнаружения. |
| **druleid** | *number* | Идентификатор правила обнаружения. |
| **dcheckid** | *number* | Идентификатор проверки обнаружения или null для данных правила обнаружения. |
| **type** | *number* | Тип проверки обнаружения:<br><br>**-1** данные правила обнаружения<br>**0**, *SVC_SSH* - проверка службы SSH<br>**1**, *SVC_LDAP* - проверка службы LDAP<br>**2**, *SVC_SMTP* - проверка службы SMTP<br>**3**, *SVC_FTP* - проверка службы FTP<br>**4**, *SVC_HTTP* - проверка службы HTTP<br>**5**, *SVC_POP* - проверка службы POP<br>**6**, *SVC_NNTP* - проверка службы NNTP<br>**7**, *SVC_IMAP* - проверка службы IMAP<br>**8**, *SVC_TCP* - проверка доступности TCP-порта<br>**9**, *SVC_AGENT* - агент Zabbix<br>**10**, *SVC_SNMPv1* - агент SNMPv1<br>**11**, *SVC_SNMPv2* - агент SNMPv2<br>**12**, *SVC_ICMPPING* - ping ICMP<br>**13**, *SVC_SNMPv3* - агент SNMPv3<br>**14**, *SVC_HTTPS* - проверка службы HTTPS<br>**15**, *SVC_TELNET* - проверка доступности Telnet |

| name | value type | description |
|---|---|---|
| **ip** | *string* | IP-адрес узла сети. |
| **dns** | *string* | DNS-имя узла сети. |
| **port** | *number* | *(optional)* Номер порта службы. |
| **key_** | *string* | *(optional)* Ключ элемента данных для проверки обнаружения типа **9** *SVC_AGENT* |
| **value** | *string* | *(optional)* Значение, полученное от службы, для большинства служб может быть пустым. |
| **status** | *number* | *(optional)* Статус службы:<br><br>**0**, *DOBJECT_STATUS_UP* - служба доступна<br>**1**, *DOBJECT_STATUS_DOWN* - служба недоступна |
| **autoregistration** | *array* | *(optional)* Массив объектов данных авторегистрации. |
| **clock** | *number* | Метка времени данных авторегистрации. |
| **host** | *string* | Имя узла сети. |
| **ip** | *string* | *(optional)* IP-адрес узла сети. |
| **dns** | *string* | *(optional)* Разрешенное DNS-имя по IP-адресу. |
| **port** | *string* | *(optional)* Порт узла сети. |
| **host_metadata** | *string* | *(optional)* Метаданные узла сети, отправленные агентом (на основе параметра конфигурации агента HostMetadata или HostMetadataItem). |
| **tasks** | *array* | *(optional)* Массив задач. |
| **type** | *number* | Тип задачи:<br><br>**0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - результат удаленной команды |
| **status** | *number* | Статус выполнения удаленной команды:<br><br>**0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - удаленная команда успешно выполнена<br>**1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - удаленная команда не выполнена |
| **error** | *string* | *(optional)* Сообщение об ошибке. |
| **parent_taskid** | *number* | ID родительской задачи. |
| **more** | *number* | *(optional)* 1 - есть еще данные истории для отправки |
| **clock** | *number* | *(optional)* Метка времени передачи данных (секунды). |
| **ns** | *number* | *(optional)* Метка времени передачи данных (наносекунды). |
| **version** | *string* | Версия прокси (<major>.<minor>.<build>). |
| server→proxy: | | |
| **response** | *string* | Информация об успешности запроса (’success’ или ’failed’). |
| **upload** | *string* | Управление загрузкой исторических данных (history, autoregistration, host availability, network discovery).<br><br>Возможные значения:<br>**enabled** - нормальная работа<br>**disabled** - сервер не принимает данные (возможно, из-за превышения внутреннего кэша) |
| **tasks** | *array* | *(optional)* Массив задач. |
| **type** | *number* | Тип задачи:<br><br>**1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - удаленная команда |
| **clock** | *number* | Время создания задачи. |
| **ttl** | *number* | Время в секундах, по истечении которого задача становится недействительной. |
| **commandtype** | *number* | Тип удаленной команды:<br><br>**0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - использовать пользовательский скрипт<br>**1**, *ZBX_SCRIPT_TYPE_IPMI* - использовать IPMI<br>**2**, *ZBX_SCRIPT_TYPE_SSH* - использовать SSH<br>**3**, *ZBX_SCRIPT_TYPE_TELNET* - использовать Telnet<br>**4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - использовать глобальный скрипт (в настоящее время функционально эквивалентен пользовательскому скрипту) |

| name | value type | description |
|---|---|---|
| **command** | *string* | Удаленная команда для выполнения. |
| **execute_on** | *number* | Цель выполнения для пользовательских скриптов:<br><br>**0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - выполнить скрипт на агенте<br>**1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - выполнить скрипт на сервере<br>**2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - выполнить скрипт на прокси |
| **port** | *number* | *(optional)* Порт для команд Telnet и SSH. |
| **authtype** | *number* | *(optional)* Тип аутентификации для команд SSH. |
| **username** | *string* | *(optional)* Имя пользователя для команд Telnet и SSH. |
| **password** | *string* | *(optional)* Пароль для команд Telnet и SSH. |
| **publickey** | *string* | *(optional)* Открытый ключ для команд SSH. |
| **privatekey** | *string* | *(optional)* Закрытый ключ для команд SSH. |
| **parent_taskid** | *number* | ID родительской задачи. |
| **hostid** | *number* | ID целевого узла сети. |

Пример:

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": "7.4.0",
"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
}]
}
```


#### 2. Протокол Zabbix агент/агент2

Пожалуйста, обратитесь к странице пассивные и активные проверки агента для получения более подробной информации о протоколах Zabbix агента и Zabbix агента 2.

#### 4. Протокол плагина Zabbix Агента 2

Протокол Zabbix агента 2 основан на коде, размере и модели данных.

Код

| Тип | Размер | Комментарии |
|---|---|---|
| Byte | 4 | Тип полезной нагрузки, в настоящее время поддерживается только JSON. |

Размер

| Тип | Размер | Комментарии |
|---|---|---|
| Byte | 4 | Размер текущей полезной нагрузки в байтах. |

Данные полезной нагрузки

| Тип | Размер | Комментарии |
|---|---|---|
| Byte | Определяется полем *Размер* | Данные в формате JSON. |

Определение данных полезной нагрузки

Общие данные

Эти параметры присутствуют во всех вопросах/ответах:

| Имя | Тип | Комментарии |
|---|---|---|
| id | uint32 | Для запросов — возрастающий идентификатор, используемый, чтобы связать запросы с ответами. Уникален в пределах направления запроса (т.е. от агента к плагину или от плагина к агенту).<br>Для ответов — идентификатор соответствующего запроса. |
| type | uint32 | Тип запроса. |

Запрос журналирования

Запрос отсылается плагином, чтобы записать журнальное сообщение в журнальный файл агента.

| | |
|---|---|
| направление | плагин → агент |
| ответ | нет |

Параметры, специфичные для запросов журналирования:

| Имя | Тип | Комментарии |
|---|---|---|
| severity | uint32 | Важность сообщения (уровень журналирования). |
| message | string | Сообщение для журнала. |

*Пример:*

```
{"id":0,"type":1,"severity":3,"message":"      "}
```

Запрос регистрации

Запрос отсылается агентом в фазе начальной загрузки агента, чтобы получить предоставляемые метрики для регистрации плагина.

| | |
|---|---|
| направление | агент → плагин |
| ответ | да |

Параметры, специфичные для запросов регистрации:

| Имя | Тип | Комментарии |
|---|---|---|
| version | string | Версия протокола <major>.<minor> |

*Пример:*

```
{"id":1,"type":2,"version":"1.0"}
```

Ответ на регистрацию

Ответ плагина на запрос на регистрацию.

| | |
|---|---|
| направление | плагин → агент |
| ответ | неприменимо |

Параметры, специфичные для ответов на регистрацию:

| Имя | Тип | Комментарии |
|---|---|---|
| name | string | Имя плагина. |
| metrics | array of strings (опционально) | Метрики с описаниями, как используется в плагине. Возвращает RegisterMetrics(). Отсутствует, если возвращается ошибка. |
| interfaces | uint32 (опционально) | Битовая маска интерфейсов, поддерживаемых плагином. Отсутствует, если возвращается ошибка. |
| error | string (опционально) | Сообщение об ошибке возвращается, если плагин не может запуститься. Отсутствует, если возвращаются метрики. |

*Примеры:*

```
{"id":2,"type":3,"metrics":["external.test", "                  ."], "interfaces": 4}
```

или

```
{"id":2,"type":3,"error":"           "}
```

Запрос на запуск

Запрос на выполнение функции запуска (Start) интерфейса Runner.

| | |
|---|---|
| направление | агент → плагин |
| ответ | нет |

Запрос не имеет специфичных параметров, он содержит только параметры общих данных.

*Пример:*

```
{"id":3,"type":4}
```

Запрос на останов

Запрос отсылается агентом, чтобы остановить плагин.

| | |
|---|---|
| направление | агент → плагин |
| ответ | нет |

Запрос не имеет специфичных параметров, он содержит только параметры общих данных.

*Пример:*

```
{"id":3,"type":5}
```

Запрос экспорта

Запрос на выполнение функции экспорта (Export) интерфейса Exporter.

| | |
|---|---|
| направление | агент → плагин |
| ответ | нет |

Параметры, специфичные для запросов экспорта:

| Имя | Тип | Комментарии |
|---|---|---|
| key | string | Ключ плагина. |
| parameters | array of strings (опционально) | Параметры для функции Export. |

*Пример:*

```
{"id":4,"type":6,"key":"test.key","parameters":["foo","bar"]}
```

Ответ экспорта

Ответ из функции Export интерфейса Exporter.

| | |
|---|---|
| направление | плагин → агент |
| ответ | неприменимо |

Параметры, специфичные для ответов экспорта:

| Имя | Тип | Комментарии |
|---|---|---|
| value | string (опционально) | Значение ответа из функции Export. Отсутствует, если возвращается ошибка. |
| error | string (опционально) | Сообщение об ошибке, если функция Export не была выполнена успешно. Отсутствует, если возвращается значение (value). |

*Примеры:*

```
{"id":5,"type":7,"value":"response"}
```

или

```
{"id":5,"type":7,"error":"            "}
```

Запрос на конфигурацию

Запрос на выполнение функции *Configure* интерфейса *Configurator*.

| | |
|---|---|
| направление | агент → плагин |
| ответ | неприменимо |

Параметры, специфичные для запросов *Configure*:

| Имя | Тип | Комментарии |
|---|---|---|
| global_options | объект JSON | Объект JSON, содержащий глобальные опции конфигурации агента. |
| private_options | объект JSON (опционально) | Объект JSON, содержащий персональные опции конфигурации плагина, если есть. |

*Пример:*

```
{"id":6,"type":8,"global_options":{...},"private_options":{...}}
```

Запрос на валидацию

Запрос на выполнение функции *Validate* интерфейса *Configurator*.

| | |
|---|---|
| направление | агент → плагин |
| ответ | да |

Параметры, специфичные для запросов *Validate*:

| Имя | Тип | Комментарии |
|---|---|---|
| private_options | объект JSON (опционально) | Объект JSON, содержащий персональные опции конфигурации плагина, если есть. |

*Пример:*

```
{"id":7,"type":9,"private_options":{...}}
```

Ответ на валидацию

Ответ от функции *Validate* интерфейса *Configurator*.

| | |
|---|---|
| направление | плагин → агент |
| ответ | неприменимо |

Параметры, специфичные для ответов на *Validate*:

| Имя | Тип | Комментарии |
|---|---|---|
| error | string (опционально) | Сообщение об ошибке, возвращаемое, если функция Validate не была выполнена успешно. Отсутствует при успешном завершении. |

*Примеры:*

```
{"id":8,"type":10}
```

или

```
{"id":8,"type":10,"error":"           "}
```

**5. Протокол Zabbix sender**

Обзор

Zabbix сервер и Zabbix прокси используют протокол связи на основе JSON для получения данных от утилиты Zabbix sender. Данные можно получить с помощью траппер элемента данных или элемента данных HTTP-агента с активированными трапами.

Сообщения запроса и ответа должны начинаться с заголовка и длины данных.

Запрос Zabbix sender

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<        >",
            "key": "trap",
            "value": "          "
        }
    ]
}
```

Ответ Zabbix сервера

```
{
    "response": "success",
    "info": "processed: 1; failed: 0; total: 1; seconds spent: 0.060753"
}
```

Запрос Zabbix sender со штампом времени

В качестве альтернативы Zabbix sender может отправить запрос со штампом времени и наносекундами.

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<        >",
            "key": "trap",
            "value": "          ",
            "clock": 1516710794,
            "ns": 592397170
        },
        {
            "host": "<        >",
            "key": "trap",
            "value": "          ",
            "clock": 1516710795,
            "ns": 192399456
        }
    ],
    "clock": 1516712029,
    "ns": 873386094
}
```


Ответ Zabbix сервера


```
{
    "response": "success",
    "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.060904"
}
```


#### 6. Заголовок

Обзор

Заголовок присутствует во всех сообщениях запросов и ответов между компонентами Zabbix. Требуется определить длину сообщения, сжато оно или нет, большой это пакет или нет.

Протокол связи Zabbix имеет ограничение размера пакета в 1 ГБ на одно соединение. Это ограничение в 1 ГБ относится как к длине полученных пакетных данных, так и к длине несжатых данных.

При отправке конфигурации на прокси-сервер Zabbix ограничение размера пакета увеличено до 4 ГБ, чтобы позволить синхронизацию больших конфигураций. Когда длина данных до сжатия превышает 4 ГБ, сервер Zabbix автоматически начинает использовать формат больших пакетов (флаг 0x04), что повышает ограничение размера пакета до 16 ГБ.

Обратите внимание, что хотя для отправки любых данных можно использовать большой формат пакетов, в настоящее время только процесс синхронизации конфигурации Zabbix-прокси (configuration syncer) может обрабатывать пакеты размером более 1 ГБ.

Структура

Заголовок состоит из четырёх полей. Все числа в заголовке имеют прямой порядок байтов (little-endian).

| Поле | Размер | Размер (большой пакет) | Описание |
|---|---|---|---|
| <  > | 4 | 4 | "ZBXD" или 5A 42 58 44 |
| <  > | 1 | 1 | Флаги протокола:<br>0x01 — протокол связи Zabbix<br>0x02 — сжатие<br>0x04 — большой пакет |
| <  > | 4 | 8 | Длина данных. |
| <  > | 4 | 8 | При использовании сжатия (флаг 0x02) — длина несжатых данных<br>Когда сжатие не используется — 00 00 00 00 |

Примеры

Здесь представлены несколько фрагментов кода, демонстрирующих, как к тем данным, которые вы хотите отослать (переменная data), добавить заголовок протокола Zabbix, чтобы пакет, который вам нужно отправить в Zabbix, получился таким, чтобы он правильно интерпретировался. Эти фрагменты кода подразумевают, что данные не превышают 1 ГБ, поэтому формат больших пакетов не используется.

Python

```
packet = b"ZBXD\1" + struct.pack("<II", len(data), 0) + data
```

или

```
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;
```

или

```
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;
```

или

```
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"
```

#### 7. Протокол экспорта JSON с переводом строки в качестве разделителя

В этом разделе представлены сведения о протоколе экспорта в в формате JSON с переводом строки в качестве разделителя, который используется при:

- экспорте данных в файлы
- потоковой передаче во внешние системы

Можно экспортировать следующее:

- триггерные события
- значения элементов данных
- динамику изменений (только экспорт в файлы)

Все файлы имеют расширение .ndjson. Каждая строка файла экспорта является объектом JSON.

Триггерные события

Для события проблемы экспортируется следующая информация:

| Поле | Тип | Описание |
|---|---|---|
| *clock* | число | Количество секунд с начала эпохи до момента обнаружения проблемы (целая часть). |
| *ns* | число | Количество наносекунд, которые необходимо добавить к `clock`, чтобы получить точное время обнаружения проблемы. |
| *value* | число | 1 (всегда). |
| *eventid* | число | Идентификатор события проблемы. |
| *name* | строка | Имя события проблемы. |
| *severity* | число | Важность события проблемы (0 — Не классифицировано, 1 — Информация, 2 — Предупреждение, 3 — Средняя, 4 — Высокая, 5 — Чрезвычайная). |
| *hosts* | массив | Список узлов сети, участвующих в выражении триггера; в массиве должен быть хотя бы один элемент. |
| - | объект | |
| *host* | строка | Имя узла сети. |
| *name* | строка | Видимое имя узла сети. |
| *groups* | массив | Список групп всех узлов сети, участвующих в выражении триггера; в массиве должен быть хотя бы один элемент. |
| - | строка | Имя группы узлов сети. |
| *tags* | массив | Список тегов проблемы (может быть пустым). |
| - | объект | |
| *tag* | строка | Имя тега. |

| Поле | Тип | Описание |
|---|---|---|
| *value* | строка | Значение тега (может быть пустым). |

Для события восстановления экспортируется следующая информация:

| Поле | Тип | Описание |
|---|---|---|
| *clock* | число | Количество секунд с начала эпохи до момента восстановления проблемы (целая часть). |
| *ns* | число | Количество наносекунд, которые нужно добавить к `clock`, чтобы получить точное время восстановления проблемы. |
| *value* | число | 0 (всегда). |
| *eventid* | число | Идентификатор события восстановления. |
| *p_eventid* | число | Идентификатор события проблемы. |

Примеры

Проблема:

```
{"clock":1519304285,"ns":123456789,"value":1,"name":"Either Zabbix agent is unreachable on Host B or polle
```

Восстановление:

```
{"clock":1519304345,"ns":987654321,"value":0,"eventid":43,"p_eventid":42}
```

Проблема (множественный режим генерации событий ПРОБЛЕМА):

```
{"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
```

Восстановление:

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":43}
```

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":42}
```

Значения элементов данных

Для собранного значения элемента данных экспортируется следующая информация:

| Поле | Тип | Описание |
|---|---|---|
| *host* | объект | Имя узла сети этого элемента данных. |
| host | строка | Имя узла сети. |
| name | строка | Видимое имя узла сети. |
| *groups* | массив | Список групп узлов сети для узла сети этого элемента данных; в массиве должен быть по крайней мере один элемент. |
| - | строка | Имя группы узла сети. |
| *item_tags* | массив | Список тегов элемента данных (может быть пустым). |
| - | объект | |
| *tag* | строка | Имя тега. |
| *value* | строка | Значение тега (может быть пустым). |
| *itemid* | число | ID элемента данных. |
| *name* | строка | Видимое имя элемента данных. |
| *clock* | число | Количество секунд с начала эпохи до момента сбора значения (целая часть). |
| *ns* | число | Количество наносекунд, которые нужно добавить к `clock` для получения точного времени сбора значения. |
| *timestamp* (только *Журнал (лог)*) | число | 0 если недоступно. |

| Поле | Тип | Описание |
|---|---|---|
| *source* (только *Журнал (лог)*) | строка | Пустая строка если недоступно. |
| *severity* (только *Журнал (лог)*) | число | 0 если недоступно. |
| *eventid* (только *Журнал (лог)*) | число | 0 если недоступно. |
| *value* | число(для числовых элементов данных) или строка (для текстовых элементов данных) | Собранное значение элемента данных. |
| *type* | число | Тип собранного значения:<br>0 — Числовой (с плавающей точной), 1 — Символ, 2 — Журнал (лог), 3 — Числовой (целое положительное), 4 — Текст |

Примеры

Числовое (беззнаковое) значение:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Числовое (с плавающей точкой) значение:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Символьное, текстовое значение:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Запись журнала (лога):

```
{"host":{"host":"Host A","name":"Host A visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Динамика изменений

Для подсчитанного значения динамики изменений экспортируется следующая информация:

| Поле | Тип | Описание |
|---|---|---|
| *host* | объект | Имя узла сети этого элемента данных. |
| host | строка | Имя узла сети. |
| name | строка | Видимое имя узла сети. |
| *groups* | массив | Список групп узлов сети этого элемента данных; в массиве должен быть по крайней мере один элемент. |
| - | строка | Имя узла сети. |
| *item_tags* | массив | Список тегов элемента данных (может быть пустым). |
| - | объект | |
| *tag* | строка | Имя тега. |
| *value* | строка | Значение тега (может быть пустым). |
| *itemid* | число | ID элемента данных. |
| *name* | строка | Видимое имя элемента данных. |
| *clock* | число | Количество секунд с начала эпохи до момента получения значения (целая часть). |
| *count* | число | Количество значений, собранных за данный час. |
| *min* | число | Наименьшее значение элемента данных за данный час. |
| *avg* | число | Среднее значение элемента данных за данный час. |
| *max* | число | Наибольшее значение элемента данных за данный час. |

| Поле | Тип | Описание |
|---|---|---|
| *type* | число | Тип значения:<br>0 — числовой (с плавающей точкой), 3 — числовой беззнаковый |

Примеры

Числовое (беззнаковое) значение:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Числовое (с плавающей точкой) значение:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

#### 4. Элементы данных

В этом разделе содержится справочная информация и технические примечания об элементах данных, включая подробные сведения о параметрах для конкретных ключей элементов данных, сведения о платформенной совместимости и примечания по настройке конкретных элементов данных.

Пожалуйста, воспользуйтесь боковым меню для поиска информации, относящейся к вашему сценарию использования.

#### 1. Параметры vm.memory.size

Обзор

В этом разделе представлены некоторые подробности относительно элемента данных vm.memory.size[<режим>] агента Zabbix.

Параметры

В этом элементе данных разрешены следующие параметры:

- **active** — память, используемая в данный момент или бывшвая в использовании совсем недавно, и поэтому ещё находится в RAM.
- **anon** — память, не связанная с файлами (повторное чтение из них невозможно).
- **available** — доступная память, вычисляется по-разному в зависимости от платформы (смотрите таблицу ниже).
- **buffers** — кэш для таких вещей, как метаданные файловой системы.
- **cached** — кэш для различных вещей.
- **exec** — исполняемый код, в основном из (программ) файлов.
- **file** — кэш содержимого недавно использованных файлов.
- **free** — память, которая доступна без каких-либо проблем любому объекту, запрашивающему память.
- **inactive** — память, помеченная как неиспользуемая.
- **pavailable** — «available» память в процентах по отношению к «total» (рассчитывается как `available/total*100`).
- **pinned** — то же, что и «wired».
- **pused** — «used» память в процентах по отношению к «total» (рассчитывается как `used/total*100`).
- **shared** — память, которая может быть доступна сразу нескольким процессам.
- **slab** — общий объем памяти, которая используется ядром для кэширования структур данных для собственного использования.
- **total** — общий объем доступной физической памяти.
- **used** — используемая память, вычисляется по-разному в зависимости от платформы (смотри таблицу ниже).
- **wired** — память, помеченная всегда оставаться в RAM. Она не может быть перемещена на диск.

**Warning:**
Некоторые из этих параметров работают только для конкретных платформ и могут быть недоступны на вашей платформе. См. элементы данных Zabbix агента.

Вычисления **available** и **used** в зависимости от платформы:

| Платформа | **”available”** | **”used”** |
|---|---|---|
| *AIX* | free + cached | реальное использование памяти |

| Платформа | **”available”** | **”used”** |
|---|---|---|
| *FreeBSD* | inactive + cached + free | active + wired + cached |
| *HP UX* | free | total - free |
| *Linux<3.14* | free + buffers + cached | total - free |
| *Linux 3.14+* (также перенесено для 3.10 на RHEL 7) | /proc/meminfo, см. описание ”MemAvailable” в документации ядра Linux [en]. Обратите внимание, что free + buffers + cached больше не равняется ’available’, так как не весь кэш страницы может быть освобождён и в расчетах используется минимальный объем свободной памяти, зарезервированной системой (low watermark). | 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:**
Сумма *vm.memory.size[used]* и *vm.memory.size[available]* не обязательно равна общему количеству памяти. Например, для FreeBSD:<br> * Активная, неактивная, wired, кэшируемая памяти считаются использованными, так как содержат некоторую полезную информацию.<br> * В то же время неактивная, кэшируемая, свободная памяти считаются доступными, так как такая память может быть незамедлительно освобождена процессу, который запросил больше памяти.
Таким образом, неактивная память помечается и как занятая, так и как свободная одновременно. Из-за этого элемент данных *vm.memory.size[used]* задумывался исключительно в информационных целях, тогда как элемент данных *vm.memory.size[available]* предназначен для использования в триггерах.

Смотрите также

1. Дополнительная информация о вычислении памяти на разных ОС [en]

### 2. Пассивные и активные проверки агента

Обзор

Этот раздел подробно описывает пассивные и активные проверки, которые выполняются Zabbix агентом и Zabbix агентом 2.

Для взаимодействия с агентами Zabbix использует коммуникационный протокол на основе JSON.

Протоколы Zabbix агента и Zabbix агента 2 были унифицированы, начиная с версии Zabbix 7.0. Отличие между запросами/ответами Zabbix агента и Zabbix агента 2 выражается значением тега «variant».

Пассивные проверки

Пассивная проверка — это простой запрос данных. Сервер Zabbix или прокси запрашивает некоторые данные (например, загрузку CPU), а агент Zabbix отправляет результат обратно на сервер.

Пассивные проверки выполняются асинхронно — не требуется получать ответ на один запрос до начала других проверок. Разрешение DNS также выполняется асинхронно.

Пуллер агента будет пытаться подключиться ко всем адресам, возвращенным DNS-поиском. Это гарантирует, что если один IP-адрес недоступен, пуллер попробует следующий доступный адрес, повышая вероятность успешного подключения. Это улучшение применяется как к серверу Zabbix, так и к прокси.

Максимальная параллельность асинхронных проверок составляет 1000 (определяется параметром MaxConcurrentChecksPerPoller).

Количество асинхронных пуллеров агента определяется параметром StartAgentPollers.

**Запрос сервера**

Для определения длины заголовка и данных см. сведения о протоколе.

```
{
  "request": "passive checks",
  "data": [
    {
      "key": "agent.version",
      "timeout": 3
    }
  ]
}
```

| Field | | Type | Mandatory | Value |
|---|---|---|---|---|
| request | | *string* | yes | "passive checks" |
| data | | *array of object* | yes | Элемент данных пассивной проверки. |
| | key | *string* | yes | Ключ элемента данных с раскрытыми макросами. |
| | timeout | *number* | yes | Тайм-аут связи. |

**Ответ агента**

```
{
  "version": "7.4.0",
  "variant": 2,
  "data": [
    {
      "value": "7.4.0"
    }
  ]
}
```

| Field | | Type | Mandatory | Value |
|---|---|---|---|---|
| version | | *string* | yes | Номер версии агента. |
| variant | | *number* | yes | Вариант агента (*1* - Zabbix agent, *2* - Zabbix agent 2). |
| data | | *array of object* | yes | Содержит результат проверки. |
| | value | *string* | no | Значение элемента данных, если проверка выполнена успешно. |
| | error | *string* | no | Сообщение об ошибке, если проверка не выполнена успешно. |

Например, для поддерживаемых элементов данных:

1. Сервер открывает TCP-соединение
2. Сервер отправляет **<HEADER><DATALEN>{”request”:”passive checks”,”data”:[{”key”:”agent.ping”,”timeout”:3}]}**
3. Агент читает запрос и отвечает **<HEADER><DATALEN>{”version”:”7.4.0”,”variant”:2,”data”:[{”value”:1}]}**
4. Сервер обрабатывает данные, чтобы получить значение, в нашем случае — ’1’
5. TCP-соединение закрывается

Для неподдерживаемых элементов данных:

1. Сервер открывает TCP-соединение
2. Сервер отправляет **<HEADER><DATALEN>{”request”:”passive checks”,”data”:[{”key”:”vfs.fs.size[/nono]”,”timeout”:3**
3. Агент читает запрос и отвечает **<HEADER><DATALEN>{”version”:”7.4.0”,”variant”:2,”data”:[{”error”:”Unsupported item key.”}]}**
4. Сервер обрабатывает данные, изменяет состояние элемента данных на неподдерживаемое с указанным сообщением об ошибке
5. TCP-соединение закрывается

Переключение на старый протокол

Чтобы обеспечить совместимость сервера или прокси Zabbix с агентами версий ниже 7.2, использующими протокол в открытом виде, реализовано переключение на старый протокол.

Пассивные проверки выполняются с использованием JSON-протокола (7.0 и новее) после перезапуска или при изменении конфигурации интерфейса.
Если в ответ не получен допустимый JSON (агент отправил ”ZBX_NOTSUPPORTED”), Zabbix сохранит для интерфейса старый протокол и **повторит** проверку, отправив только ключ элемента данных.

Обратите внимание, что каждый час сервер/прокси Zabbix снова будет пытаться работать с новым протоколом со всеми интерфейсами, при необходимости переключаясь обратно на старый протокол.

Активные проверки

Активные проверки требуют более сложной обработки. Агент сначала должен получить с сервера/прокси список элементов данных и/или удалённых команд для независимой обработки.

Сервера/прокси для получения активных проверок перечислены в параметре «ServerActive» файла конфигурации агента. Частота запросов данных проверок настраивается параметром «RefreshActiveChecks» в этом же файле конфигурации. Однако, если обновление активных проверок завершится с ошибкой, запрос повторится через (жёстко заданные в коде) 60 секунд.

**Note:**
Начиная с версии Zabbix 6.4, агент (в активном режиме) больше не получает от сервера/прокси полную копию конфигурации раз в две минуты (по умолчанию). Вместо этого, чтобы уменьшить сетевой трафик и использование ресурсов, каждые 5 секунд (по умолчанию) выполняется инкрементная синхронизация конфигурации, после чего сервер/прокси предоставляет полную копию конфигурации, **только** если агент ещё не получил её либо что-то изменилось в конфигурации узла сети, глобальных макросах или глобальных регулярных выражениях.

Затем агент периодически отправляет новые значения на сервер(ы). Если агент получил какие-либо удалённые команды для выполнения, также будет отправлен результат выполнения. Обратите внимание, что удалённое выполнение команд на активном агенте поддерживается, начиная с версии Zabbix агента 7.0.

**Note:**
Если агент находится за брандмауэром, вы можете рассмотреть возможность использования исключительно Активных проверок, т.к. в этом случае вам не понадобится менять настройки брандмауэра для разрешения начальных входящих соединений.

Получение списка элементов данных

**Запрос агента**

Запрос active checks используется для получения активных проверок, которые должны обрабатываться агентом. Этот запрос отправляется агентом при запуске, а затем с интервалами RefreshActiveChecks.

```
{
  "request": "active checks",
  "host": "Zabbix server",
  "host_metadata": "mysql,nginx",
  "interface": "zabbix.server.lan",
  "ip": "159.168.1.1",
  "port": 12050,
  "version": "7.4.0",
  "variant": 2,
  "config_revision": 1,
  "session": "e3dcbd9ace2c9694e1d7bbd030eeef6e"
}
```

| Field | Type | Mandatory | Value |
|---|---|---|---|
| request | *string* | yes | `active checks` |
| host | *string* | yes | Имя узла сети. |
| host_metadata | *string* | no | Значение конфигурационного параметра HostMetadata или метрики HostMetadataItem. |
| interface | *string* | no | Значение конфигурационного параметра HostInterface или метрики HostInterfaceItem. |
| ip | *string* | no | Первый IP-адрес параметра ListenIP, если он задан. |
| port | *number* | no | Значение параметра ListenPort, если он задан и не совпадает с портом прослушивания агента по умолчанию. |
| version | *string* | yes | Номер версии агента. |
| variant | *number* | yes | Вариант агента (*1* - Zabbix agent, *2* - Zabbix agent 2). |
| config_revision | *number* | no | Идентификатор конфигурации для инкрементальной синхронизации конфигурации. |
| session | *string* | no | Идентификатор сеанса для инкрементальной синхронизации конфигурации. |

**Ответ сервера**

Ответ active checks отправляется сервером агенту после обработки запроса active checks.

```
{
  "response": "success",
  "config_revision": 2,
  "data": [
    {
      "key": "system.uptime",
      "itemid": 1234,
      "delay": "10s",
      "lastlogsize": 0,
      "mtime": 0
    },
    {
      "key": "agent.version",
      "itemid": 5678,
      "delay": "10m",
      "lastlogsize": 0,
      "mtime": 0,
      "timeout": "30s"
    }
  ],
  "commands": [
    {
      "command": "df -h --output=source,size / | awk 'NR>1 {print $2}'",
      "id": 1324,
      "wait": 1
    }
  ]
}
```

| Field | Type | Mandatory | Value |
|---|---|---|---|
| response | *string* | yes | `success` \| `failed` |
| info | *string* | no | Информация об ошибке в случае сбоя. |
| data | *array of objects* | no | Элементы данных активных проверок. Не указывается, если конфигурация узла сети не изменилась. |
| key | *string* | no | Ключ элемента данных с раскрытыми макросами. |
| itemid | *number* | no | Идентификатор элемента данных. |
| delay | *string* | no | Интервал обновления элемента данных.<br>Гибкие интервалы/интервалы по расписанию поддерживаются как Zabbix agent, так и Zabbix agent 2 начиная с Zabbix 7.0. |
| lastlogsize | *number* | no | lastlogsize элемента данных. |
| mtime | *number* | no | mtime элемента данных. |
| timeout | *string* | no | Тайм-аут элемента данных. |
| refresh_unsupported | *number* | no | Интервал обновления неподдерживаемого элемента данных. |
| regexp | *array of objects* | no | Глобальные регулярные выражения. |
| name | *string* | no | Имя глобального регулярного выражения. |
| expression | *string* | no | Глобальное регулярное выражение. |
| expression_type | *number* | no | Тип глобального регулярного выражения. |
| exp_delimiter | *string* | no | Разделитель глобального регулярного выражения. |
| case_sensitive | *number* | no | Параметр чувствительности глобального регулярного выражения к регистру. |
| commands | *array of objects* | no | Удаленные команды для выполнения. Включаются, если выполнение удаленной команды было инициировано операцией действия или ручным выполнением скрипта. Обратите внимание, что выполнение удаленных команд на активном агенте поддерживается начиная с Zabbix agent 7.0. Более старые активные агенты будут игнорировать любые удаленные команды, включенные в ответ сервера active checks. |

| Field | Type | Mandatory | Value |
|---|---|---|---|
| command | *string* | no | Удаленная команда. |
| id | *number* | no | Идентификатор удаленной команды. |
| wait | *number* | no | Режим выполнения удаленной команды (”0” (nowait) для команд из операций действия; ”1” (wait) для команд из ручного выполнения скрипта). |
| timeout | *number* | no | Тайм-аут выполнения удаленной команды в конфигурации сервер/прокси. |
| config_revision | *number* | no | Идентификатор конфигурации для инкрементальной синхронизации конфигурации. Не указывается, если конфигурация узла сети не изменилась. Увеличивается, если конфигурация узла сети изменена. |

Сервер должен ответить success.

Например:

1. Агент открывает TCP-соединение
2. Агент запрашивает список проверок
3. Сервер отвечает списком элементов данных и удаленных команд для выполнения
4. Агент разбирает ответ
5. TCP-соединение закрывается
6. Агент начинает периодический сбор данных и выполняет удаленные команды (поддерживается начиная с Zabbix agent 7.0)

**Attention:**
Обратите внимание, что конфиденциальные данные конфигурации могут стать доступны сторонам, имеющим доступ к порту trapper сервера Zabbix, при использовании активной проверки. Это возможно, поскольку любой может выдать себя за активный агент и запросить данные конфигурации элемента данных; аутентификация не выполняется, если не используются параметры шифрования.

Отправка собранных данных

**Агент отправляет**

Запрос данных агента содержит собранные значения элементов данных и значения выполненных удаленных команд (если таковые имеются).

```
{
  "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": "7.4.0",
```

```
  "variant": 2
}
```

| Field | Type | Mandatory | Value |
|---|---|---|---|
| request | *string* | yes | `agent data` |
| data | *array of objects* | yes | Значения элементов данных. |
| id | *number* | yes | Идентификатор значения (инкрементный счетчик, используемый для проверки дублирующихся значений в случае проблем с сетью). |
| itemid | *number* | yes | Идентификатор элемента данных. |
| value | *string* | no | Значение элемента данных. |
| lastlogsize | *number* | no | lastlogsize элемента данных. |
| mtime | *number* | no | mtime элемента данных. |
| state | *number* | no | Состояние элемента данных. |
| source | *string* | no | Источник журнала событий значения. |
| eventid | *number* | no | eventid журнала событий значения. |
| severity | *number* | no | severity журнала событий значения. |
| timestamp | *number* | no | timestamp журнала событий значения. |
| clock | *number* | yes | Метка времени значения (секунды с начала эпохи). |
| ns | *number* | yes | Метка времени значения в наносекундах. |
| commands | *array of objects* | no | Результат выполнения удаленных команд. Обратите внимание, что выполнение удаленных команд на активном агенте поддерживается начиная с Zabbix agent 7.0. Более старые активные агенты будут игнорировать любые удаленные команды, включенные в ответ сервера на активные проверки. |
| id | *number* | no | Идентификатор удаленной команды. |
| value | *string* | no | Результат выполнения удаленной команды, если выполнение прошло успешно. |
| error | *string* | no | Сообщение об ошибке выполнения удаленной команды, если выполнение завершилось неудачей. |
| session | *string* | yes | Уникальный идентификатор сеанса, создаваемый каждый раз при запуске агента. |
| host | *string* | yes | Имя узла сети. |
| version | *string* | yes | Номер версии агента. |
| variant | *number* | yes | Вариант агента (*1* - Zabbix agent, *2* - Zabbix agent 2). |

Каждому значению назначается виртуальный ID. ID значения — это простой возрастающий счетчик, уникальный в пределах одного сеанса данных (идентифицируемого по токену сеанса). Этот ID используется для отбрасывания дублирующихся значений, которые могут передаваться в условиях плохого качества связи.

**Ответ сервера**

Ответ на данные агента отправляется сервером обратно агенту после обработки запроса данных агента.

```
{
  "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 | Результаты обработки элементов данных. |

**Attention:**
Если отправка некоторых значений на сервере завершается неудачей (например, потому что узел сети или элемент данных был отключен или удален), агент не будет повторять отправку этих значений.

Например:

1. Агент открывает TCP-соединение

2. Агент отправляет список значений
3. Сервер обрабатывает данные и отправляет статус обратно
4. TCP-соединение закрывается

**Attention:**
Сообщение об ошибке будет обрезано до 2048 символов на стороне сервера.

Сообщение heartbeat

**Отправка агентом**

Сообщение heartbeat отправляется активным агентом на сервер/прокси Zabbix каждые HeartbeatFrequency секунд (настраивается в файле конфигурации Zabbix agent/ agent 2).

Оно используется для мониторинга доступности активных проверок.

```
{
  "request": "active check heartbeat",
  "host": "Zabbix server",
  "heartbeat_freq": 60,
  "version": "7.4.0",
  "variant": 2
}
```

| Поле | Тип | Обязательное | Значение |
|---|---|---|---|
| request | *string* | yes | `active check heartbeat` |
| host | *string* | yes | Имя узла сети. |
| heartbeat_freq | *number* | yes | Частота heartbeat агента (параметр конфигурации HeartbeatFrequency). |
| version | *string* | yes | Номер версии агента. |
| variant | *number* | yes | Вариант агента (*1* - Zabbix agent, *2* - Zabbix agent 2). |

**Ответ перенаправления**

Когда узел сети был переназначен, сервер может указать агенту перенаправить его heartbeat (и последующие активные проверки) на другой экземпляр прокси или сервера.

```
{
  "response": "failed",
  "redirect": {
    "revision": 2,
    "address": "192.0.2.1:10055"
  }
}
```

| Поле | | Тип | Обязательное | Значение |
|---|---|---|---|---|
| response | | *string* | yes | `success` \| `failed` |
| redirect | | *object* | yes | Инструкции по перенаправлению. |
| | revision | *number* | yes | Идентификатор ревизии конфигурации. |
| | address | *string* | yes | Адрес целевого сервера/прокси. |

Более старый XML протокол

**Note:**
Zabbix воспринимает до 16 MB XML данных, закодированных в Base64, но одиночное декодированное значение не должно быть длиннее 64 КБ, в противном случае значение будет обрезано до 64 КБ в процессе декодирования.

### 3. Минимальный уровень прав для элементов данных агента Windows

Обзор

Когда системы мониторинга используют агента, хорошей практикой будет получение метрик с хоста, на котором установлен этот агент. Чтобы использовать принцип минимальных привилегий, необходимо определить метрики, которые собираются агентом.

Таблица в этом документе позволит вам выбрать минимальные права, которые будут гарантировать корректную работу Zabbix агента.

Если для работы агента выбран пользователь, отличный от «LocalSystem», то для работы агента в качестве службы Windows новый пользователь должен иметь права «Вход в качестве службы» из «Групповая политика→Назначение прав пользователя» и право создавать, записывать и удалять файл журнала Zabbix агента. Пользователь Active Directory должен быть добавлен в группу «Performance Monitor Users».

**Note:**
При работе с правами агента основываясь на группе «Минимально технически приемлемо», требуется предварительное предоставление прав на объекты мониторинга.

Общие элементы данных агента, поддерживаемые в Windows

| Ключ элемента данных | Группа пользователей | |
|---|---|---|
| | Рекомендуется | Минимально технически приемлемо (функциональность ограничена) |
| agent.hostname | Гости | Гости |
| agent.ping | Гости | Гости |
| agent.variant | Гости | Гости |
| agent.version | Гости | Гости |
| log | Администраторы | Гости |
| log.count | Администраторы | Гости |
| logrt | Администраторы | Гости |
| logrt.count | Администраторы | Гости |
| net.dns | Гости | Гости |
| net.dns.perf | Гости | Гости |
| net.dns.record | Гости | Гости |
| net.if.discovery | Гости | Гости |
| net.if.in | Гости | Гости |
| net.if.out | Гости | Гости |
| net.if.total | Гости | Гости |
| net.tcp.listen | Гости | Гости |
| net.tcp.port | Гости | Гости |
| net.tcp.service | Гости | Гости |
| net.tcp.service.perf | Гости | Гости |
| net.udp.service | Гости | Гости |
| net.udp.service.perf | Гости | Гости |
| proc.num | Администраторы | Гости |
| system.cpu.discovery | Пользователи монитора производительности | Пользователи монитора производительност |
| system.cpu.load | Пользователи монитора производительности | Пользователи монитора производительност |
| system.cpu.num | Гости | Гости |
| system.cpu.util | Пользователи монитора производительности | Пользователи монитора производительност |
| system.hostname | Гости | Гости |
| system.localtime | Гости | Гости |
| system.run | Администраторы | Гости |

| Ключ элемента данных | Группа пользователей | |
|---|---|---|
| system.sw.arch | Гости | Гости |
| system.swap.size | Гости | Гости |
| system.uname | Гости | Гости |
| system.uptime | Пользователи монитора производительности | Пользователи монитора производительност |
| vfs.dir.count | Администраторы | Гости |
| vfs.dir.get | Администраторы | Гости |
| vfs.dir.size | Администраторы | Гости |
| vfs.file.cksum | Администраторы | Гости |
| vfs.file.contents | Администраторы | Гости |
| vfs.file.exists | Администраторы | Гости |
| vfs.file.md5sum | Администраторы | Гости |
| vfs.file.regexp | Администраторы | Гости |
| vfs.file.regmatch | Администраторы | Гости |
| vfs.file.size | Администраторы | Гости |
| vfs.file.time | Администраторы | Гости |
| vfs.fs.discovery | Администраторы | Гости |
| vfs.fs.size | Администраторы | Гости |
| vfs.fs.get | Администраторы | Гости |
| vm.memory.size | Гости | Гости |
| web.page.get | Гости | Гости |
| web.page.perf | Гости | Гости |
| web.page.regexp | Гости | Гости |
| zabbix.stats | Гости | Гости |

Ключи элементов данных, специфичные для Windows

| Ключ элемента данных | Группа пользователей | |
|---|---|---|
| | Рекомендуется | Минимально технически приемлемо (ограниченные возможности) |
| eventlog | Читатели журнала событий | Гости |
| net.if.list | Гости | Гости |
| perf_counter | Пользователи монитора производительности | Пользователи монитора производительност |
| proc_info | Администраторы | Гости |
| service.discovery | Гости | Гости |
| service.info | Гости | Гости |
| services | Гости | Гости |
| wmi.get | Администраторы | Гости |
| vm.vmemory.size | Гости | Гости |

#### 4. Кодировка получаемых значений

Zabbix сервер ожидает, что каждое получаемое текстовое значение будет в кодировке UTF8. Это относится к любому типу проверок: Zabbix агент, SSH, Telnet и т.д.

Различные системы/устройства и проверки могут возвращать в значениях non-ASCII символы. Для таких случаев, почти все доступные в Zabbix ключи имеют дополнительный параметр ключа элемента данных — **<кодировка>**. Этот параметр ключа необязателен, но он должен быть указан, если получаемое значение не в кодировке UTF8 и содержит non-ASCII символы. В противном случае результат может быть неожиданным и непредсказуемым.

Описание поведения различных баз данных в этих случаях представлено ниже.

MySQL

Если значение содержит non-ASCII символ в не UTF8 кодировке — этот символ и следующий за ним символ будут отброшены при записи этого значения базой данных. Никакие предупреждающие сообщения не записываются в *zabbix_server.log*.<br> Такое поведение соответствует по крайней мере MySQL версии 5.1.61.

PostgreSQL

Если значение содержит non-ASCII символ в не UTF8 кодировке — это приведёт к ошибке в SQL запросе (PGRES_FATAL_ERROR:ERROR invalid byte sequence for encoding), и данные не будут записаны. Соответствующее сообщение с предупреждением будет записано в *zabbix_server.log*.<br> Такое поведение соответствует по крайней мере PostgreSQL версии 9.1.3.

#### 5 Поддержка больших файлов

Поддержка больших файлов, часто сокращаемая до LFS, — это термин, обозначающий возможность работать с файлами размером более 2 ГБ в 32-разрядных операционных системах. Поддержка больших файлов влияет как минимум на мониторинг лог-файлов и все элементы данных vfs.file.*. Поддержка больших файлов зависит от возможностей системы на момент компиляции Zabbix, но полностью отключена в 32-разрядной Solaris из-за несовместимости с procfs и swapctl.

#### 6. Датчик

Обзор

Каждый контроллер датчика имеет свой собственный каталог в дереве sysfs `/sys/devices`. Чтобы найти все контроллеры датчиков, проще перейти по символическим ссылкам на устройства из `/sys/class/hwmon/hwmon*`, где * — это конкретный номер (0,1,2,...).

Показания датчиков находятся либо в папке `/sys/class/hwmon/hwmon*/` для виртуальных устройств, либо в папке `/sys/class/hwmon/hwmon*/device` для невиртуальных устройств. Файл с именем `name`, расположенный в папках `hwmon*` или `hwmon*/device`, содержит имя чипа, которое соответствует имени драйвера ядра, используемого контроллером датчика.

В каждом файле имеется только одно значение показания датчика. Общая схема именования файлов, которые содержат показания датчиков, внутри любой из вышеупомянутых папок: <тип><номер>_<элемент>, где

- **тип** — для чипов датчиков «in» (напряжение), «temp» (температура), «fan» (скорость вентилятора) и т.д.
- **элемент** — «input» (измеренное значение), «max» (верхний порог), «min» (нижний порог) и т.д.
- **номер** — всегда используется для экземпляров, которые могут присутствовать более одного раза (обычно начинается с 1, за исключением датчиков напряжения, которые начинаются с 0). Если файлы не относятся к конкретным экземплярам, они имеют простое наименование без номера.

Информацию о датчиках, доступных на хосте, можно получить с помощью инструментов **sensors-detect** и **sensors** (пакет lm-sensors [en]). **Sensors-detect** помогает определить, какие модули необходимы для доступных датчиков. Когда модули загружены, можно использовать утилиту **sensors**, чтобы отобразить показания со всех контроллеров. Маркировка показаний датчиков, используемая данной утилитой, может отличаться от общепринятой схемы именования (< >< >_< > ):

- если имеется файл с именем < >< >_ , то вместо имени < >< >_< > будет использоваться метка внутри этого файла;
- если файл < >< >_ отсутствует, то для подстановки имени утилита выполняет поиск внутри `/etc/sensors.conf` (также может быть `/etc/sensors3.conf` или другой).

Эта маркировка позволяет пользователю определить тип используемого оборудования. Если нет ни файла < >< >_ , ни метки внутри файла конфигурации, то тип оборудования может быть определён по атрибуту имени (`hwmon*/device/name`). Реальные имена датчиков, которые воспринимает `zabbix_agent`, можно получить, запустив утилиту **sensors** с параметром «`-u`» (`sensors -u`).

В утилите **sensors** доступные датчики разделены по типу шины (ISA adapter, PCI adapter, SPI adapter, Virtual device, ACPI interface, HID adapter).

Получение идентификаторов датчиков

Метки датчиков не всегда могут быть использованы в том виде, как они выводятся утилитой `sensors`, т.к. у разных производителей контроллеров именование меток может отличаться. Например, вывод утилиты `sensors` может содержать следующие строки:

```
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)
```

Из них только одно отображаемое имя напрямую соответствует внутреннему идентификатору:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in0]
2.240000
```

Попытки использовать остальные отображаемые метки (такие как Vcore или +12V) работать не будут.

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,Vcore]
ZBX_NOTSUPPORTED
```

Чтобы найти внутренний идентификатор датчика, который воспринимает Zabbix, запустите `sensors -u`. В полученном выводе можно увидеть следующее:

```
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
...
```

Таким образом, метка Vcore соответствует идентификатору in1, а +12V соответствует in4. Согласно спецификации [en] hwmon sysfs ядра, это напряжения на выводах микросхемы и, в общем случае, может понадобиться масштабирование.

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in1]
1.301000
```

В Zabbix можно получать не только показания напряжения (in), но также тока (curr), температуры (temp) и скорости вращения вентилятора (fan).

Совместимость

Устаревшие дистрибутивы

Показания датчиков извлекаются из папки /proc/sys/dev/sensors:

- **устройство** — имя устройства (если <режим> используется, это регулярное выражение)
- **датчик** — имя датчика (если используется <режим>, это регулярное выражение)
- **режим** — возможные значения: avg, max, min (если этот параметр не указан, устройство и сенсор обрабатываются буквально)

Пример ключа: sensor[w83781d-i2c-0-2d,temp1].

Текущие дистрибутивы

Показания датчиков извлекаются из папки /sys/class/hwmon:

- **устройство** — имя устройства (не регулярное выражение). Имя устройства может быть фактическим именем устройства (например, 0000:00:18.3) или именем, полученным при использовании утилиты sensors (например, k8temp-pci-00c3). Выбор того, какое из этих имён использовать, оставлен за пользователем.
- **датчик** — имя датчика (не регулярное выражение).
- **режим** — возможные значения: avg, max, min (если этот параметр не указан, устройство и сенсор обрабатываются буквально).

Примеры ключей:

sensor[k8temp-pci-00c3,temp,max] или sensor[0000:00:18.3,temp1].

sensor[smsc47b397-isa-0880,in,avg] или sensor[smsc47b397.2176,in1].

### 7. Заметки о параметре «тип памяти» в элементах данных proc.mem

Обзор

Параметр **тип памяти** поддерживается на платформах Linux, AIX, FreeBSD и Solaris.

Три общих значения параметра «тип памяти» поддерживаются на всех этих платформах: `pmem`, `rss` и `vsize`. Кроме того, для некоторых платформ поддерживаются специфичные для этих платформ значения «типа памяти».

AIX

Значения, поддерживаемые для параметра «тип памяти» на AIX, смотри в таблице.

| Поддерживаемое значение | Описание | Источник в структуре procentry64 | Пытается быть совместимым с |
|---|---|---|---|
| vsize [1] | Размер виртуальной памяти | pi_size | |
| pmem | Процент физической памяти | pi_prm | ps -o pmem |
| rss | Размер резидентного набора | pi_trss + pi_drss | ps -o rssize |
| size | Размер процесса (код + данные) | pi_dvm | ”ps gvw” колонка SIZE |
| dsize | Размер данных | pi_dsize | |
| tsize | Размер текста (кода) | pi_tsize | ”ps gvw” колонка TSIZ |
| sdsize | Размер данных из разделяемой библиотеки | pi_sdsize | |
| drss | Размер резидентного набора данных | pi_drss | |
| trss | Размер резидентного набора текста | pi_trss | |

Примечания для AIX:

1. При выборе параметров для ключа элемента данных proc.mem[] на AIX, попытайтесь указать узкие критерии выбора процесса. В противном случае есть риск того, что в результат proc.mem[] окажутся сосчитаны нежелательные процессы.

Пример:

```
$ zabbix_agentd -t proc.mem[,,,          ,rss]
proc.mem[,,,          ,rss]          [u|2879488]
```

Данный пример иллюстрирует, как указание только параметра командной строки (в виде регулярного выражения) приводит к тому, что Zabbix агент считает и себя тоже - вероятно, это не то, чего хотелось.

2. Не используйте «ps -ef» для просмотра процессов — он показывает только процессы, не относящиеся к ядру. Используйте «ps -Af», чтобы увидеть все процессы, которые будет видеть агент Zabbix.
3. Давайте рассмотрим пример «topasrec», как proc.mem[] агента Zabbix выбирает процессы.

```
$ ps -Af | grep topasrec
root 1074798410Mar 16-0:00 /usr/bin/topasrec-L -s 300 -R 1 -r 6 -o /var/perf daily/ -ypersistent=1 -O type
```

proc.mem[] имеет аргументы:

```
proc.mem[< >,<      >,<   >,<cmdline>,<      >]
```

Первый критерий — имя процесса (аргумент <имя>). В нашем примере агент Zabbix увидит его как «topasrec». Для соответствия вам нужно либо указать «topasrec», либо оставить его пустым. Второй критерий — имя пользователя (аргумент <пользователь>). Для соответствия вам нужно либо указать «root», либо оставить его пустым. Третий критерий, используемый при выборе процесса, — это аргумент <cmdline>. Агент Zabbix увидит его значение как «/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». Для сопоставления вам нужно либо указать регулярное выражение, соответствующее этой строке, либо оставить его пустым.

Аргументы <режим> и <тип памяти> применяются после использования трёх критериев, упомянутых выше.

FreeBSD

Значения, поддерживаемые для параметра «тип памяти» на FreeBSD, смотрите в таблице.

| Поддерживаемое значение | Описание | Источник в структуре kinfo_proc | Пытается быть совместимым с |
|---|---|---|---|
| vsize | Размер виртуальной памяти | kp_eproc.e_vm.vm_map_size или ki_size | ps -o vsz |
| pmem | Процент физической памяти | вычисляется из rss | ps -o pmem |
| rss | Размер резидентного набора | kp_eproc.e_vm.vm_rssize или ki_rssize | ps -o rss |
| size [1] | Размер процесса (код + данные + стек) | tsize + dsize + ssize | |
| tsize | Размер текста (кода) | kp_eproc.e_vm.vm_tsize или ki_tsize | ps -o tsiz |
| dsize | Размер данных | kp_eproc.e_vm.vm_dsize или ki_dsize | ps -o dsiz |
| ssize | Размер стека | kp_eproc.e_vm.vm_ssize или ki_ssize | ps -o ssiz |

Linux

Значения, поддерживаемые для параметра «тип памяти» на Linux, смотрите в таблице.

| Поддерживаемое значение | Описание | Источник в файле /proc/<pid>/status |
|---|---|---|
| vsize [1] | Размер виртуальной памяти | VmSize |
| pmem | Процент физической памяти | (VmRSS/total_memory) * 100 |
| rss | Размер резидентного набора | VmRSS |
| data | Размер сегмента данных | VmData |
| exe | Размер сегмента кода | VmExe |
| hwm | Пиковый размер резидентного набора | VmHWM |
| lck | Размер заблокированной памяти | VmLck |
| lib | Размер разделяемых библиотек | VmLib |
| peak | Пиковый размер виртуальной памяти | VmPeak |
| pin | Размер закреплённых страниц | VmPin |
| pte | Размер записей таблицы страниц | VmPTE |
| size | Размер сегментов кода + данных + стека процесса | VmExe + VmData + VmStk |
| stk | Размер сегмента стека | VmStk |
| swap | Размер используемого места в разделе подкачки | VmSwap |

Примечания для Linux:

1. Не все значения параметра «тип памяти» поддерживаются старыми версиями ядер Linux. Например, ядра Linux 2.4 не поддерживают значения `hwm`, `pin`, `peak`, `pte` и `swap`.
2. Мы заметили, что процесс само-диагностики активных проверок Zabbix агента с `proc.mem[...,...,...,...,data]` отображает значение, которое на 4 КБ больше, чем отдаёт строка `VmData` в файле агента /proc/<pid>/status. Во время измерения себя сегмент данных агента увеличивается на 4 КБ и затем возвращается к предыдущему размеру.

Solaris

Значения, поддерживаемые для параметра «тип памяти» на Solaris, смотрите в таблице.

| Поддерживаемое значение | Описание | Источник в структуре psinfo | Пытается быть совместимым с |
|---|---|---|---|
| vsize [1] | Размер образа процесса | pr_size | ps -o vsz |
| pmem | Процент физической памяти | pr_pctmem | ps -o pmem |
| rss | Размер резидентного набора<br>Он может быть недооценён - см. описание rss в «man ps». | pr_rssize | ps -o rss |

Примечания

[1] Значение по умолчанию.

#### 8. Заметки по выбору процессов в элементах данных proc.mem и proc.num

Процессы, изменяющие свою командную строку

Некоторые программы используют изменение своей командной строки как метод отображения своей текущей активности. Пользователь может увидеть эту активность, запуская команды ps и top. Примеры таких программ: *PostgreSQL*, *Sendmail*, *Zabbix*.

Давайте рассмотрим пример из Linux. Предположим, что мы хотим наблюдать количество процессов Zabbix агента.

Команда ps отобразит интересующие процессы следующим образом:

```
$ 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]
...
```

Выбор процессов по имени и пользователю выполняет задачу:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd,zabbix]'
6
```

Теперь давайте переименуем исполняемый файл zabbix_agentd в zabbix_agentd_30 и перезапустим его.

Теперь ps отображает:

```
$ 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]
...
```

Теперь выбор процессов по имени и пользователю выдаёт неверный результат:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd_30,zabbix]'
1
```

Почему простое переименование исполняемого файла на более длинное имя приводит к совершенно другому результату?

Zabbix agent начинает с проверки имени процесса. Открывается файл /proc/<pid>/status и проверяется строка Name. В нашем случае строки Name это:

```
$ 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
```

Имя процесса в файле status обрезается до 15 символов.

Аналогичный результат можно увидеть при помощи команды 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
 ...
```

Очевидно, что этот вывод не идентичен нашему значению zabbix_agentd_30 в параметре name ключа proc.num[]. Будучи не в состоянии найти совпадение по имени процесса в файле status, Zabbix агент обращается к файлу /proc/<pid>/cmdline.

То, как агент просматривает файл «cmdline», может быть проиллюстрировано при помощи выполнения команды:

```
$ 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
```

В нашем случае файлы /proc/<pid>/cmdline содержат невидимые, непечатаемые нулевые байты, которые используются для завершения строки в языке C. В этом примере нулевые байты отображаются как «<NUL>».

Zabbix агент проверяет «cmdline» основного процесса и извлекает zabbix_agentd_30, которое соответствует значению zabbix_agentd_30 нашего параметра name. Таким образом, основной процесс учитывается элементом данных proc.num[zabbix_agentd_30,zabbix].

При проверке следующего процесса агент из файла cmdline получает zabbix_agentd_30: collector [idle 1 sec], и это не соответствует нашему значению zabbix_agentd_30 в параметре name. Таким образом, будет сосчитан только основной процесс, который не меняет свою командную строку. Остальные процессы агента модифицируют свои командные строки и игнорируются.

Этот пример показывает, что в данном случае параметр name нельзя использовать в proc.mem[] и proc.num[] для выбора процессов.

**Note:**
Для элемента данных proc.get[], когда агент Zabbix проверяет «cmdline» на предмет имени процесса, он будет использовать только часть имени, начиная с последнего слеша и до первого пробела или знака двоеточия. Имя процесса, полученное из файла cmdline, будет использоваться только в том случае, если его начало полностью совпадает с сокращённым именем процесса в файле status. Алгоритм одинаков как для имени процесса в фильтре, так и для вывода JSON.

Использование параметра cmdline с надлежащим регулярным выражением даст правильный результат:

```
$ zabbix_get -s localhost -k 'proc.num[,zabbix,,zabbix_agentd_30[ :]]'
6
```

Будьте осторожны при использовании элементов данных proc.get[], proc.mem[] и proc.num[] при наблюдении за программами, которые модифицируют свои командные строки.

Перед тем как поместить параметры name и cmdline в элементы данных proc.get[], proc.mem[] и proc.num[], вы, возможно, захотите протестировать эти параметры, используя элемент данных proc.num[] и команду ps.

Потоки ядра Linux

Нельзя выбрать потоки при помощи параметров cmdline в элементах данных proc.get[], proc.mem[] и proc.num[]

Давайте возьмем в качестве примера один из потоков ядра:

```
$ ps -ef| grep kthreadd
root         2     0  0 09:33 ?        00:00:00 [kthreadd]
```

Его можно выбрать при помощи параметра :

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd,root]'
1
```

Но выбор при помощи параметра cmdline не работает:

```
$ zabbix_get -s localhost -k 'proc.num[,root,,kthreadd]'
0
```

Причина такого поведения кроется в том, что Zabbix агент берёт регулярное выражение, которое указано в параметре cmdline, и применяет его к содержимому относящегося к процессу файла /proc/<pid>/cmdline. В случае потоков ядра, их файлы /proc/<pid>/cmdline пустые. Поэтому параметр cmdline никогда не совпадёт.

Подсчёт потоков в элементах данных proc.mem[] и proc.num[]

Потоки ядра Linux подсчитываются элементом данных proc.num[], но не сообщают информацию о памяти в элементе данных proc.mem[] item. Например:

```
$ 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.
```

Но что случится, если имеется пользовательский процесс с таким же именем как и у потока ядра? Тогда это будет выглядеть примерно так:

```
$ 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[] подсчитал как поток ядра, так и пользовательский процесс. proc.mem[] сообщил информацию о памяти только по пользовательскому процессу и вычислил память потока ядра как будто оно было нулевым. Это отличается от случая выше, когда сообщалось ZBX_NOTSUPPORTED.

Будьте внимательны при использовании элементов данных proc.mem[] и proc.num[], если случится, что имя процесса совпадёт с одним из потоков.

Перед добавлением параметров в элементы данных proc.mem[] и proc.num[] вы, возможно, захотите протестировать эти параметры, используя элемент данных proc.num[] и команду ps.

#### 9. Подробности реализации проверок net.tcp.service и net.udp.service

В этом разделе подробно описана реализация проверок net.tcp.service и net.udp.service для различных сервисов, которые можно указать в параметре «сервис».

Параметры элемента данных net.tcp.service

**ftp**

Создаёт TCP-соединение и ожидает, что первые 4 символа ответа будут «220 », после чего посылает «QUIT\r\n». По умолчанию используется порт 21, если он не указан.

**http**

Создаёт TCP-соединение без ожидания и отсылки чего-либо. По умолчанию используется порт 80, если он не указан.

**https**

Использует (и работает только с) libcurl, не проверяет подлинность сертификата, не проверяет имя хоста в SSL сертификате, только извлекает заголовок ответа (запрос HEAD). По умолчанию используется порт 443, если он не указан.

**imap**

Создаёт TCP-соединение и ожидает, что первые 4 символа ответа будут «* OK», после чего посылает «a1 LOGOUT\r\n». По умолчанию используется порт 143, если он не указан.

**ldap**

Открывает соединение на сервер LDAP и выполняет операцию поиска (LDAP search) с фильтром, выставленным в (objectClass=*). Ожидает успешного извлечения первого атрибута у первого элемента. По умолчанию используется порт 389, если он не указан.

**nntp**

Создаёт TCP-соединение и ожидает, что первые 3 символа ответа будут «200» или «201», после чего посылает «QUIT\r\n». По умолчанию используется порт 119, если он не указан.

**pop**

Создаёт TCP-соединение и ожидает, что первые 3 символа ответа будут «+OK», после чего посылает «QUIT\r\n». По умолчанию используется порт 110, если он не указан.

**smtp**

Создаёт TCP-соединение и ожидает, что первые 3 символа ответа будут «220», после которых идёт пробел, конец строки или минус. Строки, содержащие минус, относятся к многострочному ответу, и ответ будет заново читаться до тех пор, пока не будет получена строка без минуса. После этого отсылается «QUIT\r\n». По умолчанию используется порт 25, если он не указан.

**ssh**

Создаёт TCP-соединение. Если соединение было установлено, обе стороны обмениваются строками идентификации (SSH-major.minor-XXXX), где major и minor являются версиями протокола, а XXXX - строкой. Zabbix проверяет, найдена ли соответствующая спецификации строка, и затем отсылает назад строку «SSH-major.minor-zabbix_agent\r\n» или «0\n» при несовпаднии. По умолчанию используется порт 22, если он не указан.

**tcp**

Создаёт TCP-соединение без ожидания и отсылки чего-либо. В отличие от остальных проверок, требует указания параметра «порт».

**telnet**

Создаёт TCP-соединение и ожидает приглашение на ввод имени пользователя («:» в конце строки). По умолчанию используется порт 23, если он не указан.

Параметры элемента данных net.udp.service

**ntp**

Отсылает по UDP пакет SNTP и проверяет ответ в соответствии с RFC 4330, раздел 5 [en]. По умолчанию используется порт 123, если он не указан.

### 10. Параметры proc.get

Обзор

Элемент данных **proc.get**[<имя>,<пользователь>,<cmdline>,<режим>] поддерживается в Linux, Windows, FreeBSD, OpenBSD и NetBSD.

Список параметров процесса, возвращаемых элементом данных, зависит от операционной системы и значения аргумента «режим».

Linux

В Linux для каждого режима возвращаются следующие параметры процесса:

| режим=process | режим=thread | режим=summary |
|---|---|---|
| pid: идентификатор процесса | pid: идентификатор процесса | name: имя процесса |
| ppid: идентификатор родительского процесса | ppid: идентификатор родительского процесса | processes: количество процессов |
| name: имя процесса | name: имя процесса | vsize: размер виртуальной памяти |
| cmdline: командная строка с аргументами | user: пользователь (реальный), от имени которого работает процесс | pmem: процент реальной памяти |
| user: пользователь (реальный), от имени которого работает процесс | group: группа (реальная), от имени которой работает процесс | rss: размер резидентного набора |
| group: группа (реальная), от имени которой работает процесс | uid: идентификатор пользователя | data: размер сегмента данных |

| режим=process | режим=thread | режим=summary |
|---|---|---|
| uid: идентификатор пользователя | gid: идентификатор группы, от имени которой работает процесс | exe: размер сегмента кода |
| gid: идентификатор группы, от имени которой работает процесс | tid: идентификатор потока (thread ID) | lib: размер разделяемых библиотек |
| vsize: размер виртуальной памяти | tname: имя потока | lck: размер заблокированной памяти (locked memory) |
| pmem: процент реальной памяти | cputime_user: общее время работы CPU в режиме пользователя (значение выражено в тиках часов — поделите на `getconf CLK_TCK` или `sysconf(_SC_CLK_TCK)`, чтобы получить секунды) | pin: размер закреплённых (pinned) страниц |
| rss: размер резидентного набора | cputime_system: общее время работы CPU в режиме системы (значение выражено в тиках часов — поделите на `getconf CLK_TCK` или `sysconf(_SC_CLK_TCK)`, чтобы получить секунды) | pte: размер записей таблицы страниц |
| data: размер сегмента данных | state: состояние потока | size: размер процесса (код + данные + сегменты стека) |
| exe: размер сегмента кода | ctx_switches: количество переключений контекста | stk: размер сегмента стека |
| hwm: пиковый размер резидентного набора | page_faults: количество ошибок страниц | swap: размер используемого пространства подкачки |
| lck: размер заблокированной памяти (locked memory) | | cputime_user: общее время работы CPU в режиме пользователя (значение выражено в тиках часов — поделите на `getconf CLK_TCK` или `sysconf(_SC_CLK_TCK)`, чтобы получить секунды) |
| lib: размер разделяемых библиотек | | cputime_system: общее время работы CPU в режиме системы (значение выражено в тиках часов — поделите на `getconf CLK_TCK` или `sysconf(_SC_CLK_TCK)`, чтобы получить секунды) |
| peak: пиковый размер виртуальной памяти | | ctx_switches: количество переключений контекста |
| pin: размер закреплённых (pinned) страниц | | threads: количество потоков |
| pte: размер записей таблицы страниц | | page_faults: количество ошибок страниц |
| size: размер процесса (код + данные + сегменты стека) | | pss: пропорциональный размер набора памяти |
| stk: размер сегмента стека | | |
| swap: размер используемого пространства подкачки | | |
| cputime_user: общее время работы CPU в режиме пользователя (значение выражено в тиках часов — поделите на `getconf CLK_TCK` или `sysconf(_SC_CLK_TCK)`, чтобы получить секунды) | | |
| cputime_system: общее время работы CPU в режиме системы (значение выражено в тиках часов — поделите на `getconf CLK_TCK` или `sysconf(_SC_CLK_TCK)`, чтобы получить секунды) | | |
| state: состояние процесса (прозрачно извлекается из procfs, длинная форма) | | |

| режим=process | режим=thread | режим=summary |
|---|---|---|
| ctx_switches: количество переключений контекста<br>threads: количество потоков<br>page_faults: количество ошибок страниц<br>pss: пропорциональный размер набора памяти | | |

ОС на основе BSD

На FreeBSD, OpenBSD и NetBSD для каждого режима возвращаются следующие параметры процесса:

| режим=process | режим=thread | режим=summary |
|---|---|---|
| pid: идентификатор процесса | pid: идентификатор процесса | name: имя процесса |
| ppid: идентификатор родительского процесса | ppid: идентификатор родительского процесса | processes: количество процессов |
| jid: идентификатор песочницы (jail, только FreeBSD) | jid: идентификатор песочницы (jail, только FreeBSD) | vsize: размер виртуальной памяти |
| jname: имя песочницы (jail, только FreeBSD) | jname: имя песочницы (jail, только FreeBSD) | pmem: процент реальной памяти (только FreeBSD) |
| name: имя процесса | name: имя процесса | rss: размер резидентного набора |
| cmdline: командная строка с аргументами | user: пользователь (реальный), от имени которого работает процесс | size: размер процесса (код + данные + стек) |
| user: пользователь (реальный), от имени которого работает процесс | группа (реальная), от имени которой работает процесс | tsize: размер текста (код) |
| group: группа (реальная), от имени которой работает процесс | uid: идентификатор пользователя | dsize: размер данных |
| uid: идентификатор пользователя | gid: идентификатор группы, от имени которой работает процесс | ssize: размер стека |
| gid: идентификатор группы, от имени которой работает процесс | tid: идентификатор потока (thread ID) | cputime_user: общее время работы CPU в режиме пользователя (в секундах) |
| vsize: размер виртуальной памяти | tname: имя потока | cputime_system: общее время работы CPU в секундах в режиме системы (в секундах) |
| pmem: процент реальной памяти (только FreeBSD) | cputime_user: общее время работы CPU в режиме пользователя (в секундах) | ctx_switches: количество переключений контекста |
| rss: размер резидентного набора | cputime_system: общее время работы CPU в режиме системы (в секундах) | threads: количество потоков (не поддерживается для NetBSD) |
| size: размер процесса (код + данные + стек) | state: состояние потока | stk: размер сегмента стека |
| tsize: размер текста (код) | ctx_switches: количество переключений контекста | page_faults: количество ошибок страниц |
| dsize: размер данных | io_read_op: количество раз, когда системе пришлось выполнить ввод | fds: количество файловых дескрипторов (только OpenBSD) |
| ssize: размер стека | io_write_op: количество раз, когда системе пришлось выполнить вывод | swap: размер используемого пространства подкачки |
| cputime_user: общее время работы CPU в режиме пользователя (в секундах) | | io_read_op: количество раз, когда системе пришлось выполнить ввод |
| cputime_system: общее время работы CPU в режиме системы (в секундах) | | io_write_op: количество раз, когда системе пришлось выполнить вывод |
| state: состояние процесса (disk sleep/running/sleeping/tracing stop/zombie/other) | | |
| ctx_switches: количество переключений контекста | | |

| режим=process | режим=thread | режим=summary |
|---|---|---|
| threads: количество потоков (не поддерживается для NetBSD)<br>page_faults: количество ошибок страниц<br>fds: количество файловых дескрипторов (только OpenBSD)<br>swap: размер используемого пространства подкачки<br>io_read_op: количество раз, когда системе пришлось выполнить ввод<br>io_write_op: количество раз, когда системе пришлось выполнить вывод | | |

Windows

В Windows для каждого режима возвращаются следующие параметры процесса:

| режим=process | режим=thread | режим=summary |
|---|---|---|
| pid: идентификатор процесса<br>ppid: идентификатор родительского процесса<br>name: имя процесса<br>user: пользователь, под которым выполняется процесс<br>sid: SID пользователя<br>vmsize: размер виртуальной памяти<br>wkset: размер рабочего набора процесса<br>cputime_user: общее время работы ЦП в режиме пользователя (в миллисекундах)<br>cputime_system: общее время работы ЦП в режиме системы (в миллисекундах)<br>threads: количество потоков<br>page_faults: количество ошибок страниц<br>handles: количество дескрипторов<br>io_read_b: прочитано байтов ввода-вывода<br>io_write_b: записано байтов ввода-вывода<br>io_read_op: операций чтения ввода-вывода<br>io_write_op: операций записи ввода-вывода<br>io_other_b: передано байтов ввода-вывода, кроме операций чтения и записи<br>io_other_op: операций ввода-вывода, кроме операций чтения и записи | pid: идентификатор процесса<br>ppid: идентификатор родительского процесса<br>name: имя процесса<br>user: пользователь, под которым выполняется процесс<br>sid: SID пользователя<br>tid: идентификатор потока | name: имя процесса<br>processes: количество процессов<br>vmsize: размер виртуальной памяти<br>wkset: размер рабочего набора процесса<br>cputime_user: общее время работы ЦП в режиме пользователя (в миллисекундах)<br>cputime_system: общее время работы ЦП в режиме системы (в миллисекундах)<br>threads: количество потоков<br>page_faults: количество ошибок страниц<br>handles: количество дескрипторов<br>io_read_b: прочитано байтов ввода-вывода<br>io_write_b: записано байтов ввода-вывода<br>io_read_op: операций чтения ввода-вывода<br>io_write_op: операций записи ввода-вывода<br>io_other_b: передано байтов ввода-вывода, кроме операций чтения и записи<br>io_other_op: операций ввода-вывода, кроме операций чтения и записи |

### 11. Настройки недостижимости/недоступности интерфейса узла сети

Обзор

Несколько параметров конфигурации определяют, каким должно быть поведение Zabbix сервера, когда агентские проверки (Zabbix, SNMP, IPMI, JMX) завершаются с ошибками и узел сети становится недостижимым.

Недостижимый интерфейс

Интерфейс узла сети считается недостижимым после ошибки при проверке (сетевая ошибка, превышение времени ожидания) посредством Zabbix, SNMP, IPMI или JMX агентов. Начиная с версии Zabbix 6.2.0, активные проверки Zabbix агента также влияют на доступность интерфейса. Когда активные проверки становятся недоступными, они вносят вклад в общее состояние доступности интерфейса агента.

С момента, когда интерфейс становится недоступным, **UnreachableDelay** определяет, как часто он будет проверяться повторно, используя один из элементов данных (включая правила LLD). Такие повторные проверки выполняются уже с помощью unreachable поллеров (или IPMI поллеров для IPMI проверок). По умолчанию интервал между последовательными проверками доступности составляет 15 секунд.

**Attention:**
Проверки, выполняемые асинхронными поллерами, не переносятся в unreachable поллеры.

В журнале Zabbix сервера недостижимость обозначается записями такого рода:

```
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
```

В сообщениях журнала указывается точный элемент данных, который не удалось выполнить, и его тип ( Zabbix агент).

**Note:**
Параметр *Timeout* также будет влиять на то, как быстро интерфейс будет проверен повторно в течение периода недостижимости. Если время ожидания составляет 20 секунд и UnreachableDelay составляет 30 секунд, то следующая проверка будет через 50 секунд после первой попытки.

Параметр **UnreachablePeriod** определяет общую длительность периода недостижимости. По умолчанию, UnreachablePeriod равен 45 секундам. Это значение должно быть в несколько раз больше, чем UnreachableDelay, чтобы гарантировать, что интерфейс будет перепроверен несколько раз, прежде чем будет помечен как недоступный.

Внутренний элемент данных, zabbix[host,active_agent,available], позволяет отслеживать доступность активных проверок в сценариях недостижимости.

Перевод интерфейса обратно в доступное состояние

По окончании периода недостижимости интерфейс опрашивается заново, со снижением приоритета элемента данных, который перевел интерфейс в недостижимое состояние. Если устанавливается связь с интерфейсом, который был недостижимым, мониторинг автоматически возвращается к норме:

```
resuming Zabbix agent checks on host "New host": connection restored
```

**Note:**
После того как интерфейс становится доступным, опрос всех элементов данных немедленно не осуществляется узлом сети по двум причинам:

- это может создать слишком большую нагрузку на узел сети,
- время восстановления интерфейса не всегда совпадает с плановым опросом элемента данных согласно заданному расписанию проверок.

Поэтому, когда интерфейс становится доступным, опрос элементов данных производится не сразу, а перепланируется в соответствии с расписанием их следующих проверок.

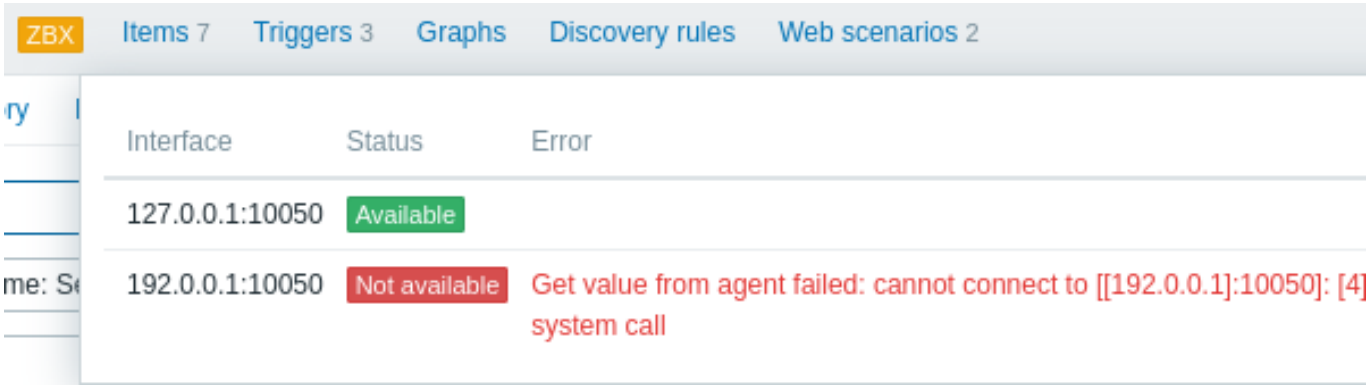
Недоступный интерфейс

После того как UnreachablePeriod завершился, а интерфейс так и не стал доступным, он считается недоступным.

В журнале сервера это обозначается записями такого рода:

```
temporarily disabling Zabbix agent checks on host "New host": interface unavailable
```

При этом в веб-интерфейсе иконка доступности узла сети меняет свой цвет с зеленого/серого на желтый/красный (при наведении указателя мыши на иконку доступности узла сети можно видеть информацию о недоступности интерфейса):




Параметр **UnavailableDelay** определяет, как часто интерфейс проверяется повторно в течение его недоступности.

По умолчанию — раз в 60 секунд (таким образом, в этом случае «temporarily disabling» из файла журнала выше будет означать деактивацию проверок сроком на одну минуту).

Когда соединение с интерфейсом будет восстановлено, мониторинг тоже автоматически вернётся к норме:

```
enabling Zabbix agent checks on host "New host": interface became available
```

#### 12. Удалённый мониторинг статистики Zabbix

Обзор

Некоторые внутренние метрики Zabbix сервера и прокси можно сделать доступными удалённо для другого экземпляра Zabbix или сторонних утилит. Может быть полезно, чтобы поставщики поддержки/сервисов имели возможность удалённо мониторить Zabbix сервера/прокси своих клиентов, или в организациях, где Zabbix не является основным инструментом мониторинга, чтобы внутренние метрики Zabbix могли контролироваться сторонней системой при использовании зонтичного мониторинга.

Внутренняя статистика Zabbix открыта для настраиваемого списка адресов, перечисленных в новом параметре «StatsAllowedIP» сервера/прокси. Запросы будут приниматься только с этих адресов.

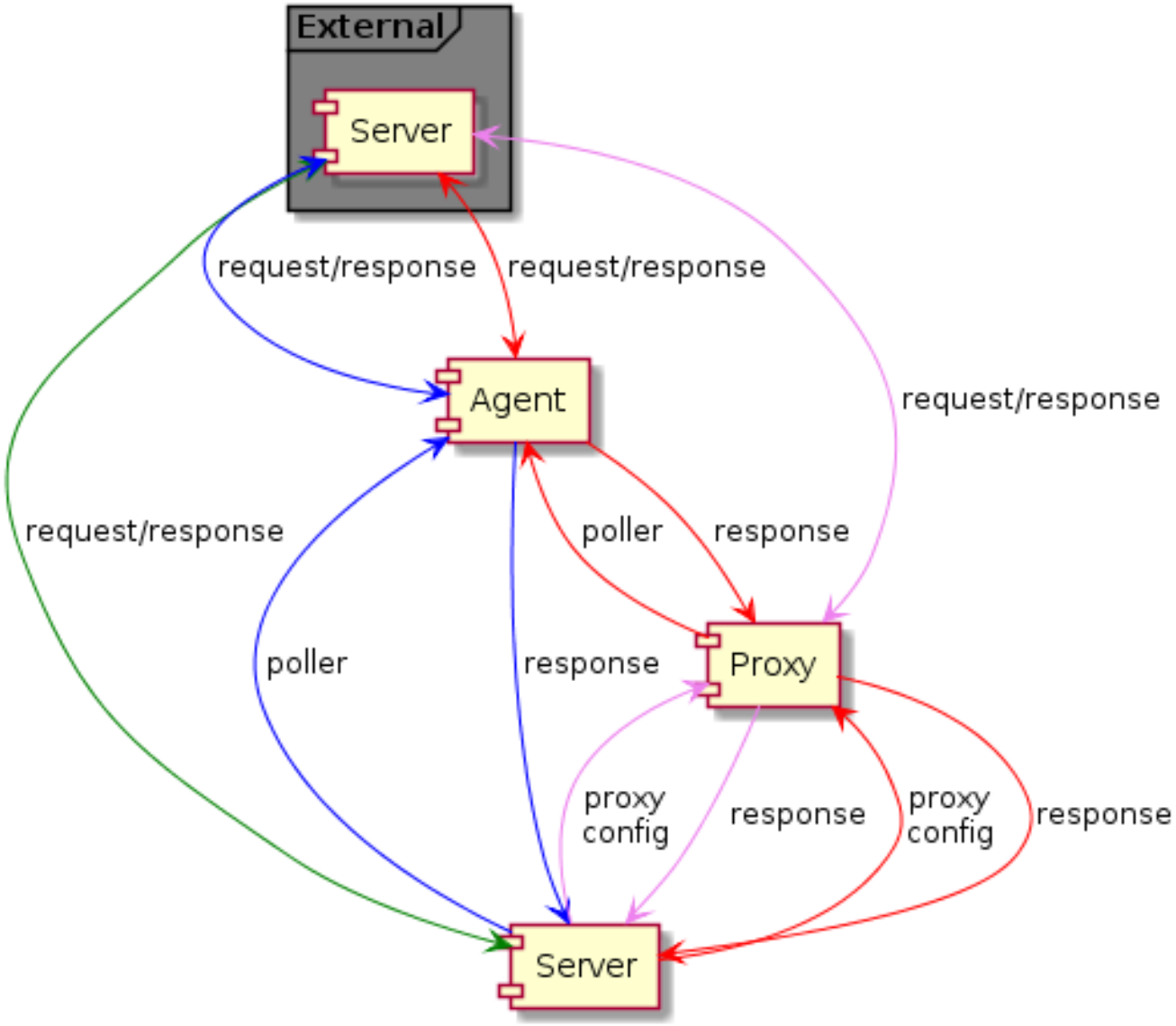
Элементы данных

Для настройки запроса внутренней статистики с другого экземпляра Zabbix вы можете использовать два элемента данных:

- Внутренний элемент данных `zabbix[stats,<ip>,<  >]` - для прямых удалённых запросов к Zabbix серверу/прокси. <ip> и <порт> используются для идентификации необходимого целевого экземпляра.
- Элемент данных агента `zabbix.stats[<ip>,<  >]` для удалённых запросов к Zabbix серверу/прокси через агента. <ip> и <порт> используются для идентификации необходимого целевого экземпляра.

Смотрите также: Внутренние элементы данных, Элементы данных Zabbix агента.

Следующая диаграмма иллюстрирует использование обоих элементов данных в зависимости от контекста.




- - Сервер → внешний экземпляр Zabbix (`zabbix[stats,<ip>,< >]`)
- - Сервер → прокси → внешний экземпляр Zabbix (`zabbix[stats,<ip>,< >]`)
- - Сервер → агент → внешний экземпляр Zabbix (`zabbix.stats[<ip>,< >]`)
- - Сервер → прокси → агент → внешний экземпляр Zabbix (`zabbix.stats[<ip>,< >]`)

Чтобы убедиться, что целевой экземпляр разрешает запросы к себе со стороны внешнего экземпляра, укажите адрес внешнего экземпляра в параметре 'StatsAllowedIP' на целевом экземпляре.

Предоставляемые метрики

Элементы данных собирают статистику одним запросом и возвращают JSON, который является основой зависимых элементов данных и из которого они получают данные. Любым из двух элементов данных возвращаются следующие внутренние метрики:

- `zabbix[boottime]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[preprocessing]` (только сервер)
- `zabbix[preprocessing_queue]` (только сервер)
- `zabbix[process,< >,< >,< >]` (статистика только на основе типов процессов)
- `zabbix[rcache,< >,< >]`
- `zabbix[requiredperformance]`
- `zabbix[triggers]` (только сервер)
- `zabbix[uptime]`

- zabbix[vcache,buffer,<  >] (только сервер)
- zabbix[vcache,cache,<  >]
- zabbix[version]
- zabbix[vmware,buffer,<  >]
- zabbix[wcache,<  >,<  >] (тип кэша «trends» только сервер)

Шаблоны

Для удалённого мониторинга внутренних метрик Zabbix сервера или прокси с внешнего экземпляра доступны шаблоны:

- Remote Zabbix server health
- Remote Zabbix proxy health

Обратите внимание, что для использования шаблонов удалённого мониторинга нескольких внешних экземпляров, для мониторинга каждого внешнего экземпляра потребуется отдельный узел сети.

Процесс траппер

Получение запросов внутренних метрик с внешнего экземпляра Zabbix обрабатывается процессом траппер, который проверяет запрос, собирает метрики, создаёт буфер данных JSON и отправляет назад подготовленный JSON, например, с сервера:

```
{
  "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
      }
    }
  }
}
```

Элементы данных внутренней очереди

Также имеются два других элемента данных, которые позволяют получить внутреннюю статистику очереди удалённым запросом с другого экземпляра Zabbix:

- Внутренний элемент данных `zabbix[stats,<ip>,<  >,queue,< >,< >]` — для прямых запросов внутренней очереди с удалённого Zabbix сервера/прокси
- Элемент данных агента `zabbix.stats[<ip>,<  >,queue,< >,< >]` — для запросов внутренней очереди удалённых Zabbix сервера/прокси через агента

Смотрите также: Внутренние элементы данных, Элементы данных Zabbix агента.

### 13. Настройка Kerberos с Zabbix

Обзор

В Zabbix аутентификацию Kerberos можно использовать в веб-мониторинге и HTTP элементах данных.

На этой странице описывается пример настройки Kerberos для выполнения Zabbix сервером веб-мониторинга страницы `www.example.com` с использованием принципала Kerberos для процесса Zabbix в Debian/Ubuntu.

Конфигурация

1. Установите KDC и клиентские утилиты:

```
sudo apt update
sudo apt install krb5-kdc krb5-admin-server krb5-user
```

Во время настройки пакетов отвечайте на запросы, например:

```
Default Kerberos version 5 realm: EXAMPLE.COM
Kerberos servers for your realm: localhost (or your FQDN)
Administrative server for your Kerberos realm: localhost (or your FQDN)
```

2. Сопоставьте понятное имя хоста (необязательно, для локального тестирования).

Отредактируйте /etc/hosts и добавьте запись для вашего DC и веб-сервера, если у вас нет DNS:

```
sudo vi /etc/hosts
```

Пример строки, которую можно добавить:

```
192.168.1.100  dc01.example.com dc01
```

3. Настройте Kerberos-клиент и realm KDC:

```
sudo vi /etc/krb5.conf
```

Пример настроек:

```
[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
```

Если вы планируете использовать `.localdomain` или другие непубличные имена, добавьте явные сопоставления domain→realm, чтобы работало сопоставление hostname→realm. Несоответствия здесь вызывают ошибки `Server not found in Kerberos database`.

4. Инициализируйте базу данных Kerberos (однократно, на хосте KDC). При запросе задайте надежный мастер-пароль:

```
sudo krb5_newrealm
```

5. Создайте principal `HTTP/host.fqdn@REALM`, используя точное имя хоста, которое будут применять клиенты; предпочтительно в нижнем регистре (например, `HTTP/web.example.com@EXAMPLE.COM`). Несоответствие регистра или имени вызывает `Server not found in Kerberos database`.

```
sudo kadmin.local
```

Внутри `kadmin.local`:

```
addprinc kerb-admin@EXAMPLE.COM      # administrative principal
addprinc -randkey HTTP/dc01.example.com@EXAMPLE.COM
ktadd -k /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
quit
```

Переместите keytab на веб-хост (или оставьте локально, если это та же машина) и задайте права, пригодные для Apache:

```
chown www-data:www-data /etc/apache2/http.keytab
chmod 600 /etc/apache2/http.keytab
#### verify
sudo -u www-data -k /etc/apache2/http.keytab
```

6. Установите и включите модуль Apache GSSAPI:

```
sudo apt install libapache2-mod-auth-gssapi
sudo a2enmod auth_gssapi
sudo a2enmod headers
sudo systemctl restart apache2
```

> **Attention:**
> Не все версии mod_auth_gssapi поддерживают каждую директиву `Gssapi*`. Если Apache завершается с ошибкой `Invalid command 'GssapiCredStore'`, удалите неподдерживаемую директиву или обновите модуль.

7. Настройте VirtualHost (при необходимости скорректируйте `DocumentRoot` / путь к вашему веб-интерфейсу Zabbix):

```
sudo vi /etc/apache2/sites-available/zabbix.conf
```

Внутри `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>
```

Перезапустите Apache:

```
sudo systemctl restart apache2
```

8. Включите/запустите службы KDC и проверьте прослушиваемые порты (хост KDC):

```
sudo systemctl enable --now krb5-kdc krb5-admin-server
ss -tnlp | grep :80    # or: sudo netstat -tnlp | grep :80
```

9. Получите TGT для тестирования (запускайте от имени пользователя, который будет использовать билет).

Ожидается, что в списке билетов будет `krbtgt/EXAMPLE.COM@EXAMPLE.COM`. Запускайте `kinit` от имени того же

пользователя ОС, которому нужен билет (например, zabbix для проверок веба или www-data/Apache для интерактивных тестов SSO в браузере). Билеты, выданные другому пользователю ОС, не будут видны, если не настроены KRB5CCNAME и права доступа.

```
kinit kerb-admin@EXAMPLE.COM
klist
```

10. Проверьте обмен SPNEGO с помощью curl (с клиента с действительным TGT). Ответ 200 OK (или перенаправление в приложение) означает, что SPNEGO успешно выполнен:

```
curl -v --negotiate -u : http://dc01.example.com/
```

11. При необходимости, если веб-интерфейс Zabbix должен принимать входы по HTTP-аутентификации, включите HTTP-аутентификацию в веб-интерфейсе Zabbix (ui/conf/zabbix.conf.php):

```
$ALLOW_HTTP_AUTH = true;
```

В веб-интерфейсе перейдите в *Users* > *Authentication* и откройте вкладку *HTTP settings*. Отметьте флажок *Enable HTTP authentication* и нажмите *Ok* во всплывающем окне. Выберите "HTTP login form" в раскрывающемся списке *Default login form*. Решите, подходит ли *Case-sensitive login* для политики вашего каталога. Нажмите кнопку *Update*, чтобы завершить.

12. Настройка браузера (Firefox используется в качестве примера): задайте network.negotiate-auth.trusted-uris для хоста(ов), выполняющих Negotiate (dc01.example.com), чтобы браузер автоматически отправлял Kerberos-токены.

Внутри about:config:

```
network.negotiate-auth.trusted-uris = dc01.example.com
```

Теперь при переходе на http://dc01.example.com вы должны сразу входить в Zabbix без формы.

13. Поддерживайте ключи/билеты в актуальном состоянии. По умолчанию срок действия Kerberos-билета составляет примерно 10 часов. Добавьте cron/systemd timer, чтобы избежать истечения срока:

```
####for the web service
kinit -kt /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
####for the monitoring user
kinit -kt /var/lib/zabbix/kerb.keytab kerb-admin@EXAMPLE.COM
```

14. Проверки обслуживания:

- klist -k /etc/apache2/http.keytab — проверьте, что service principal присутствует в keytab.
- sudo tail -f /var/log/apache2/error.log — отслеживайте ошибки GSSAPI (gss_acquire_cred[_from]() failed to get server creds означает проблему с keytab/правами доступа или отсутствие principal).
- Если curl --negotiate возвращает 401/403, это часто означает неверный principal, отсутствие билета, несоответствие Host header или проблему с правами файловой системы; проверьте журналы и сопоставления доменов в /etc/krb5.conf.

Замечания о безопасности и правах доступа к файлам

Keytab-файлы должны быть доступны на чтение только той учётной записи, которой они нужны. Пример доступов: 0400 с владельцем zabbix:zabbix для keytab-файла пользователя zabbix, либо 0440 и root:www-data для keytab-файла для Apache.

Избегайте хранения долговременных паролей в открытом виде на хосте. По возможности используйте keytab-файлы или принципалы машин, подключённых к домену.

При запуске тестов или скриптов, устанавливающих KRB5CCNAME или копирующих keytab-файлы, дважды проверьте владельцев и права доступа после операции — веб-сервер, отклоняющий учётные данные, обычно является проблемой с правами доступа к файлу.

#### 14. Параметры modbus.get

Обзор

В таблице ниже представлены детальные описания параметров элемента данных modbus.get.

Parameters

| Параметр | Описание | По умолчанию | Пример |
|---|---|---|---|
| *точка входа* | Протокол и адрес точки входа, определяемый в виде :// _<br><br>Возможные значения протокола: *rtu*, *ascii* (только Агент 2), *tcp*<br><br>Формат строки подключения:<br><br>для *tcp* — :<br>для последовательной линии: *rtu*, *ascii* — _ : :<br>где<br>«скорость» — 1200, 9600 и т.д.<br>«параметры» — количество бит (5,6,7 или 8), чётность (n,e или o для нет(none)/чёт(even)/нечет(odd)), стоп-битов (1 или 2) | протокол: нет<br><br>протокол *rtu/ascii*: имя_порта: нет скорость: 115200 параметры: 8n1<br><br>протокол *tcp*: адрес: нет порт: 502 | tcp://192.168.6.1:511<br>tcp://192.168.6.2<br>tcp://[::1]:511<br>tcp://::1<br>tcp://localhost:511<br>tcp://localhost<br>rtu://COM1:9600:8n<br>ascii://COM2:1200:7o2<br>rtu://ttyS0:9600<br>ascii://ttyS1 |
| *id ведомого* | Адрес modbus устройства, для которого он предназначен (от 1 до 247), см. MODBUS Messaging Implementation Guide [en] (страница 23)<br><br>устройство tcp (не GW) будет игнорировать это поле | для последовательной линии: 1<br><br>для tcp: 255 (0xFF) | 2 |
| *функция* | Пусто или значение поддерживаемой функции:<br><br>1 — Чтение Флага,<br>2 — Чтение Дискретного Входа,<br>3 — Чтение Регистров Временного Хранения,<br>4 — Чтение Входных Регистров | пусто | 3 |
| *адрес* | адрес первого регистра, флага или входа.<br><br>Если параметр «функция» пустой, то «адрес» должен быть в диапазоне для:<br>Флага — 00001 — 09999<br>Дискретного входа — 10001 — 19999<br>Входного регистра — 30001 — 39999<br>Регистра Временного Хранения — 40001 — 49999<br><br>Если параметр «функция» не пуст, то поле «адрес» будет от 0 до 65535 использоваться без модификации (PDU) | пустая функция: 00001<br><br>непустая функция: 0 | 9999 |
| *количество* | Количество последовательных «типов», которые будут считаны с устройства, где:<br><br>для флага или дискретного входа «тип» = 1 бит<br>в остальных случаях: (количество*sizeof(тип))/2 = реальное количество регистров для чтения<br>Если «сдвиг» не равен 0, то значение будет добавлено к «реальному количеству»<br>Допустимый диапазон для «реального количества» составляет 1:65535 | 1 | 2 |

| Параметр | Описание | По умолчанию | Пример |
|---|---|---|---|
| *тип* | Тип данных:<br><br>для Чтения Флага и Чтения Дискретного Входа — *bit*<br><br>для Чтения Регистров Временного Хранения и Чтения Входных Регистров:<br>*int8* — 8 бит<br>*uint8* — 8 бит (без знака)<br>*int16* — 16 бит<br>*uint16* — 16 бит (без знака)<br>*int32* — 32 бита<br>*uint32* — 32 бита (без знака)<br>*float* — 32 бита<br>*uint64* — 64 бита (без знака)<br>*double* — 64 бита | bit<br>uint16 | uint64 |
| *порядок следования байтов* | Тип порядка следования байтов:<br>*be* — Big Endian<br>*le* — Little Endian<br>*mbe* — Mid-Big Endian<br>*mle* — Mid-Little Endian<br><br>Ограничения:<br>для 1 бит — be<br>для 8 бит — be,le<br>для 16 бит — be,le | be | le |
| *сдвиг* | Количество регистров, начиная с «адреса», результаты которых будут отброшены.<br><br>Размер каждого регистра составляет 16 бит (требуется для поддержки оборудования, не поддерживающего произвольный доступ для чтения). | 0 | 4 |

### 15. Создание пользовательских имён счётчиков производительности для VMware

Обзор

Путь счетчика производительности VMware имеет формат `group/counter[rollup]`, где:

- `group` - группа счетчиков производительности, например, *cpu*
- `counter` - имя счетчика производительности, например, *usagemhz*
- `rollup` - тип rollup счетчика производительности, например, *average*

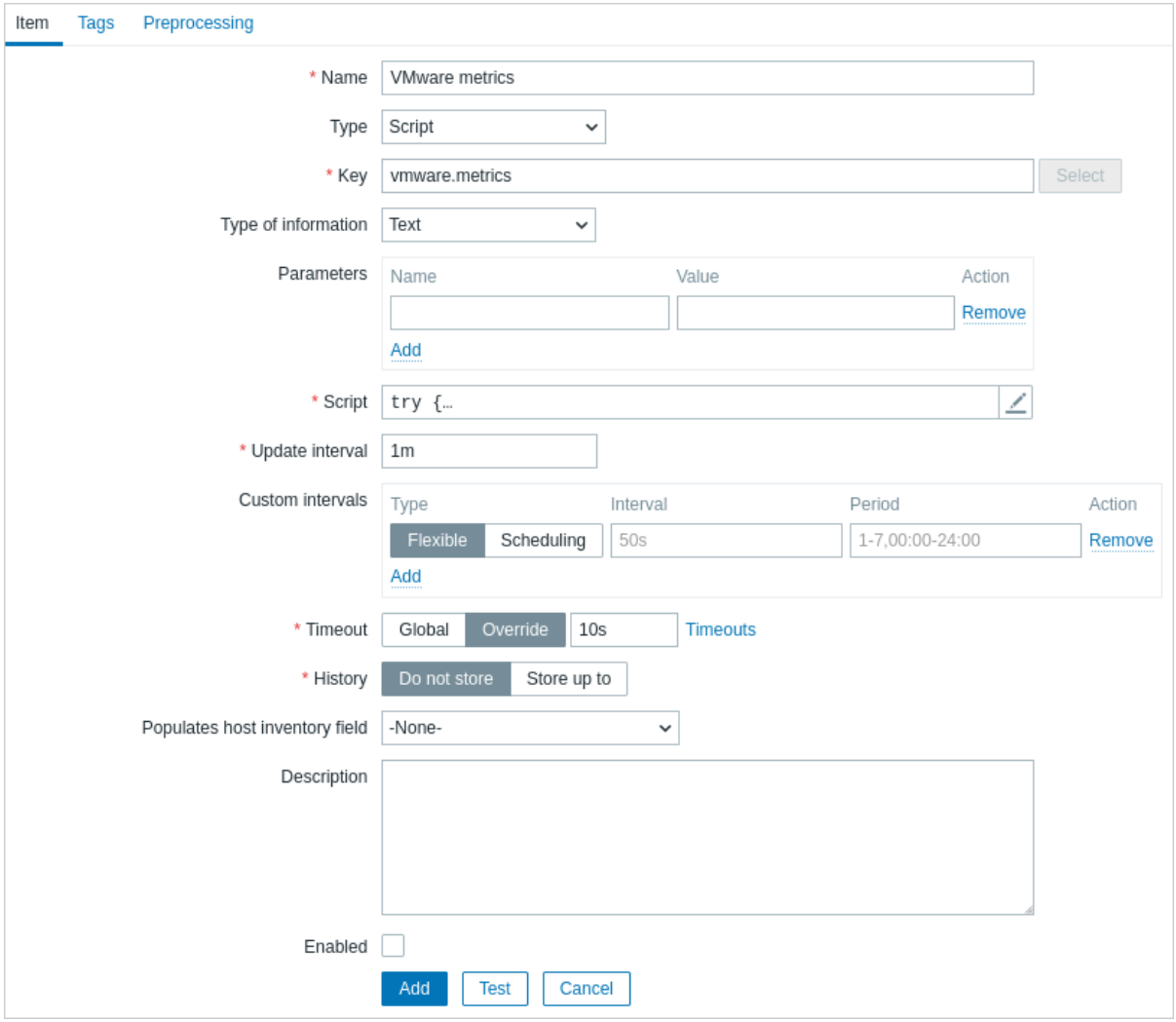
Таким образом, приведенный выше пример даст следующий путь счетчика: `cpu/usagemhz[average]`

Описания групп счетчиков производительности, имена счетчиков и типы rollup можно найти в документации VMware.

Можно получить внутренние имена и создать пользовательские имена счетчиков производительности, используя элемент данных script в Zabbix.

Конфигурация

1. Создайте отключенный элемент данных типа Script на основном узле сети VMware (где присутствует элемент данных **eventlog[]**) со следующими параметрами:




- *Name*: VMware metrics
- *Type*: Script
- *Key*: vmware.metrics
- *Type of information*: Text
- *Script*: скопируйте и вставьте скрипт, приведенный ниже
- *Timeout*: 10
- *History*: Do not store
- *Enabled*: unmarked

Скрипт

```
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;
```

После настройки элемента данных нажмите кнопку *Test*, затем *Get value*.

Скопируйте полученный XML в любой XML-форматтер и найдите нужную метрику.

Пример XML для одной метрики:

```
<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>
```

Используйте XPath, чтобы извлечь путь счетчика из полученного XML. Для приведенного выше примера XPath будет следующим:

| field | xPath | value |
|---|---|---|
| group | //groupInfo[../key=6]/key | cpu |
| counter | //nameInfo[../key=6]/key | usagemhz |
| rollup | //rollupType[../key=6] | average |

Итоговый путь счетчика производительности в этом случае: `cpu/usagemhz[average]`

#### 16. Значения, возвращаемые для system.sw.packages.get

Обзор

В этом разделе приведены сведения о значениях, возвращаемых для элемента данных Zabbix агента `system.sw.packages.get`.

Подробности

Вывод этого элемента данных представляет собой массив объектов, каждый из которых содержит следующие ключи:

- **name** — имя пакета
- **manager** — менеджер пакетов, который сообщил эти данные (`rpm`, `dpkg`, `pacman`, `pkgtool` или `portage`)
- **version** — версия пакета
- **size** — размер несжатого пакета в байтах (если недоступно, устанавливается значение 0)
- **arch** — архитектура пакета
- **buildtime** — объект с 2 элементами:
  - **timestamp** — отметка времени UNIX, когда пакет был собран (если недоступно, устанавливается значение 0)
  - **value** — дата и время, когда был собран пакет, в удобном для чтения виде (если недоступно, устанавливается пустая строка)
- **installtime** — объект с 2 элементами:
  - **timestamp** — отметка времени UNIX, когда пакет был установлен (если недоступно, устанавливается значение 0)
  - **value** — дата и время установки пакета в удобном для чтения виде (если недоступно, устанавливается пустая строка)

Пример:

```
[
    {
        "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": ""
        }
    }
]
```

#### 17. Значения, возвращаемые для net.dns.get

Overview

В этом разделе приведены сведения о значениях, возвращаемых для элемента данных Zabbix агента 2 `net.dns.get`.

Подробности

Вывод этого элемента данных — объект, содержащий информацию о записи DNS на основе параметров, указанных в ключе элемента данных.

Например, элемент данных `net.dns.get[,example.com]` может вернуть следующий JSON отклонённого запроса:

```
{
    "flags": [
        "RA"
    ],
    "query_time": "0.019030",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "SOA"
        }
    ],
    "response_code": "REFUSED",
    "zbx_error_code": 0
}
```

С указанием IP-адреса DNS-сервера, элемент данных `net.dns.get[192.0.2.0,example.com]` может вернуть следующий 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
}
```


Если возникла проблема с подключением, элемент данных `net.dns.get[192.0.2.0,example.com]` может содержать ошибку:


```
{
    "zbx_error_code": -1,
    "zbx_error_msg": "Communication error: read udp 192.0.2.0:12345->192.0.2.0:53: i/o timeout"
}
```


Возможны следующие типы кодов ошибок:

| Сценарий | "zbx_error_code" | "zbx_error_msg" |
|---|---|---|
| Ошибок нет, и ответ DNS получен и проанализирован. | 0 | |
| DNS не работает. | -1 | "Communication error" |
| Ошибка при анализе JSON | -2 | "Received unexpected response" |

С дополнительными параметрами элемент данных `net.dns.get[192.0.2.0,example.com,ANY,5,5,tcp,"cdflag,rdflag,dnss` может вернуть следующий 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": "192.0.2.0"
            },
            "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
}
```


Смотрите также

Более подробную информацию о записях DNS смотрите [en]:

- Доменные имена — Реализация и спецификация
- Параметры системы доменных имен (DNS)

#### 18 Примечания к элементам данных system.cpu.util в Windows

system.cpu.util элемент данных предоставляет процент загрузки CPU.

Когда на агенте Zabbix для Windows запускается процесс сборщика, инициализируется буфер для N элементов данных для N логических процессоров (потоков). Значения периодически обновляются в буфере с использованием счетчиков производительности Windows. Эти значения возвращаются при запросе `system.cpu.util[n]`, где 0 <= n < N.

Еще один элемент в буфере используется для system.cpu.util[total].

Zabbix автоматически выбирает разные счетчики производительности, чтобы поддерживать как системы NUMA, так и более старые версии Windows без поддержки групп процессоров.

Если количество логических процессоров (потоков) меньше или равно 64, а число групп процессоров равно 1, то:

```
N —                        (    ).

\Processor(_Total)\% Processor Time
\Processor(0)\% Processor Time
\Processor(1)\% Processor Time
\Processor(2)\% Processor Time
...
\Processor(N-1)\% Processor Time
```

Иначе:

```
G —                    ,   N —                    (    )      .

\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
```

#### 5 Поддерживаемые функции

Нажмите на соответствующую группу функций, чтобы увидеть подробности.

| Группа функций | Функции |
|---|---|
| Агрегатные функции | avg, bucket_percentile, count, histogram_quantile, item_count, kurtosis, mad, max, min, skewness, stddevpop, stddevsamp, sum, sumofsquares, varpop, varsamp |
| Функции Foreach | avg_foreach, bucket_rate_foreach, count_foreach, exists_foreach, last_foreach, max_foreach, min_foreach, sum_foreach |
| Побитовые функции | bitand, bitlshift, bitnot, bitor, bitrshift, bitxor |
| Функции даты и времени | date, dayofmonth, dayofweek, now, time |
| Функции истории | change, changecount, count, countunique, find, first, firstclock, fuzzytime, last, lastclock, logeventid, logseverity, logsource, logtimestamp, monodec, monoinc, nodata, percentile, rate |
| Функции трендов | baselinedev, baselinewma, trendavg, trendcount, trendmax, trendmin, trendstl, trendsum |
| Математические функции | 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 |
| Функции операторов | between, in |
| Прогностические функции | forecast, timeleft |

| Группа функций | Функции |
|---|---|
| Строковые функции | ascii, bitlength, bytelength, char, concat, insert, jsonpath, left, length, ltrim, mid, repeat, replace, right, rtrim, trim, xmlxpath |

Если не указано иное, эти функции поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

Функции Foreach поддерживаются только для агрегатных вычислений.

#### 1 Агрегатные функции

Если не указано иное, все перечисленные здесь функции поддерживаются в:

- выражениях триггеров
- рассчитываемых элементах данных
- макросах выражений

Агрегатные функции могут работать либо с:

- историей элементов данных, например, `min(/host/key,1h)`
- функциями foreach в качестве единственного параметра, например, `min(last_foreach(/*/key))` (только в рассчитываемых элементах данных; не может использоваться в триггерах)

Функции перечислены без дополнительной информации. Щелкните функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| avg | Среднее значение элемента данных за заданный период вычисления. |
| bucket_percentile | Вычисляет процентиль по корзинам гистограммы. |
| count | Количество значений в массиве, возвращаемом функцией foreach. |
| histogram_quantile | Вычисляет φ-квантиль по корзинам гистограммы. |
| item_count | Количество существующих элементов в конфигурации, соответствующих критериям фильтра. |
| kurtosis | "Хвостатость" распределения вероятностей в собранных значениях за заданный период вычисления. |
| mad | Медианное абсолютное отклонение в собранных значениях за заданный период вычисления. |
| max | Наибольшее значение элемента данных за заданный период вычисления. |
| min | Наименьшее значение элемента данных за заданный период вычисления. |
| skewness | Асимметрия распределения вероятностей в собранных значениях за заданный период вычисления. |
| stddevpop | Стандартное отклонение генеральной совокупности в собранных значениях за заданный период вычисления. |
| stddevsamp | Выборочное стандартное отклонение в собранных значениях за заданный период вычисления. |
| sum | Сумма собранных значений за заданный период вычисления. |
| sumofsquares | Сумма квадратов собранных значений за заданный период вычисления. |
| varpop | Дисперсия генеральной совокупности собранных значений за заданный период вычисления. |
| varsamp | Выборочная дисперсия собранных значений за заданный период вычисления. |

Общие параметры

- `/ _ /` является общим обязательным первым параметром для функций, ссылающихся на историю элемента данных узла сети
- `( |# )<: _ >` является общим вторым параметром для функций, ссылающихся на историю элемента данных узла сети, где:
  - **сек** — максимальный период вычисления в секундах (могут использоваться суффиксы времени), либо
  - **#число** — максимальная область вычисления среди последних собранных значений (если начинается со знака решётки)

- **сдвиг_времени** (опционально) позволяет сдвигать точку вычислений по времени назад в прошлое. Смотрите более подробную информацию относительно того, как указывать сдвиг времени.

Подробности функций

Некоторые общие замечания о параметрах функций:

- Параметры функций разделяются запятой
- Необязательные параметры функций (или части параметров) обозначаются угловыми скобками < >
- Параметры, специфичные для функций, описаны с каждой функцией
- Параметры / _ / и ( |# )<: _ > никогда не должны заключаться в кавычки

avg(/host/key,(sec|#num)<:time shift>)

Среднее значение элемента данных в течение заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемые функции foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Параметры: см. общие параметры.

Сдвиг времени полезен, когда нужно сравнить текущее среднее значение со средним значением некоторое время назад.

Примеры:

```
avg(/host/key,1h) #calculate the average value for the last hour until now
avg(/host/key,#5) #calculate the average value of the five latest values
avg(/host/key,1h:now-1d) #calculate the average value for an hour from 25 hours ago to 24 hours ago from n
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"])) #calculate the average processor load
avg(/host/proc.num,5m)>300 #trigger if the average number of processes in the last 5 minutes has been abov
```

bucket_percentile(item filter,time period,percentage)

Вычисляет процентиль по корзинам гистограммы.<br>

Параметры:

- **item filter** - см. item filter;<br>
- **time period** - см. time period;<br>
- **percentage** - процент (0-100).

Комментарии:

- Поддерживается только в вычисляемых элементах данных;
- Эта функция является псевдонимом для `histogram_quantile(percentage/100, bucket_rate_foreach(item filter, time period, 1))`.

Примеры:

```
bucket_percentile(/*/http_request_duration_seconds_bucket[*],5m,95) #      95-                    5
bucket_percentile(/*/apiserver_request_duration_seconds_bucket[*],10m,99) #      99-              API
```

count(func_foreach(item filter,<time period>),<operator>,<pattern>)

Количество значений в массиве, возвращаемом функцией foreach.<br> Поддерживаемые функции foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Параметры:

- **func_foreach** - функция foreach, для которой нужно подсчитать количество возвращаемых значений. Подробности см. в разделе функции foreach. Обратите внимание, что count_foreach и bucket_rate_foreach поддерживают дополнительные параметры.
- **item filter** - см. синтаксис фильтра элементов данных;<br>
- **time period** - см. период времени;<br>
- **operator** (должен быть в двойных кавычках). Поддерживаемые `operators`:<br>*eq* - равно<br>*ne* - не равно<br>*gt* - больше<br>*ge* - больше или равно<br>*lt* - меньше<br>*le* - меньше или равно<br>*like* - совпадает, если содержит шаблон (с учетом регистра)<br>*bitand* - побитовое И<br>*regexp* - совпадение с учетом регистра с регулярным выражением, заданным в `pattern`<br>*iregexp* - совпадение без учета регистра с регулярным выражением, заданным в `pattern`<br>
- **pattern** - требуемый шаблон (строковые аргументы должны быть в двойных кавычках); поддерживается, если *operator* указан в третьем параметре.

Комментарии:

- Использование **count()** с функцией foreach, работающей с историческими данными (max_foreach, avg_foreach и т. д.), может повлиять на производительность, тогда как использование **exists_foreach()**, которая работает только с конфигурационными данными, такого эффекта не вызовет.
- Необязательные параметры *operator* или *pattern* нельзя оставлять пустыми после запятой, их можно только полностью опустить.
- При использовании *bitand* в качестве третьего параметра четвертый параметр *pattern* можно указать как два числа, разделенные символом '/': **number_to_compare_with/mask**. count() вычисляет "побитовое И" для значения и *mask* и сравнивает результат с *number_to_compare_with*. Если результат "побитового И" равен *number_to_compare_with*, значение учитывается.<br>Если *number_to_compare_with* и *mask* равны, достаточно указать только *mask* (без '/').
- При использовании *regexp* или *iregexp* в качестве третьего параметра четвертый параметр *pattern* может быть обычным или глобальным (начинающимся с '@') регулярным выражением. В случае глобальных регулярных выражений чувствительность к регистру наследуется из настроек глобального регулярного выражения. Для сопоставления regexp значения с плавающей точкой всегда будут представлены с 4 знаками после '.'. Также обратите внимание, что для больших чисел разница в десятичном (хранимом в базе данных) и двоичном (используемом сервером Zabbix) представлении может повлиять на 4-й десятичный знак.

Примеры:

```
count(max_foreach(/*/net.if.in[*],1h)) #calculate the number of net.if.in items that received data in the
count(last_foreach(/*/vfs.fs.size[*,pused]),"gt",95) #calculate the number of file systems with over 95% c
```

histogram_quantile(quantile,bucket1,value1,bucket2,value2,...)

Вычисляет φ-квантиль по корзинам гистограммы.<br> Поддерживается foreach function: *bucket_rate_foreach*.

Параметры:

- **quantile** - 0 ≤ φ ≤ 1;<br>
- **bucketN, valueN** - пары параметров, введенные вручную (>=2), или ответ bucket_rate_foreach.

Комментарии:

- Поддерживается только в вычисляемых элементах данных;
- Функционально соответствует 'histogram_quantile' в PromQL;
- Возвращает -1, если значения последней корзины 'Infinity' ("+*inf*") равны 0.

Примеры:

```
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)

Количество существующих элементов данных в конфигурации, которые соответствуют критериям фильтра.<br> Поддерживаемый тип значения: *Integer*.

Параметр:

- **item filter** - критерии выбора элементов данных, позволяют ссылаться по группе узлов сети, узлу сети, ключу элемента данных и тегам. Поддерживаются подстановочные знаки. См. item filter для получения дополнительных сведений.<br>

Комментарии:

- Поддерживается только в вычисляемых элементах данных;
- Работает как псевдоним для функции *count(exists_foreach(item_filter))*.

Примеры:

```
item_count(/*/agent.ping?[group="Host group 1"]) #calculate the number of hosts with the agent.ping item i
```

kurtosis(/host/key,(sec|#num)<:time shift>)

"Хвостатость" распределения вероятностей в собранных значениях за заданный период оценки. См. также: Kurtosis.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая функция foreach: *last_foreach*.

Параметры: см. общие параметры.

Примеры:

```
kurtosis(/host/system.cpu.util[,user],5m)>5 #trigger on sharp spikes of CPU utilization over 5 minutes
```

mad(/host/key,(sec|#num)<:time shift>)

Медианное абсолютное отклонение в собранных значениях за заданный период оценки. См. также: Median absolute deviation.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. common parameters.

Примеры:

```
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>)

Наибольшее значение элемента данных в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемые функции foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Параметры: см. общие параметры.

Примеры:

```
max(/host/key,1h) - min(/host/key,1h) #                                                    (
max(last_foreach(/*/vfs.fs.size[*,pused]?[group="Linux servers"])) #                           Li
```

min(/host/key,(sec|#num)<:time shift>)

Наименьшее значение элемента данных в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемые функции foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Параметры: см. общие параметры.

Примеры:

```
max(/host/key,1h) - min(/host/key,1h) #                                                    (
min(last_foreach(/*/vfs.file.contents["/sys/class/net/enp0s3/operstate"]?[group="Linux servers"])) #
```

skewness(/host/key,(sec|#num)<:time shift>)

Асимметрия распределения вероятностей в собранных значениях за заданный период оценки. См. также: Skewness.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. common parameters.

Примеры:

```
skewness(/host/vm.memory.size[used],5m)<-20 #trigger on sudden negative drops indicating crashes/restarts
```

stddevpop(/host/key,(sec|#num)<:сдвиг времени>)

Популяционное стандартное отклонение в собранных значениях за заданный период оценки. См. также: Standard deviation.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. common parameters.

Примеры:

```
stddevpop(/host/system.cpu.util[,user],10m)>7 #    ,          CPU            (                    7    1
```

stddevsamp(/host/key,(sec|#num)<:сдвиг времени>)

Выборочное стандартное отклонение по собранным значениям в пределах заданного периода оценки. См. также: Standard deviation.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. общие параметры.

Для работы этой функции требуется как минимум два значения данных.

Примеры:

```
stddevsamp(/host/system.cpu.util[,user],10m)>7 #    ,          CPU            (                    7    1
```

sum(/host/key,(sec|#num)<:time shift>)

Сумма собранных значений за заданный период оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемые foreach functions: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Параметры: см. common parameters.

Примеры:

```
sum(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #calculate the total incoming network traffic f
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"])) #calculate the total disk space for all
sum(last_foreach(/*/net.dns[*,*,*])) #calculate the total number of successful DNS checks
```

sumofsquares(/host/key,(sec|#num)<:time shift>)

Сумма квадратов собранных значений в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. common parameters.

Примеры:

```
sumofsquares(/host/calculated.net.if.rate,1m) #                              (               )
sumofsquares(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #
```

varpop(/host/key,(sec|#num)<:time shift>)

Дисперсия совокупности собранных значений в пределах заданного периода оценки. См. также: Variance.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. common parameters.

Примеры:

```
varpop(/host/system.cpu.util[,user],10m)>50 #trigger if CPU usage fluctuates heavily (variance is above 50
```

varsamp(/узел сети/key,(sec|#num)<:time shift>)

Выборочная дисперсия собранных значений в пределах заданного периода оценки. См. также: Variance.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Поддерживаемая foreach function: *last_foreach*.

Параметры: см. common parameters.

Для работы этой функции требуется как минимум два значения данных.

Примеры:

```
varsamp(/      /system.cpu.util[,user],10m)>50 #    ,          CPU           (               50    10    )
```

См. все поддерживаемые функции.

#### 1. Функции foreach

Обзор

Функции foreach используются в агрегатных вычислениях для возврата одного агрегированного значения для каждого элемента данных, который выбран с помощью используемого **фильтра элементов данных**. Возвращается массив значений.

> **Attention:**
> Функции foreach поддерживаются только в вычисляемых элементах данных как часть формул агрегатных вычислений. Их нельзя вызывать в обычных ключах элементов данных или выражениях триггеров вне этого контекста.

Например, функция `avg_foreach` вернет массив значений, где каждое значение — это *среднее* значение истории выбранного элемента данных за указанный временной интервал.

Фильтр элементов данных является частью синтаксиса, используемого функциями foreach. В фильтре элементов данных поддерживается использование подстановочных знаков, поэтому необходимые элементы можно выбирать весьма гибко.

Поддерживаемые функции

| Функция | Описание |
|---|---|
| *avg_foreach* | Возвращает среднее значение для каждого элемента данных. |
| *bucket_rate_foreach* | Возвращает пары (верхняя граница разряда, значение доли), подходящие для использования в функции histogram_quantile(), где значение «верхняя граница разряда» — это значение параметра ключа элемента данных, определённого параметром <номер параметра>. |

| Функция | Описание |
|---|---|
| *count_foreach* | Возвращает количество значений для каждого элемента данных. |
| *exists_foreach* | Возвращает «1» для каждого активированного элемента данных. |
| *last_foreach* | Возвращает последнее значение для каждого элемента данных. |
| *max_foreach* | Возвращает наибольшее значение для каждого элемента данных. |
| *min_foreach* | Возвращает наименьшее значение для каждого элемента данных. |
| *sum_foreach* | Возвращает сумму значений для каждого элемента данных. |

Синтаксис функции

Функции foreach поддерживают два общих параметра: `item filter` (подробности см. ниже) и `time period`:

```
foreach_function(item filter,time period)
```

Например:

`avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m)`

вернет среднее значение за пять минут для каждого элемента данных `mysql.qps` в группе узлов сети MySQL.

Обратите внимание, что некоторые функции поддерживают дополнительные параметры.

Синтаксис фильтра элементов данных

Фильтр элементов данных:

```
/host/key[parameters]?[conditions]
```

состоит из четырех частей, где:<br> `host` - имя узла сети;<br> `key` - ключ элемента данных (без параметров);<br> `parameters` - параметры ключа элемента данных;<br> `conditions` - условия на основе группы узлов сети и/или тега элемента данных (в виде выражения).<br>

Пробелы допускаются только внутри выражения условий.

**Использование подстановочного знака**

- Подстановочный знак можно использовать для замены имени узла сети, ключа элемента данных или отдельного параметра ключа элемента данных.
- Либо узел сети, либо ключ элемента данных должны быть указаны без подстановочного знака. Поэтому `/host/*` и `/*/key` являются допустимыми фильтрами, а `/*/*` - недопустимым.
- Подстановочный знак нельзя использовать для *части* имени узла сети, ключа элемента данных, параметра ключа элемента данных.
- Подстановочный знак не сопоставляется более чем с одним параметром ключа элемента данных. Поэтому для каждого параметра подстановочный знак должен указываться отдельно (то есть `key[abc,*,*]`).

**Выражение условий**

Выражение условий поддерживает:

- Операнды:<br> **group** - группа узлов сети;<br> **tag** - тег элемента данных;<br> `"<text>"` - строковая константа, с использованием escape-символа \ для экранирования " и \.
- Операторы сравнения строк с учетом регистра: =, <>
- Логические операторы: `and`, `or`, `not`
- Группировку с помощью круглых скобок: ( )

Кавычки для строковых констант обязательны. Поддерживается только полное строковое сравнение с учетом регистра.

**Warning:**
При указании тегов в фильтре (то есть `tag="tagname:value"`) двоеточие ":" используется в качестве разделителя. Все, что следует после него, считается значением тега. Поэтому в настоящее время не поддерживается указание имени тега, содержащего ":".

**Примеры**

Сложный фильтр может использоваться для ссылки на ключ элемента данных, группу узлов сети и теги, как показано в примерах:

| Syntax example | Description |
|---|---|
| `/host/key[abc,*]` | Соответствует похожим элементам данных на этом узле сети. |
| `/*/key` | Соответствует одному и тому же элементу данных любого узла сети. |
| `/*/key?[group="ABC" and tag="tagname:value"]` | Соответствует одному и тому же элементу данных любого узла сети из группы ABC, имеющему теги 'tagname:value'. |
| `/*/key[a,*,c]?[(group="ABC" and tag="Tag1") or (group="DEF" and (tag="Tag2" or tag="Tag3:value"))]` | Соответствует похожим элементам данных любого узла сети из группы ABC или DEF с соответствующими тегами. |

Все указанные элементы данных должны существовать и собирать данные. В вычисления включаются только включенные элементы данных на включенных узлах сети. Элементы данных в неподдерживаемом состоянии не включаются.

**Attention:**
Если ключ элемента данных указанного элемента данных изменен, фильтр необходимо обновить вручную.

Указание родительской группы узлов сети включает родительскую группу и все вложенные группы узлов сети с их элементами данных.

Период времени

**Второй** параметр позволяет указать период времени для агрегации. Период времени может быть задан только как время; количество значений (с префиксом **#**) не поддерживается.

Для удобства в этом параметре можно использовать поддерживаемые символы единиц измерения, например, `5m` (пять минут) вместо `300s` (300 секунд) или `1d` (один день) вместо `86400` (86400 секунд).

Для функции `last_foreach` период времени является необязательным параметром (поддерживается начиная с Zabbix 7.0) и может быть опущен:

```
last_foreach(/*/key?[group="host group"])
```

Период времени не поддерживается функцией `exists_foreach`.

Дополнительные параметры

**bucket_rate_foreach**

Функция `bucket_rate_foreach` поддерживает третий необязательный параметр:

```
bucket_rate_foreach(item filter,time period,<parameter number>)
```

где `<parameter number>` — это позиция значения "bucket" в ключе элемента данных. Например, если значение "bucket" в `myItem[aaa,0.2]` равно `0.2`, то его позиция — 2.

Значение по умолчанию для <parameter number> — 1.

**count_foreach**

Функция `count_foreach` поддерживает третий и четвертый необязательные параметры:

```
count_foreach(item filter,time period,<operator>,<pattern>)
```

Где:

- **operator** — условный оператор для значений элементов данных (должен быть в двойных кавычках). Поддерживаемые operators:<br>*eq* - равно<br>*ne* - не равно<br>*gt* - больше<br>*ge* - больше или равно<br>*lt* - меньше<br>*le* - меньше или равно<br>*like* - совпадает, если содержит шаблон (с учетом регистра)<br>*bitand* - побитовое И<br>*regexp* - совпадение с регулярным выражением, заданным в `pattern`, с учетом регистра<br>*iregexp* - совпадение с регулярным выражением, заданным в `pattern`, без учета регистра.<br>
- **pattern** — требуемый шаблон (строковые аргументы должны быть в двойных кавычках); поддерживается, если в третьем параметре указан *operator*.

Примечания:

- Необязательные параметры *operator* или *pattern* нельзя оставлять пустыми после запятой, их можно только полностью опустить.

- При использовании *bitand* в качестве третьего параметра четвертый параметр `pattern` можно указать как два числа, разделенные '/': **number_to_compare_with/mask**. `count_foreach()` вычисляет "побитовое И" для значения и *mask* и сравнивает результат с *number_to_compare_with*. Если результат "побитового И" равен *number_to_compare_with*, значение учитывается.<br>Если *number_to_compare_with* и *mask* равны, достаточно указать только *mask* (без '/').
- При использовании *regexp* или *iregexp* в качестве третьего параметра четвертый параметр `pattern` может быть обычным или глобальным регулярным выражением (начинающимся с '@'). В случае глобальных регулярных выражений чувствительность к регистру наследуется из настроек глобального регулярного выражения. Для сопоставления regexp значения с плавающей точкой всегда будут представлены с 4 знаками после '.'. Также обратите внимание, что для больших чисел разница в десятичном (хранимом в базе данных) и двоичном (используемом сервером Zabbix) представлении может повлиять на 4-й десятичный знак.

См. агрегатные вычисления для получения дополнительных сведений и примеров использования функций foreach.

Поведение, зависящее от доступности

Следующая таблица иллюстрирует, как каждая функция ведёт себя в случае ограниченной доступности узла сети / элемента данных и данных истории.

| Функция | Узел сети деактивирован | Узел сети с данными недоступен | Узел сети без данных недоступен | Элемент данных деактивирован | Элемент данных не поддерживается | Ошибка извлечения данных (SQL) |
|---|---|---|---|---|---|---|
| *avg_foreach* | игнорировать | вернуть среднее | игнорировать | игнорировать | игнорировать | игнорировать |
| *bucket_rate_foreach* | игнорировать | вернуть bucket rate | игнорировать | игнорировать | игнорировать | игнорировать |
| *count_foreach* | игнорировать | вернуть количество | 0 | игнорировать | игнорировать | игнорировать |
| *exists_foreach* | игнорировать | 1 | 1 | игнорировать | 1 | n/a |
| *last_foreach* | игнорировать | вернуть последнее | игнорировать | игнорировать | игнорировать | игнорировать |
| *max_foreach* | игнорировать | вернуть максимум | игнорировать | игнорировать | игнорировать | игнорировать |
| *min_foreach* | игнорировать | вернуть минимум | игнорировать | игнорировать | игнорировать | игнорировать |
| *sum_foreach* | игнорировать | вернуть сумму | игнорировать | игнорировать | игнорировать | игнорировать |

Если для элемента данных указано *игнорировать*, к агрегированию ничего не добавляется.

**2 Побитовые функции**

Все функции, перечисленные здесь, поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

Функции перечислены без дополнительной информации. Щелкните по функции, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| bitand | Значение побитового AND значения элемента данных и маски. |
| bitlshift | Побитовый сдвиг значения элемента данных влево. |
| bitnot | Значение побитового NOT значения элемента данных. |
| bitor | Значение побитового OR значения элемента данных и маски. |
| bitrshift | Побитовый сдвиг значения элемента данных вправо. |
| bitxor | Значение побитового исключающего OR значения элемента данных и маски. |

Подробности функции

Некоторые общие замечания о параметрах функций:

- параметры функций разделяются запятыми
- в качестве параметров допустимы выражения

- необязательные параметры функций (или части параметров) обозначаются угловыми скобками < >

bitand(значение,маска)

Результат операции «побитовое И» значения элемента данных и маски.<br> Поддерживаемые типы значений: *Integer*.

Параметры:

- **значение** — проверяемое значение;
- **маска** (обязательно) — 64-битное целое без знака (0 - 18446744073709551615).

Несмотря на то, что сравнение выполняется побитово, все значения должны быть представлены и затем возвращаются в десятичном формате. Например, проверка третьего бита выполняется путём сравнения с числом 4, а не 100.

Примеры:

```
bitand(last(/  _  /  ),12)=8 or bitand(last(/host/key),12)=4 #                    ,
bitand(last(/  _  /  ),20)=16 #              ,       —
```

bitlshift(значение,сдвиг_битов)

Побитовый сдвиг значения элемента данных влево.<br> Поддерживаемые типы значений: *Integer*.

Параметры:

- **значение** — проверяемое значение;
- **сдвиг_битов** (обязательно) — количество битов для сдвига.

Несмотря на то, что сравнение выполняется побитово, все значения должны быть представлены и затем возвращаются в десятичном формате. Например, проверка третьего бита выполняется путём сравнения с числом 4, а не 100.

bitnot(значение)

Результат операции «побитовое НЕ» значения элемента данных.<br> Поддерживаемые типы значений: *Integer*.

Параметр:

- **значение** — проверяемое значение.

Несмотря на то, что сравнение выполняется побитово, все значения должны быть представлены и затем возвращаются в десятичном формате. Например, проверка третьего бита выполняется путём сравнения с числом 4, а не 100

bitor(значение,маска)

Результат операции «побитовое ИЛИ» значения элемента данных и маски.<br> Поддерживаемые типы значений: *Integer*.

Параметры:

- **значение** — проверяемое значение;
- **маска** (mandatory) — 64-битное целое без знака (0 - 18446744073709551615).

Несмотря на то, что сравнение выполняется побитово, все значения должны быть представлены и затем возвращаются в десятичном формате. Например, проверка третьего бита выполняется путём сравнения с числом 4, а не 100.

bitrshift(значение,сдвиг_битов)

Побитовый сдвиг значения элемента данных вправо.<br> Поддерживаемые типы значений: *Integer*.

Параметры:

- **значение** — проверяемое значение;
- **сдвиг_битов** (обязательно) — количество битов для сдвига.

Несмотря на то, что сравнение выполняется побитово, все значения должны быть представлены и затем возвращаются в десятичном формате. Например, проверка третьего бита выполняется путём сравнения с числом 4, а не 100.

bitxor(значение,маска)

Результат операции «побитовое исключающее ИЛИ» значения элемента данных и маски.<br> Поддерживаемые типы значений: *Integer*.

Параметры:

- **значение** — проверяемое значение;
- **маска** (обязательно) — 64-битное целое без знака (0 - 18446744073709551615).

Несмотря на то, что сравнение выполняется побитово, все значения должны быть представлены и затем возвращаются в десятичном формате. Например, проверка третьего бита выполняется путём сравнения с числом 4, а не 100.

См. все поддерживаемые функции.

#### 3 Функции даты и времени

Все функции, перечисленные здесь, поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

**Attention:**
Функции даты и времени нельзя использовать в выражении сами по себе; в выражение должна быть включена как минимум одна функция из другой группы, ссылающаяся на элемент данных узла сети (за исключением функции nodata()). Подробную информацию о том, как функции даты и времени работают в выражениях, см. в разделе Время вычисления.

Функции перечислены без дополнительной информации. Щелкните функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| date | Текущая дата в формате YYYYMMDD. |
| dayofmonth | День месяца в диапазоне от 1 до 31. |
| dayofweek | День недели в диапазоне от 1 до 7. |
| now | Количество секунд с начала эпохи (00:00:00 UTC, 1 января 1970 года). |
| time | Текущее время в формате HHMMSS. |

Подробности функций

date

Текущая дата в формате ГГГГММДД (YYYYMMDD).

Пример:

```
date()<20220101
```

dayofmonth

Число месяца в диапазоне от 1 до 31.

Пример:

```
dayofmonth()=1
```

dayofweek

День недели в диапазоне от 1 до 7 (Пн — 1, Вс — 7).

Пример (только рабочие дни):

```
dayofweek()<6
```

Пример (только выходные):

```
dayofweek()>5
```

now

Количество секунд с начала эпохи Unix (00:00:00 1 января 1970 года по Гринвичу).

Пример:

```
now()<1640998800
```

time

Текущее время в формате ЧЧММСС (HHMMSS).

Пример (только ночное время, 00:00—06:00):

```
time()<060000
```

См. все поддерживаемые функции.

**4 Функции истории**

Все функции, перечисленные здесь, поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

Функции перечислены без дополнительной информации. Нажмите на функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| change | Величина разницы между предыдущим и последним значением. |
| changecount | Количество изменений между соседними значениями в пределах заданного периода оценки. |
| count | Количество значений в пределах заданного периода оценки. |
| countunique | Количество уникальных значений в пределах заданного периода оценки. |
| find | Поиск совпадения значения в пределах заданного периода оценки. |
| first | Первое (самое старое) значение в пределах заданного периода оценки. |
| firstclock | Метка времени первого (самого старого) значения в пределах заданного периода оценки. |
| fuzzytime | Проверка, насколько время пассивного агента отличается от времени сервера/прокси Zabbix. |
| last | Самое последнее значение. |
| lastclock | Метка времени N-го самого последнего значения в пределах заданного периода оценки. |
| logeventid | Проверка, соответствует ли ID события последней записи журнала регулярному выражению. |
| logseverity | Уровень серьезности последней записи журнала. |
| logsource | Проверка, соответствует ли источник журнала последней записи журнала регулярному выражению. |
| logtimestamp | Метка времени сообщения журнала для N-го самого последнего значения элемента данных журнала. |
| monodec | Проверка, было ли монотонное уменьшение значений. |
| monoinc | Проверка, было ли монотонное увеличение значений. |
| nodata | Проверка на отсутствие полученных данных. |
| percentile | P-й процентиль периода, где P (процент) задается третьим параметром. |
| rate | Средняя скорость увеличения в секунду для монотонно возрастающего счетчика в пределах заданного периода времени. |

Общие параметры

- `/ _ /` является общим обязательным первым параметром для функций, ссылающихся на историю элемента данных узла сети
- `( |# )<: _ >` является общим вторым параметром для функций, ссылающихся на историю элемента данных узла сети, где:
  - **сек** — максимальный период вычисления в секундах (могут использоваться суффиксы времени), либо
  - **#число** — максимальная область вычисления среди последних собранных значений (если начинается со знака решётки)
  - **сдвиг_времени** (опционально) позволяет сдвигать точку вычислений по времени назад в прошлое. Смотрите более подробную информацию относительно того, как указывать сдвиг времени.

Подробности функций

Некоторые общие замечания о параметрах функций:

- параметры функций разделяются запятыми
- необязательные параметры функций (или части параметров) обозначаются угловыми скобками < >
- параметры, специфичные для функций, описываются с каждой функцией
- параметры `/ _ /` и `( |# )<: _ >` никогда не должны заключаться в кавычки

change(/host/key)

Величина разницы между предыдущим и последним значением.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.<br> Для строк возвращает: 0 - значения равны; 1 - значения различаются.

Параметры: см. общие параметры.

Комментарии:

- Числовая разница будет вычислена, как показано на следующих примерах входных значений ('previous' и 'latest' value = разница):<br>'1' и '5' = +4<br>'3' и '1' = −2<br>'0' и '-2.5' = −2.5.<br>
- См. также: abs для сравнения.

Примеры:

```
change(/host/system.uptime)<0 #                                          (                )
change(/host/system.cpu.load[all,avg1])>2 #      CPU (         )              2
change(/host/vfs.fs.size[/,free])<-1G #                                  1
```

changecount(/host/key,(sec|#num)<:time shift>,<mode>)

Количество изменений между соседними значениями в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- См. общие параметры<br>
- **mode** (должен быть заключен в двойные кавычки) - возможные значения: *all* - считать все изменения (по умолчанию); *dec* - считать уменьшения; *inc* - считать увеличения

Для нечисловых типов значений параметр *mode* игнорируется.

Примеры:

```
changecount(/host/icmpping,10m)>5 #      ping           5      10
changecount(/host/vfs.file.contents["/sys/class/net/eth0/operstate"],1h)>5 #             eth0          5
changecount(/host/proc.num[httpd],15m)>10 #            httpd                10    15
changecount(/host/key,#10,"inc") #                  (                 )          10
changecount(/host/key,24h,"dec") #                  (                 )         24
```

count(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Количество значений в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- См. общие параметры<br>
- **operator** (должен быть в двойных кавычках) Поддерживаемые operators:<br>*eq* - равно (по умолчанию для integer, float)<br>*ne* - не равно<br>*gt* - больше<br>*ge* - больше или равно<br>*lt* - меньше<br>*le* - меньше или равно<br>*like* (по умолчанию для string, text, log) - совпадает, если содержит шаблон (с учетом регистра)<br>*bitand* - побитовое И<br>*regexp* - совпадение с регулярным выражением, заданным в pattern, с учетом регистра<br>*iregexp* - совпадение с регулярным выражением, заданным в pattern, без учета регистра<br>
- **pattern** - требуемый шаблон (строковые аргументы должны быть в двойных кавычках)

Комментарии:

- Элементы данных типа Float сопоставляются с точностью 2.22e-16.
- *like* не поддерживается как оператор для значений integer.
- *like* и *bitand* не поддерживаются как операторы для значений float.
- Для значений string, text и log поддерживаются только операторы *eq*, *ne*, *like*, *regexp* и *iregexp*.
- При использовании *bitand* в качестве оператора четвертый параметр pattern можно указать как два числа, разделенные '/': **number_to_compare_with/mask**. count() вычисляет "побитовое И" значения и *mask* и сравнивает результат с *number_to_compare_with*. Если результат "побитового И" равен *number_to_compare_with*, значение учитывается.<br>Если *number_to_compare_with* и *mask* равны, достаточно указать только *mask* (без '/').
- При использовании *regexp* или *iregexp* в качестве оператора четвертый параметр pattern может быть обычным или глобальным регулярным выражением (начинающимся с '@'). В случае глобальных регулярных выражений чувствительность к регистру наследуется из настроек глобального регулярного выражения. Для сопоставления regexp значения float всегда будут представлены с 4 десятичными знаками после '.'. Также обратите внимание, что для больших чисел разница в десятичном (хранимом в базе данных) и двоичном (используемом сервером Zabbix) представлении может повлиять на 4-ю десятичную цифру.

Примеры:

```
count(/host/icmpping,30m,,"0")>5 #ping          5      30
count(/host/key,10m,"like","error") #                    10                ,        'error'
count(/host/key,10m,,12) #                    10                ,      '12'
count(/host/key,10m,"gt",12) #                    10                ,          '12'
```

```
count(/host/key,#10,"gt",12) #                    10                ,          '12'
count(/host/key,10m:now-1d,"gt",12) #              24       10       24                  ,
count(/host/key,10m,"bitand","6/7") #                  10                ,      '110' (         )   3
count(/host/key,10m:now-1d) #              24       10       24
```

countunique(/узел_сети/ключ,(сек|#число)<:сдвиг_времени>,<оператор>,<шаблон>)

Количество уникальных значений за определённый период вычисления.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- смотрите общие параметры<br>
- **оператор** (должен быть в двойных кавычках). Поддерживаются :<br>*eq* — равно (по умолчанию для integer и float)<br>*ne* — не равно<br>*gt* — больше<br>*ge* — больше или равно<br>*lt* — меньше<br>*le* — меньше или равно<br>*like* (по умолчанию для элементов данных символ, текст и журнал (лог)) — содержит подстроку шаблона (чувствительно к регистру)<br>*bitand* — побитовое И<br>*regexp* — чувствительное к регистру соответствие регулярному выражению, данному в <br>*iregexp* — нечувствительное к регистру соответствие регулярному выражению, данному в <br>
- **шаблон** — искомый шаблон (строковые аргументы должны быть в двойных кавычках)

Комментарии:

- Элементы данных с плавающей точкой совпадают с точностью 2.22e-16.
- *like* в качестве оператора не поддерживается для целочисленных значений.
- Для значений с типами символ, текст и журнал (лог) поддерживаются только операторы *eq*, *ne*, *like*, *regexp* и *iregexp*.
- Для оператора *bitand* четвёртый параметр может быть указан как два числа, разделённых косой чертой («/»): **число_для_сравнения/маска**. countunique() вычисляет «побитовое И» из значения и *маски* и сравнивает результат с *числом_для_сравнения*. Если результат «побитового И» равен *числу_для_сравнения*, то значение считается.<br>Если *число_для_сравнения* и *маска* равны, требуется указывать только *маску* (без «/»).
- Для операторов *regexp* и *iregexp* четвёртый параметр может быть обычным или глобальным (начинающимся с «@») регулярным выражением. В случае глобальных регулярных выражений чувствительность к регистру наследуется из настроек глобального регулярного выражения. При проверке соответствия регулярному выражению, значения с плавающей точкой всегда будут представлены с 4 цифрами после десятичной точки («.»). Также обратите внимание, что для больших чисел различие между десятичным (хранящимся в базе данных) и двоичным (используемом сервером Zabbix) представлениями может повлиять на 4-ю десятичную цифру.

Примеры:

```
countunique(/  _  /  ,10m) #                    10
countunique(/  _  /  ,10m,"like","error") #                  10              ,          «error»
countunique(/  _  /  ,10m,,12) #                  10              ,        «12»
countunique(/  _  /  ,10m,"gt",12) #                  10              ,        «12»
countunique(/  _  /  ,#10,"gt",12) #                  10                    ,          «12»
countunique(/  _  /  ,10m:now-1d,"gt",12) #                    (24      10       )   (24       )
countunique(/  _  /  ,10m,"bitand","6/7") #                  10              ,
countunique(/  _  /  ,10m:now-1d) #                    (24      10       )   (24       )
```

find(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Находит совпадение значения в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.<br> Возвращает: 1 - найдено; 0 - в противном случае.

Параметры:

- См. общие параметры<br>
- **sec** или **#num** (необязательно) - по умолчанию используется последнее значение, если не указано
- **operator** (должен быть в двойных кавычках) Поддерживаемые `operators`:<br>*eq* - равно (по умолчанию для integer, float)<br>*ne* - не равно<br>*gt* - больше<br>*ge* - больше или равно<br>*lt* - меньше<br>*le* - меньше или равно<br>*like* (по умолчанию для string, text, log) - совпадает, если содержит строку, заданную в `pattern` (с учетом регистра)<br>*bitand* - побитовое И<br>*regexp* - совпадение с регулярным выражением, заданным в `pattern`, с учетом регистра<br>*iregexp* - совпадение с регулярным выражением, заданным в `pattern`, без учета регистра<br>
- **pattern** - требуемый шаблон (строковые аргументы должны быть в двойных кавычках); регулярное выражение Perl Compatible Regular Expression (PCRE), если `operator` - *regexp*, *iregexp*

Комментарии:

- Если обрабатывается более одного значения, возвращается '1', если найдено хотя бы одно совпадающее значение.
- *like* не поддерживается как оператор для значений integer.

- *like* и *bitand* не поддерживаются как операторы для значений float.
- Для значений string, text и log поддерживаются только операторы *eq*, *ne*, *like*, *regexp* и *iregexp*.
- При использовании *regexp* или *iregexp* в качестве оператора четвертый параметр `pattern` может быть обычным или глобальным регулярным выражением (начинающимся с '@'). В случае глобальных регулярных выражений чувствительность к регистру наследуется из настроек глобального регулярного выражения.

Примеры:

```
find(/host/key,10m,"like","error") #          ,        'error',          10
find(/host/agent.version,,"like","beta")=1 #    Zabbix     beta-    ,
find(/host/log[/var/log/nginx/access.log],,"regexp"," 500 ")=1 #                    -
```

first(/узел_сети/ключ,сек<:сдвиг_времени>)

Первое (самое старое) значение за указанный период вычисления.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- смотрите общие параметры

Смотрите также last().

Пример:

```
first(/  _  /  ,1h) #
```

firstclock(/host/key,sec<:time shift>)

Временная метка самого старого значения в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- См. common parameters.

Функция завершается с ошибкой, если за указанный период не было собрано данных.

См. также lastclock().

Примеры:

```
firstclock(/host/key,1h) #
firstclock(/host/key,1h:now-24h) #
```

fuzzytime(/host/key,sec)

Проверяет, насколько время пассивного агента отличается от времени сервера/прокси Zabbix.<br> Поддерживаемые типы значений: *Float*, *Integer*.<br> Возвращает: 1 - разница между значением пассивного элемента данных (как временной меткой) и временной меткой сервера/прокси Zabbix (временем сбора значения) меньше или равна *sec* секундам; 0 - в противном случае.

Параметры:

- См. common parameters.

Комментарии:

- Обычно используется с элементом данных 'system.localtime', чтобы проверить, что локальное время синхронизировано с локальным временем сервера Zabbix. *Обратите внимание*, что 'system.localtime' должен быть настроен как пассивная проверка для агента Zabbix; в Zabbix agent 2 он может быть настроен как активная проверка.
- Также может использоваться с ключом `vfs.file.time[/path/file,modify]`, чтобы проверить, что файл долгое время не обновлялся.
- Не рекомендуется использовать эту функцию в сложных выражениях триггеров (с участием нескольких элементов данных), поскольку это может привести к неожиданным результатам (разница во времени будет измеряться по самой последней метрике), например: `fuzzytime(/Host/system.localtime,60s)=0 or last(/Host/trap)<>0`.

Примеры:

```
fuzzytime(/host/system.localtime,5m)=0 #                                /    Zabbix         5
fuzzytime(/host/system.localtime,5m)=0 and nodata(/host/system.localtime,10m)=0 #
```

last(/host/key,<#num<:time shift»)

Самое последнее значение.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- См. общие параметры<br>
- **#num** (необязательно) - N-е по счету самое последнее значение

Комментарии:

- Обратите внимание, что период времени с хэштегом (#N) здесь работает иначе, чем во многих других функциях. Например: `last(/host/key)` всегда равно `last(/host/key,#1)`; `last(/host/key,#3)` - третье по счету самое последнее значение (*а не* три последних значения).
- Zabbix не гарантирует точный порядок значений, если в истории за одну секунду существует более двух значений.
- См. также first().

Примеры:

```
last(/host/key) #
last(/host/key,#2) #
last(/host/key,#1)<>last(/host/key,#2) #
```

lastclock(/host/key,<#num<:time shift»)

Метка времени N-го самого недавнего значения в пределах заданного периода оценки.<br> Поддерживаемые типы значений: *Float*, *Integer*, *String*, *Text*, *Log*.

Параметры:

- См. общие параметры;<br>
- **#num** (необязательно) - N-е самое недавнее значение.

Функция завершается с ошибкой, если за указанный период не было собрано ни одного данных или не было собрано N-е значение.

См. также firstclock().

Примеры:

```
lastclock(/host/key) #
lastclock(/host/key,#2) #              2-
now()-lastclock(/host/heartbeat.item)>300 #                        heartbeat         5
lastclock(/host/system.cpu.load[all,avg1])-lastclock(/host/system.cpu.load[all,avg1],#2)>300 #
((now()-lastclock(/host/system.cpu.load[all,avg1])<120 and last(/host/system.cpu.load[all,avg1])>5)) #
```

logeventid(/host/key,<#num<:time shift»,<pattern>)

Проверяет, соответствует ли идентификатор события последней записи журнала регулярному выражению.<br> Поддерживаемые типы значений: *Log*.<br> Возвращает: 0 - не соответствует; 1 - соответствует.

Параметры:

- См. общие параметры<br>
- **#num** (необязательно) - N-е самое недавнее значение<br>
- **pattern** (необязательно) - регулярное выражение, описывающее требуемый шаблон, в стиле Perl Compatible Regular Expression (PCRE) (строковые аргументы должны быть заключены в двойные кавычки)

Примеры:

```
logeventid(/host/eventlog[Security],,"^4625$")=1 #                        ,          "4625" (
logeventid(/host/eventlog[System],,"^(6008|41)$")=1 #                        ,          "6008"    "41"
```

logseverity(/host/key,<#num<:time shift»)

Уровень серьезности последней записи журнала.<br> Поддерживаемые типы значений: *Log*.<br> Возвращает: 0 - уровень серьезности по умолчанию; N - уровень серьезности (целое число, полезно для журналов событий Windows: 1 - Information, 2 - Warning, 4 - Error, 7 - Failure Audit, 8 - Success Audit, 9 - Critical, 10 - Verbose).

Параметры:

- См. общие параметры<br>
- **#num** (необязательно) - N-е по счету самое недавнее значение

Zabbix берет уровень серьезности журнала из поля **Information** журнала событий Windows.

Примеры:

```
logseverity(/host/log[/var/log/syslog],10m)>3 #                                  "3"
logseverity(/host/eventlog[System],10m)=4 #                              ,     "Error"
```

logsource(/host/key,<#num<:time shift»,<pattern>)

Проверяет, соответствует ли источник последней записи журнала регулярному выражению.<br> Поддерживаемые типы значений: *Log*.<br> Возвращает: 0 - не соответствует; 1 - соответствует.

Параметры:

- См. общие параметры<br>
- **#num** (необязательно) - N-е по счету самое недавнее значение<br>
- **pattern** (необязательно) - регулярное выражение, описывающее требуемый шаблон, в стиле Perl Compatible Regular Expression (PCRE) (строковые аргументы должны быть заключены в двойные кавычки)

Обычно используется для журналов событий Windows.

Примеры:

```
logsource(/host/eventlog[Application],,"MSSQLSERVER")=1 #             ,                  "MSSQLSERVER"
logsource(/host/eventlog[System],,"Service Control Manager")=1 #             ,                  "Service
logsource(/host/eventlog[System],,"Service Control Manager")=1 and logeventid(/host/eventlog[System],,"^70
```

logtimestamp(/host/key,<#num<:time shift»)

Временная метка сообщения журнала для N-го по новизне значения элемента данных типа log.<br> Поддерживаемые типы значений: *Log*.

Параметры:

- См. общие параметры;<br>
- **#num** (необязательно) - N-е по новизне значение.

Комментарии:

- Вычисление сдвига времени основано на часах значения элемента данных, а не на временной метке сообщения журнала;
- Функция завершается с ошибкой, если:
  - получен элемент данных не типа log;
  - за указанный период не было собрано данных;
  - не было собрано N-е значение;
  - сообщение журнала не содержит значения временной метки.

Примеры:

```
logtimestamp(/host/key) #
logtimestamp(/host/key,#2) #
logtimestamp(/host/key,#3:now-1d) #
```

monodec(/host/key,(sec|#num)<:time shift>,<mode>)

Проверяет, было ли монотонное уменьшение значений.<br> Поддерживаемые типы значений: *Integer*.<br> Возвращает: 1 - если все элементы в период времени непрерывно уменьшаются; 0 - в противном случае.

Параметры:

- См. общие параметры<br>
- **mode** (должен быть в двойных кавычках) - *weak* (каждое значение меньше предыдущего или равно ему; по умолчанию) или *strict* (каждое значение уменьшилось)

Примеры:

```
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 #the number of nginx processes has monotonously decreased ov
```

monoinc(/host/key,(sec|#num)<:сдвиг времени>,<mode>)

Проверяет, было ли монотонное увеличение значений.<br> Поддерживаемые типы значений: *Integer*.<br> Возвращает: 1 - если все элементы в период времени непрерывно увеличиваются; 0 - в противном случае.

Параметры:

- См. общие параметры<br>

- **mode** (должен быть в двойных кавычках) - *weak* (каждое значение больше или равно предыдущему; по умолчанию) или *strict* (каждое значение строго больше предыдущего)

Примеры:

```
monoinc(/host/system.localtime,#3,"strict")=0 #
monoinc(/host/vfs.dir.count[/mnt/data/logs],24h,"weak")=0 #     ,                              24     (
```

nodata(/host/key,sec,<mode>)

Проверка отсутствия полученных данных.<br> Поддерживаемые типы значений: *Integer*, *Float*, *Character*, *Text*, *Log*.<br> Возвращает: 1 - если данные не поступали в течение заданного периода времени; 0 - в противном случае.

Параметры:

- См. общие параметры<br>
- **sec** - период не должен быть меньше 30 секунд, поскольку процесс history syncer вычисляет эту функцию только каждые 30 секунд; `nodata(/host/key,0)` запрещено
- **mode** - если задано значение *strict* (в двойных кавычках), эта функция не будет зависеть от доступности прокси (см. комментарии для подробностей)

Комментарии:

- триггеры 'nodata', контролируемые через прокси, по умолчанию чувствительны к доступности прокси - если прокси становится недоступен, триггеры 'nodata' не сработают сразу после восстановления соединения, а пропустят данные за задержанный период. Обратите внимание, что для пассивных прокси подавление активируется, если соединение восстановлено более чем через 15 секунд и не менее чем через 2 секунды. Для активных прокси подавление активируется, если соединение восстановлено более чем через 15 секунд. Чтобы отключить чувствительность к доступности прокси, используйте третий параметр, например: `nodata(/host/key,5m,"strict")`; в этом случае функция сработает, как только истечет период оценки (пять минут) без данных.<br>
- Эта функция будет отображать ошибку, если в течение периода, заданного 1-м параметром:<br>- нет данных и сервер Zabbix был перезапущен<br>- нет данных и обслуживание было завершено<br>- нет данных и элемент данных был добавлен или повторно включен.<br>
- Ошибки отображаются в столбце *Info* в конфигурации триггера.<br>
- Эта функция может работать некорректно, если между сервером Zabbix, прокси и агентом есть различия во времени. См. также: Требование к синхронизации времени.

Пример:

```
nodata(/host/agent.ping,5m)=1 #     ,           Zabbix                    5
```

percentile(/host/key,(sec|#num)<:сдвиг времени>,percentage)

P-й процентиль периода, где P (процент) задается третьим параметром.<br> Поддерживаемые типы значений: *Float*, *Integer*.

Параметры:

- См. общие параметры<br>
- **percentage** - число с плавающей точкой от 0 до 100 включительно, с точностью до 4 знаков после десятичной точки

Примеры:

```
percentile(/host/net.if.in[eth0,bytes],1h,95)>1000000 #95-                          (   / )    1            (
percentile(/host/system.cpu.util,5m,95)>80 #95-                                    CPU       80
percentile(/host/icmppingsec[192.168.0.2],15m,95)>0.15 #                        150   ,                    (
percentile(/host/net.if.in[eth0,bytes],1h,50) #      50-         (           )                       ;
(percentile(/host/net.if.in[eth0,bytes],1h,50)+percentile(/host/net.if.in[eth0,bytes],1h,51))/2 #
```

rate(/host/key,sec<:time shift>)

Средняя скорость увеличения монотонно возрастающего счетчика за секунду в пределах заданного периода времени.<br> Поддерживаемые типы значений: *Float*, *Integer*.

Параметры:

- См. общие параметры

Функционально соответствует 'rate' в PromQL.

Примеры:

```
rate(/host/key,30s) #               30           20,              0.67.
rate(/host/net.if.in[eth0,bytes],5m)>500000 #                     eth0       500  /         5
rate(/host/app.requests.count,1m)>100 #                       >100
```

См. все поддерживаемые функции.

#### 5 Функции трендов

Функции трендов, в отличие от функций истории, используют данные трендов для вычислений.

Тренды хранят почасовые агрегированные значения. Функции трендов используют эти почасовые средние значения и поэтому полезны для долгосрочного анализа.

Результаты функций трендов кэшируются, поэтому несколько вызовов одной и той же функции с одинаковыми параметрами получают информацию из базы данных только один раз. Кэш функций трендов управляется параметром сервера TrendFunctionCacheSize.

Триггеры, которые ссылаются **только** на функции трендов, вычисляются один раз за наименьший временной период в выражении. Например, триггер вида

```
trendavg(/host/key,1d:now/d) > 1 or trendavg(/host/key2,1w:now/w) > 2
```

будет вычисляться один раз в день. Если триггер содержит как функции трендов, так и функции истории (или даты и времени и/или nodata()), он вычисляется в соответствии с обычными принципами.

Все перечисленные здесь функции поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

Функции перечислены без дополнительной информации. Нажмите на функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| baselinedev | Возвращает число отклонений (по алгоритму stddevpop) между последним периодом данных и теми же периодами данных в предыдущих сезонах. |
| baselinewma | Вычисляет базовую линию путем усреднения данных из того же временного интервала в нескольких равных временных периодах (’сезонах’) с использованием алгоритма взвешенного скользящего среднего. |
| trendavg | Среднее значение трендов в пределах заданного периода времени. |
| trendcount | Количество успешно полученных значений истории, использованных для вычисления значения тренда в пределах заданного периода времени. |
| trendmax | Максимальное значение трендов в пределах заданного периода времени. |
| trendmin | Минимальное значение трендов в пределах заданного периода времени. |
| trendstl | Возвращает уровень аномалий за период обнаружения - десятичное значение от 0 до 1, равное `((the number of anomaly values)/(total number of values))`. |
| trendsum | Сумма значений трендов в пределах заданного периода времени. |

Общие параметры

- / _ / является общим обязательным первым параметром
- _ : _ является общим вторым параметром, где:
  - **период_времени** — период времени (минимум «1h»), определённый как <N><единица_времени>, где N — количество единиц времени, _ — h (час), d (сутки), w (неделя), M (месяц) или y (год).
  - **сдвиг_времени** — смещение периода времени (смотрите примеры функций)

Подробности функций

Некоторые общие замечания о параметрах функций:

- параметры функций разделяются запятыми
- необязательные параметры функций (или части параметров) обозначаются угловыми скобками < >
- параметры, специфичные для функций, описываются с каждой функцией
- параметры / _ / и _ : _ никогда не должны заключаться в кавычки

baselinedev(/узел_сети/ключ,период_данных:сдвиг_времени,сезон_ед,сезон_кол-во)

Возвращает количество стандартных отклонений (по алгоритму stddevpop, среднеквадратичное отклонение) между последним периодом данных и теми же периодами данных в предыдущие сезоны.<br>

Параметры:

- Смотрите общие параметры<br>
- **период_данных** — период сбора данных в течение сезона, определённый как <N><единица_времени>, где<br>N — количество единиц времени<br> _ — h (час), d (сутки), w (неделя), M (месяц) или y (год), должно быть меньше или равно сезону<br>
- **сезон_ед** — календарная единица, определяющая один сезон (h, d, w, M, y); не может быть меньше периода данных<br>
- **сезон_кол-во** — количество оцениваемых сезонов

Примеры:

```
baselinedev(/  _  /  ,1d:now/d,"M",6) #                    (          )
baselinedev(/  _  /  ,1h:now/h,"d",10)  #
```

baselinewma(/узел_сети/ключ,период_данных:сдвиг_времени,сезон_ед,сезон_кол-во)

Вычисляет базовый уровень путем усреднения данных за один и тот же интервал времени в течение нескольких равных периодов времени («сезонов») с использованием алгоритма взвешенного скользящего среднего.<br>

Параметры:

- Смотрите общие параметры<br>
- **период_данных** — период сбора данных в течение сезона, определённый как <N><единица_времени>, где:<br>N — количество единиц времени<br> _ — h (час), d (сутки), w (неделя), M (месяц) или y (год), должно быть меньше или равно сезону<br>Сдвиг_времени] — смещение периода времени, определяет конец интервала сбора данных в сезонах (смотрите примеры)<br>
- **сезон_ед** — календарная единица, определяющая однин сезон (h, d, w, M, y); не может быть меньше периода данных<br>
- **сезон_кол-во** — количество оцениваемых сезонов

Примеры:

```
baselinewma(/  _  /  ,1h:now/h,"d",3) #                                  ,          .
baselinewma(/  _  /  ,2h:now/h,"d",3) #                                           ,
baselinewma(/  _  /  ,1d:now/d,"M",4) #                    ,     «  »,      4    ,
```

trendavg(/узел_сети/ключ,период_времени:сдвиг_времени)

Среднее из значений динамики изменений за определённый период времени.

Параметры:

- Смотрите общие параметры

Примеры:

```
trendavg(/  _  /  ,1h:now/h) #              (    , 12:00-13:00)
trendavg(/  _  /  ,1h:now/h-1h) #             (11:00-12:00)
trendavg(/  _  /  ,1h:now/h-2h) #             (10:00-11:00)
trendavg(/  _  /  ,1M:now/M-1y) #
```

trendcount(/узел_сети/ключ,период_времени:сдвиг_времени)

Количество успешно извлечённых значений истории, используемых для расчёта значений динамики изменений за определённый период времени.

Параметры:

- Смотрите общие параметры

Примеры:

```
trendcount(/  _  /  ,1h:now/h) #              (    , 12:00-13:00)
trendcount(/  _  /  ,1h:now/h-1h) #             (11:00-12:00)
trendcount(/  _  /  ,1h:now/h-2h) #             (10:00-11:00)
trendcount(/  _  /  ,1M:now/M-1y) #
```

trendmax(/узел_сети/ключ,период_времени:сдвиг_времени)

Максимальное значение из динамики изменений за определённый период времени.

Параметры:

- Смотрите общие параметры

Примеры:

```
trendmax(/  _  /  ,1h:now/h) #                (    , 12:00-13:00)
trendmax(/  _  /  ,1h:now/h) - trendmin(/  _  /  ,1h:now/h) →                                (
trendmax(/  _  /  ,1h:now/h-1h) #                  (11:00-12:00)
trendmax(/  _  /  ,1h:now/h-2h) #                  (10:00-11:00)
trendmax(/  _  /  ,1M:now/M-1y) #
```

trendmin(/узел_сети/ключ,период_времени:сдвиг_времени)

Минимальное значение из динамики изменений за определённый период времени.

Параметры:

- Смотрите общие параметры

Примеры:

```
trendmin(/  _  /  ,1h:now/h) #                (    , 12:00-13:00)
trendmax(/  _  /  ,1h:now/h) - trendmin(/  _  /  ,1h:now/h) →                                (
trendmin(/  _  /  ,1h:now/h-1h) #                  (11:00-12:00)
trendmin(/  _  /  ,1h:now/h-2h) #                  (10:00-11:00)
trendmin(/  _  /  ,1M:now/M-1y) #
```

trendstl(/узел_сети/ключ,период_оценки:сдвиг_времени,период_обнаружения,сезон,<отклонения>,<devalg>,<s_window>)

Возвращает частоту аномалий за период обнаружения — десятичное значение от 0 до 1, которое определяется как (( )/( )).

Параметры:

- Смотрите общие параметры<br>
- **период_оценки** — период времени, который необходимо разложить (минимум «1h»), определённый как <N><единица_времени>, где<br>N — количество единиц времени<br> _ — h (час), d (сутки), w (неделя), M (месяц) или y (год)<br>
- **период_обнаружения** — период времени до конца периода оценки, для которого вычисляются аномалии (минимум «1h», не может быть дольше периода оценки), определённый как <N><единица_времени>, где<br>N — количество единиц времени<br> _ — h (час), d (сутки), w (неделя)<br>
- **сезон** — самый короткий период времени, где ожидается повторяющийся шаблон («сезон»), (минимум «2h», не может быть дольше периода оценки, количество записей в периоде оценки должно быть больше двукратной результирующей частоты (сезон/h)), определённой как <N><единица_времени>, где<br>N — количество единиц времени<br> _ — h (час), d (сутки), w (неделя)
- **отклонения** — количество отклонений (вычисленных по алгоритму devalg), которые следует считать аномалией (может быть десятичным числом), (должно быть больше либо равно 1, по умолчанию: 3);
- **devalg** (должно быть в двойных кавычках) — алгоритм отклонений, может быть *stddevpop*, *stddevsamp* или *mad* (по умолчанию)
- **s_window** — протяжённость (в лагах) окна LOESS для извлечения сезонов (по умолчанию: 10 * количество записей в периоде оценки + 1)

Примеры:

```
trendstl(/  _  /  ,100h:now/h,10h,2h) #              100                  ,                      10
trendstl(/  _  /  ,100h:now/h-10h,100h,2h,2.1,"mad") #            100                    ,        10
trendstl(/  _  /  ,100d:now/d-1d,10d,1d,4,,10) #            100                                 ,
trendstl(/  _  /  ,1M:now/M-1y,1d,2h,,"stddevsamp") #                        ,
```

trendsum(/узел_сети/ключ,период_времени:сдвиг_времени)

Сумма значений динамики изменений за определённый период времени.

Параметры:

- Смотрите общие параметры

Примеры:

```
trendsum(/  _  /  ,1h:now/h) #              (     , 12:00-13:00)
trendsum(/  _  /  ,1h:now/h-1h) #            (11:00-12:00)
trendsum(/  _  /  ,1h:now/h-2h) #            (10:00-11:00)
trendsum(/  _  /  ,1M:now/M-1y) #
```

См. все поддерживаемые функции.

#### 6 Математические функции

Все функции, перечисленные здесь, поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

Математические функции поддерживаются для типов значений float и integer, если не указано иное.

Функции перечислены без дополнительной информации. Щелкните функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| abs | Абсолютное значение значения. |
| acos | Арккосинус значения как угол, выраженный в радианах. |
| asin | Арксинус значения как угол, выраженный в радианах. |
| atan | Арктангенс значения как угол, выраженный в радианах. |
| atan2 | Арктангенс координат ординаты (значения) и абсциссы, заданных как угол, выраженный в радианах. |
| avg | Среднее значение ссылочных значений элементов данных. |
| cbrt | Кубический корень значения. |
| ceil | Округлить значение вверх до ближайшего большего или равного целого числа. |
| cos | Косинус значения, где значение — это угол, выраженный в радианах. |
| cosh | Гиперболический косинус значения. |
| cot | Котангенс значения, где значение — это угол, выраженный в радианах. |
| degrees | Преобразует значение из радианов в градусы. |
| e | Число Эйлера (2.718281828459045). |
| exp | Число Эйлера в степени значения. |
| expm1 | Число Эйлера в степени значения минус 1. |
| floor | Округлить значение вниз до ближайшего меньшего или равного целого числа. |
| log | Натуральный логарифм. |
| log10 | Десятичный логарифм. |
| max | Наибольшее значение ссылочных значений элементов данных. |
| min | Наименьшее значение ссылочных значений элементов данных. |
| mod | Остаток от деления. |
| pi | Константа Pi (3.14159265358979). |
| power | Степень значения. |
| radians | Преобразует значение из градусов в радианы. |
| rand | Возвращает случайное целое значение. |
| round | Округлить значение до указанного числа знаков после запятой. |
| signum | Возвращает '-1', если значение отрицательное, '0', если значение равно нулю, '1', если значение положительное. |
| sin | Синус значения, где значение — это угол, выраженный в радианах. |
| sinh | Гиперболический синус значения, где значение — это угол, выраженный в радианах. |
| sqrt | Квадратный корень значения. |
| sum | Сумма ссылочных значений элементов данных. |
| tan | Тангенс значения. |
| truncate | Обрезать значение до указанного числа знаков после запятой. |

Подробности функций

Некоторые общие замечания о параметрах функций:

- параметры функций разделяются запятыми
- в качестве параметров допустимы выражения
- необязательные параметры функций (или части параметров) обозначаются угловыми скобками < >

abs(значение)

Абсолютное значение (по модулю) значения.

Параметр:

- **значение** — проверяемое значение

Например, абсолютное значение как «3», так и «-3» будет «3».

Пример:

```
abs(last(/  _  /  ))>10
```

acos(значение)

Арккосинус значения как угол, выраженный в радианах.

Параметр:

- **значение** — проверяемое значение

Значение должно быть в диапазоне от -1 до 1. Например, арккосинус значения «0.5» будет «2.0943951».

Пример:

```
acos(last(/  _  /  ))
```

asin(значение)

Арксинус значения как угол, выраженный в радианах.

Параметр:

- **значение** — проверяемое значение

Значение должно быть в диапазоне от -1 до 1. Например, арксинус значения «0.5» будет «-0.523598776».

Пример:

```
asin(last(/  _  /  ))
```

atan(значение)

Арктангенс значения как угол, выраженный в радианах.

Параметр:

- **значение** — проверяемое значение

Например, арктангенс значения «1» будет «0.785398163».

Пример:

```
atan(last(/  _  /  ))
```

atan2(значение,абсцисса)

Арктангенс указанных координат ординаты (значение) и абсцисса как угол, выраженный в радианах.

Параметры:

- **значение** — проверяемое значение;
- **абсцисса** — значение абсциссы.

Например, арктангенс координат, где ордината и абсцисса имеют значение «1», будет «2.21429744».

Пример:

```
atan2(last(/  _  /  ),2)
```

avg(<значение1>,<значение2>,...)

Среднее из значений, на которые ссылаются перечисленные элементы данных.

Параметр:

- **значениеX** — значение, возвращаемое другой функцией, работающей с историей элементов данных.

Пример:

```
avg(avg(/  _  /  ,1h),avg(/  _  2/  2,1h))
```

cbrt(значение)

Кубический корень из значения.

Параметр:

- **значение** — проверяемое значение

Например, кубический корень из «64» будет «4», а из «63» будет «3.97905721».

Пример:

```
cbrt(last(/  _  /  ))
```

ceil(значение)

Округлить значение вверх до ближайшего целого.

Параметр:

- **значение** — проверяемое значение

Например, «2.4» будет округлено до «3». Смотрите также floor().

Пример:

```
ceil(last(/  _  /  ))
```

cos(значение)

Косинус значения, где значение воспринимается как угол, выраженный в радианах.

Параметр:

- **значение** — проверяемое значение

Например, косинус значения «1» будет «0.54030230586».

Пример:

```
cos(last(/  _  /  ))
```

cosh(значение)

Гиперболический косинус значения. Возвращает значение как действительное число, не в научной нотации.

Параметр:

- **значение** — проверяемое значение

Например, гиберболический косинус значения «1» будет «1.54308063482».

Пример:

```
cosh(last(/  _  /  ))
```

cot(значение)

Котангенс значения, где значение воспринимается как угол, выраженный в радианах.

Параметр:

- **значение** — проверяемое значение

Например, котангенс значения «1» будет «0.54030230586».

Пример:

```
cot(last(/  _  /  ))
```

degrees(значение)

Конвертация значения из радиан в градусы.

Параметр:

- **значение** — проверяемое значение

Например, значение «1», сконвертированное в градусы, будет «57.2957795».

Пример:

```
degrees(last(/  _  /  ))
```

e

Число Эйлера (2.718281828459045).

Пример:

```
e()
```

exp(значение)

Число Эйлера (e) в степени значения.

Параметр:

- **значение** — проверяемое значение

Например, число Эйлера, возведённое в степень «2», будет «7.38905609893065».

Пример:

```
exp(last(/  _  /  ))
```

expm1(значение)

Число Эйлера (e) в степени значения, минус один.

Параметр:

- **значение** — проверяемое значение

Например, число Эйлера, возведённое в степень «2», после вычитания единицы будет «6.38905609893065».

Пример:

```
expm1(last(/  _  /  ))
```

floor(значение)

Округление вниз до ближайшего целого.

Параметр:

- **значение** — проверяемое значение

Например, «2.6» будет округлено вниз до «2». Смотрите также ceil().

Пример:

```
floor(last(/  _  /  ))
```

log(значение)

Натуральный логарифм.

Параметр:

- **значение** — проверяемое значение

Например, натуральный логарифм значения «2» будет «0.69314718055994529».

Пример:

```
log(last(/  _  /  ))
```

log10(значение)

Десятичный логарифм.

Параметр:

- **значение** — проверяемое значение

Например, десятичный логарифм значения «5» will be «0.69897000433».

Пример:

```
log10(last(/  _  /  ))
```

max(<значение1>,<значение2>,...)

Наибольшее из значений, на которые ссылаются перечисленные элементы данных.

Параметр:

- **значениеX** — значение, возвращаемое другой функцией, работающей с историей элементов данных.

Пример:

```
max(avg(/  _  /  ,1h),avg(/  _  2/  2,1h))
```

min(<значение1>,<значение2>,...)

Наименьшее из значений, на которые ссылаются перечисленные элементы данных.

Параметр:

- **значениеX** — значение, возвращаемое другой функцией, работающей с историей элементов данных.

Пример:

```
min(avg(/  _  /  ,1h),avg(/  _  2/  2,1h))
```

mod(значение,делитель)

Остаток от деления.

Параметры:

- **значение** — проверяемое значение;
- **делитель** — знаменатель деления.

Например, остаток от деления значения «5» с делителем «2» будет «1».

Пример:

```
mod(last(/  _  /  ),2)
```

pi

Константа пи (3.14159265358979).

Пример:

```
pi()
```

power(значение,степень)

Возведение в степень.

Параметры:

- **значение** — проверяемое значение;
- **степень** — используемый показатель степени.

Например, третья степень значения «2» будет «8».

Пример:

```
power(last(/  _  /  ),3)
```

radians(значение)

Конвертация значения из градусов в радианы.

Параметр:

- **значение** — проверяемое значение

Например, значение «1», сконвертированное в радианы, будет «0.0174532925».

Пример:

```
radians(last(/  _  /  ))
```

rand

Вернуть случайное целое число. Псевдослучайное число, сгенерированное с использованием времени в качестве seed (достаточно для математичеких целей, но не для криптографии).

Пример:

```
rand()
```

round(значение,дес_знаков)

Округлить значение до десятичных знаков.

Параметры:

- **значение** — проверяемое значение;
- **дес_знаков** — количество десятичных знаков для округления (0 также допустим).

Например, значение «2.5482», округлённое до 2 десятичных знаков, будет «2.55».

Пример:

```
round(last(/  _  /  ),2)
```

signum(значение)

Возвращает «-1» для отрицательного значения, «0» для нуля, «1» для положительного значения.

Параметр:

- **значение** — проверяемое значение.

Пример:

```
signum(last(/  _  /  ))
```

sin(значение)

Синус значения, воспринимаемого как угол, выраженный в радианах.

Параметр:

- **значение** — проверяемое значение

Например, синус значения «1» будет «0.8414709848».

Пример:

```
sin(last(/  _  /  ))
```

sinh(значение)

Гиперболический синус значения.

Параметр:

- **значение** — проверяемое значение

Например, гиперболический синус значения «1» будет «1.17520119364».

Пример:

```
sinh(last(/  _  /  ))
```

sqrt(значение)

Квадратный корень значения.<br> Эта функция выдаст ошибку для отрицательного значения.

Параметр:

- **значение** — проверяемое значение

Например, квадратный корень из значения «3.5» will be «1.87082869339».

Пример:

```
sqrt(last(/  _  /  ))
```

sum(<значение1>,<значение2>,...)

Сумма значений, на которые ссылаются перечисленные элементы данных.

Параметр:

- **значениеX** — значение, возвращаемое другой функцией, работающей с историей элементов данных.

Пример:

```
sum(avg(/  _  /  ,1h),avg(/  _  2/  2,1h))
```

tan(значение)

Тангенс значения.

Параметр:

- **значение** — проверяемое значение

Например, тангенс значения «1» будет «1.55740772465».

Пример:

```
tan(last(/  _  /  ))
```

truncate(значение,дес_знаков)

Усечение значения до указанного числа десятичных знаков.

Параметры:

- **значение** — проверяемое значение;
- **дес_знаков** — количество десятичных знаков, оставляемых после усечения (0 также допустим).

Например, значение как «2.5482», усечённое до 2 десятичных знаков, будет «2.54».

Пример:

```
truncate(last(/  _  /  ),2)
```

См. все поддерживаемые функции.

**7 Функции операторов**

Все функции, перечисленные здесь, поддерживаются в:

- Триггерных выражениях
- Вычисляемых элементах данных
- Макросах выражений

Функции перечислены без дополнительной информации. Щелкните по функции, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| between | Проверяет, принадлежит ли значение заданному диапазону. |
| in | Проверяет, равно ли значение хотя бы одному из перечисленных значений. |

Подробности функций

Некоторые общие замечания о параметрах функций:

- параметры функций разделяются запятыми
- в качестве параметров допустимы выражения

between(значение,мин,макс)

Проверка того, попадает ли значение в заданный диапазон.<br> Поддерживаемые типы значений: *Integer*, *Float*.<br> Возвращает: 1 — в диапазоне; 0 — в противном случае.

Параметры:

- **значение** — проверяемое значение;<br>
- **мин** — минимальное значение;<br>
- **макс** — максимальное значение.

Пример:

```
between(last(/  _  /  ),1,10)=1 #          ,                  1    10
```

in(значение,значение1,значение2,...значениеN)

Проверка того, что значение равно хотя бы какому-нибудь из перечисленных значений.<br> Поддерживаемые типы значений: *Integer*, *Float*, *Character*, *Text*, *Log*.<br> Возвращает: 1 — если равно; 0 — в противном случае.

Параметры:

- **значение** — проверяемое значение;<br>

- **значение1,значение2,...значениеN** — перечисленные значения (строковые значения должны быть в двойных кавычках).

Значение сравнивается с перечисленными значениями как число, если все эти значения могут быть преобразованы в число; в противном случае сравнивается как строка.

Примеры:

```
in(last(/  _  /  ),5,10)=1 #          ,                    5,    10
in("text",last(/  _  /  ),last(/  _  /  ,#2))=1 #          ,                              «text»
```

См. все поддерживаемые функции.

#### 8 Прогностические функции

Все функции, перечисленные здесь, поддерживаются в:

- Выражениях триггеров
- Вычисляемых элементах данных
- Макросах выражений

Функции перечислены без дополнительной информации. Щелкните функцию, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| forecast | Будущее значение, max, min, delta или avg элемента данных. |
| timeleft | Время в секундах, необходимое элементу данных для достижения указанного порога. |

Общие параметры

- /  _  /  является общим обязательным первым параметром для функций, ссылающихся на историю элемента данных узла сети
- (  |#  )<:  _  > является общим вторым параметром для функций, ссылающихся на историю элемента данных узла сети, где:
  - **сек** — максимальный период вычисления в секундах (могут использоваться суффиксы времени), либо
  - **#число** — максимальная область вычисления среди последних собранных значений (если начинается со знака решётки)
  - **сдвиг_времени** (опционально) позволяет сдвигать точку вычислений по времени назад в прошлое. Смотрите более подробную информацию относительно того, как указывать сдвиг времени.

Подробности функций

Некоторые общие замечания о параметрах функций:

- Параметры функций разделяются запятой
- Необязательные параметры функций (или части параметров) обозначаются < >
- Параметры, специфичные для функций, описаны с каждой функцией
- Параметры /host/key и (sec|#num)<:time shift> никогда не должны заключаться в кавычки

forecast(/узел_сети/ключ,(сек|#число)<:сдвиг_времени>,время,<аппроксимация>,<режим>)

Будущее значение, наибольшее, наименьшее, дельта или среднее элемента данных.<br> Поддерживаемые типы значений: *Float*, *Integer*.

Параметры:

- Смотрите общие параметры;<br>
- **время** — горизонт предсказания в секундах (могут использоваться суффиксы времени); поддерживаются отрицательные значения;<br>
- **аппроксимация** (опционально; должна быть в двойных кавычках) — функция, используемая для аппроксимации данных истории. Поддерживаются аппроксимации:<br>*linear* — линейная функция (по умолчанию)<br>*polynomialN* — полиномиальная степени N (1 <= N <= 6)<br>*exponential* — экспоненциальная функция<br>*logarithmic* — логарифмическая функция<br>*power* — степенная функция<br>Обратите внимание, что *polynomial1* эквивалентно *linear*;
- **режим** (опционально; должен быть в двойных кавычках) — требуемый вывод. Поддерживаются режимы:<br>*value* — значение (по умолчанию)<br>*max* — максимум<br>*min* — минимум<br>*delta* — *max-min*<br>*avg* — среднее<br>Обратите внимание, что *value* оценивает значение элемента данных на момент  _  +  ; *max*, *min*, *delta* и *avg* исследуют оценку значения элемента данных в интервале между  _  и  _  +  .

Комментарии:

- Если возвращаемое значение больше, чем 1.7976931348623157E+308, или меньше, чем -1.7976931348623157E+308, то возвращаемое значение усекается до 1.7976931348623157E+308 или -1.7976931348623157E+308 соответственно;
- Переходит в неподдерживаемое состояние только при некорректном использовании в выражении (ошибочный тип элемента данных, недопустимые параметры), в противном случае при ошибках возвращает -1;
- Смотрите также дополнительную информацию о прогнозирующих функуциях триггеров.

Примеры:

```
forecast(/  _  /  ,#10,1h) #                          ,              10
forecast(/  _  /  ,1h,30m) #                        30
forecast(/  _  /  ,1h:now-1d,12h) #                       12
forecast(/  _  /  ,1h,10m,"exponential") #                        10    ,
forecast(/  _  /  ,1h,2h,"polynomial3","max") #                  ,
forecast(/  _  /  ,#2,-20m) #                    20                              (                    ,
```

timeleft(/узел_сети/ключ,(сек|#число)<:сдвиг_времени>,порог,<аппроксимация>)

Время в секундах, которое требуется для достижения элементом данных указанного порогового значения.<br> Поддерживаемые типы значений: *Float*, *Integer*.

Параметры:

- Смотрите общие параметры;<br>
- **порог** — достигаемое значение (могут использоваться суффиксы единиц измерения);
- **аппроксимация** (опционально; должна быть в двойных кавычках) — см. forecast().

Комментарии:

- Если возвращаемое значение больше, чем 1.7976931348623157E+308, то возвращаемое значение усекается до 1.7976931348623157E+308;
- Возвращает 1.7976931348623157E+308, если пороговое значение недостижимо;
- Переходит в неподдерживаемое состояние только при некорректном использовании в выражении (ошибочный тип элемента данных, недопустимые параметры), в противном случае при ошибках возвращает -1;
- Смотрите также дополнительную информацию о прогнозирующих функуциях триггеров.

Примеры:

```
timeleft(/  _  /  ,#10,0) #          ,                              ,              10
timeleft(/  _  /  ,1h,100) #          ,                           100,
timeleft(/  _  /  ,1h:now-1d,100) #          ,                          100,
timeleft(/  _  /  ,1h,200,"polynomial2") #          ,                             200,
```

См. все поддерживаемые функции.

#### 9 Строковые функции

Все функции, перечисленные здесь, поддерживаются в:

- Выражения триггеров
- Вычисляемые элементы данных
- Макросы выражений

Функции перечислены без дополнительной информации. Щелкните по функции, чтобы увидеть полные сведения.

| Function | Description |
|---|---|
| ascii | ASCII-код самого левого символа значения. |
| bitlength | Длина значения в битах. |
| bytelength | Длина значения в байтах. |
| char | Возвращает символ, интерпретируя значение как ASCII-код. |
| concat | Строка, полученная путем объединения значений связанных элементов данных или постоянных значений. |
| insert | Вставляет указанные символы или пробелы в строку, начиная с указанной позиции в строке. |
| jsonpath | Возвращает результат JSONPath. |
| left | Возвращает самые левые символы значения. |
| length | Длина значения в символах. |
| ltrim | Удаляет указанные символы с начала строки. |
| mid | Возвращает подстроку из N символов, начиная с позиции символа, указанной в ’start’. |

| Function | Description |
|---|---|
| repeat | Повторяет строку. |
| replace | Находит шаблон в значении и заменяет его на replacement. |
| right | Возвращает самые правые символы значения. |
| rtrim | Удаляет указанные символы с конца строки. |
| trim | Удаляет указанные символы с начала и конца строки. |
| xmlxpath | Возвращает результат XML XPath. |

Подробности функций

Некоторые общие замечания о параметрах функций:

- параметры функций разделяются запятыми
- в качестве параметров допустимы выражения
- строковые параметры должны заключаться в двойные кавычки; в противном случае они могут быть интерпретированы неверно
- необязательные параметры функций (или части параметров) обозначаются угловыми скобками< >

ascii(значение)

Код ASCII самого левого символа значения.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметр:

- **значение** — проверяемое значение

Например, значение «Abc» вернёт «65» (код ASCII для буквы «A»).

Пример:

```
ascii(last(/  _  /  ))
```

bitlength(значение)

Длина значения в битах.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*, *Integer*.

Параметр:

- **значение** — проверяемое значение

Пример:

```
bitlength(last(/  _  /  ))
```

bytelength(значение)

Длина значения в байтах.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*, *Integer*.

Параметр:

- **значение** — проверяемое значение

Пример:

```
bytelength(last(/  _  /  ))
```

char(значение)

Вернуть символ, интерпретируя значение как код ASCII.<br> Поддерживаемые типы значений: *Integer*.

Параметр:

- **значение** — проверяемое значение

Значение должно быть в диапазоне 0-255. Например, значение «65» (интерпретируемое как код ASCII) вернёт «A».

Пример:

```
char(last(/  _  /  ))
```

concat(<значение1>,<значение2>,...)

Строка, получаемая в результате конкатенации упоминаемых значений элементов данных либо констант.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*, *Float*, *Integer*.

Параметр:

- **значениеX** — значение, возвращаемое одной из функций истории, либо константа (string, integer или float number). Должно содержать не менее двух параметров.

Например, значение вроде «Zab», объединённое с «bix» (строки-константы), вернёт «Zabbix».

Примеры:

```
concat(last(/  _  /  ),"bix")
concat("1 min: ",last(/  _  /system.cpu.load[all,avg1]),", 15 min: ",last(/  _  /system.cpu.load[all,avg1
```

insert(значение,старт,длина,замена)

Вставить указанные символы или пробелы в символьную строку, начиная с указанной позиции в строке.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **старт** — начальная позиция;<br>
- **длина** — заменяемых позиций;<br>
- **замена** — подставляемая строка.

Например, значение вроде «Zabbbix» будет заменено на «Zabbix», если «bb» (начиная с позиции 3, заменяемых позиций 2) заменяется строкой «b».

Пример:

```
insert(last(/  _  /  ),3,2,"b")
```

jsonpath(значение,путь,<умолчание>)

Вернуть результат JSONPath.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **путь** — путь (должен быть заключён в кавычки);<br>
- **умолчание** — необязательное резервное значение на случай, если запрос JSONPath не возвращает данных. Обратите внимание, что при других ошибках возвращается сбой (например, «unsupported construct»).

Пример:

```
jsonpath(last(/  _  /proc.get[zabbix_agentd,,,summary]),"$..size")
```

left(значение,кол-во)

Вернуть самые левые символы значения.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **кол-во** — количество возвращаемых символов.

Например, можно вернуть «Zab» из «Zabbix», указывая для возврата 3 крайних слева символа. См. также right().

Пример:

```
left(last(/  _  /  ),3) #
```

length(value)

Длина значения в символах.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметр:

- **значение** — проверяемое значение.

Примеры:

```
length(last(/  _  /  )) #
length(last(/  _  /  ,#3)) #
length(last(/  _  /  ,#1:now-1d)) #
```

ltrim(значение,<символы>)

Удалить указанные символы с начала строки.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **символы** (опционально) — укажите удаляемые символы.

По умолчанию удаляются пробелы (если не указано опциональных символов). См. также: rtrim(), trim().

Примеры:

```
ltrim(last(/  _  /  )) #
ltrim(last(/  _  /  ),"Z") #             «Z»
ltrim(last(/  _  /  )," Z") #                «Z»
```

mid(значение,старт,длина)

Вернуть подстроку из N символов, начиная с символьной позиции, указанной параметром «старт».<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **старт** — начальная позиция подстроки;<br>
- **длина** — количество позиций подстроки, которые нужно вернуть.

Например, можно вернуть подстроку «abbi» из значения «Zabbix», если указать начальную позицию 2, а длину 4.

Пример:

```
mid(last(/  _  /  ),2,4)="abbi"
```

repeat(значение,кол-во)

Повторить строку.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **кол-во** — количество повторений.

Пример:

```
repeat(last(/  _  /  ),2) #
```

replace(значение,образец,замена)

Найти в значении образец и заменить его заменой. Будут заменены все вхождения образца.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **образец** — искомый образец;<br>
- **замена** — строка, которой заменяется образец.

Пример:

```
replace(last(/  _  /  ),"ibb","abb") —          «ibb»   «abb»
```

right(value,count)

Вернуть самые правые символы значения.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **кол-во** — количество возвращаемых символов.

Например, можно вернуть «bix» из «Zabbix», если указать возвращать 3 символа справа. См. также left().

Пример:

```
right(last(/  _  /  ),3) #
```

rtrim(значение,<символы>)

Удалить указанные символы с конца строки.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **символы** (опционально) — укажите удаляемые символы.

По умолчанию удаляются пробелы (если не указано опциональных символов). См. также: ltrim(), trim().

Примеры:

```
rtrim(last(/  _  /  )) #
rtrim(last(/  _  /  ),"x") #              «x»
rtrim(last(/  _  /  ),"x ") #              «x»
```

trim(значение,<символы>)

Удалить указанные символы с начала и конца строки.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **символы** (опционально) — укажите удаляемые символы.

По умолчанию удаляются пробелы (если не указано опциональных символов). См. также: ltrim(), rtrim().

Примеры:

```
trim(last(/  _  /  )) #
trim(last(/  _  /  ),"_") #     «_»
```

xmlxpath(значение,путь,<умолчание>)

Вернуть результат XML XPath.<br> Поддерживаемые типы значений: *String*, *Text*, *Log*.

Параметры:

- **значение** — проверяемое значение;<br>
- **путь** — путь (должен быть заключён в кавычки);<br>
- **умолчание** — необязательное резервное значение на случай, если запрос XML XPath возвращает пустой набор узлов. Оно не будет возвращено, если пустой результат не является набором узлов (т.е. пустой строкой). При других ошибках возвращается сбой (например, «invalid expression»).

Пример:

```
xmlxpath(last(/host/xml_result),"/response/error/status")
```

См. все поддерживаемые функции.

#### 6. Макросы

Возможно использовать встроенные Поддерживаемые макросы и Пользовательские макросы, поддерживаемые по местоположению.

#### 1 Поддерживаемые макросы по расположению

Обзор

Таблица макросов, поддерживаемых в зависимости от расположения, содержит справочную информацию по всем макросам (встроенным, выражениям и пользовательским), которые поддерживаются в конкретном месте.

Если поддерживаются все макросы из связанной группы, указывается только имя группы. Щелкните по имени каждого макроса или имени группы, чтобы увидеть полные сведения о макросах.

Список сведений о макросах содержит подробный перечень всех **встроенных** макросов, сгруппированных по области применения. Обратите внимание, что для настройки значений макросов (например, для сокращения или извлечения определенных подстрок) можно использовать функции макросов.

Макросы, поддерживаемые в зависимости от расположения

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| **Action** | | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| *Условие периода времени, Длительность шага операции по умолчанию, Длительность шага* | Пользовательские макросы (только глобальные) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются |
| **Уведомления и команды** | | |
| *На основе триггера* | Макросы Action<br>Макросы Дата/время<br>{ESC.HISTORY}<br>Макросы События, макросы События причины/симптома<br>Макросы обновления события: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS}<br>Макрос выражения: {?EXPRESSION} {FUNCTION.VALUE}<br>Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы целевого узла сети (только команды)<br>Макросы инвентаря узла сети<br>Макросы элемента данных (кроме {ITEM.STATE}, {ITEM.STATE.ERROR})<br>Макросы прокси<br>Макросы триггера: {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}<br>Пользовательские макросы | См. также: Использование макросов в сообщениях |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| *Обновление проблемы* | Макросы Action<br>Макросы Дата/время<br>{ESC.HISTORY}<br>Макросы События, макросы События причины/симптома, макросы События восстановления<br>Макросы обновления события: все (кроме {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY})<br>Макрос выражения: {?EXPRESSION}<br>Макросы функций<br>Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы целевого узла сети (только команды)<br>Макросы инвентаря узла сети<br>Макросы элемента данных (кроме {ITEM.STATE}, {ITEM.STATE.ERROR})<br>Макросы прокси<br>Макросы триггера: {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}<br>{USER.FULLNAME}<br>Пользовательские макросы | |
| *Восстановление проблемы* | Макросы События восстановления<br>{FUNCTION.RECOVERY.VALUE}<br>{TRIGGER.EXPRESSION.RECOVERY.EXPLAIN} | |
| *Обнаружение* | Макросы Action<br>Макросы Дата/время<br>Макросы обнаружения<br>Макросы события: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP}<br>Макросы целевого узла сети (только команды)<br>Макросы прокси | |
| *Автоматическая регистрация* | Макросы Action<br>Макросы Дата/время<br>Макросы события: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP}<br>Макросы узла сети: {HOST.HOST}, {HOST.IP}, {HOST.PORT}, {HOST.METADATA}<br>Макросы целевого узла сети (только команды)<br>Макросы прокси | |
| *На основе сервиса* | Макросы Action<br>Макросы Дата/время<br>{ESC.HISTORY}<br>Макросы события: все, кроме {EVENT.ACK.STATUS}, {EVENT.OPDATA}<br>Макросы сервиса<br>Пользовательские макросы | |
| *Восстановление сервиса* | Макросы События (кроме {EVENT.ACK.STATUS}, {EVENT.OPDATA}), макросы События восстановления | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| *Обновление сервиса* | Макросы Action<br>Макросы Дата/время<br>{ESC.HISTORY}<br>Макросы События (кроме {EVENT.ACK.STATUS}, {EVENT.OPDATA})<br>Макросы обновления события: {EVENT.UPDATE.DATE}, {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}, {EVENT.UPDATE.STATUS}, {EVENT.UPDATE.TIME}, {EVENT.UPDATE.TIMESTAMP}<br>Макросы сервиса<br>Пользовательские макросы | |
| **Внутренние уведомления** | Action макросы<br>Дата/время макросы<br>{ESC.HISTORY}<br>Макросы события (кроме {EVENT.NSEVERITY}, {EVENT.SEVERITY}), макросы События восстановления<br>Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети<br>Макросы элемента данных: {ITEM.DESCRIPTION}, {ITEM.DESCRIPTION.ORIG}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, {ITEM.NAME}, {ITEM.NAME.ORIG}, {ITEM.VALUETYPE}<br>Макросы прокси | |
| *На основе элемента данных* | Макросы элемента данных: {ITEM.STATE}, {ITEM.STATE.ERROR} | |
| *На основе правила LLD* | Макросы уведомления низкоуровневого обнаружения | |
| *На основе триггера* | Макросы триггера: {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}<br>Пользовательские макросы | |
| **Параметры alert script Connector** | Alert макросы<br>Пользовательские макросы (только глобальные) | |
| *Поле URL, Username, Password, Bearer token, HTTP proxy, SSL certificate, SSL key file, SSL key password* | Пользовательские макросы (только глобальные) | |
| *Поле Timeout* | Пользовательские макросы (только глобальные) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| **Имя графика** | Макрос выражения: {?EXPRESSION} | В этом макросе в данном расположении поддерживается только одна функция **avg**, **last**, **max** или **min** с параметром в секундах (можно использовать суффиксы времени). В качестве узла сети в макросе можно использовать макрос {HOST.HOST<1-9>} (см. также индексированные макросы). Пример использования: {?avg(/{HOST.HOST}/item.key,1h)} Неподдерживаемое использование: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)} |
| **Узел сети, прототип узла сети** | | |
| *Interface IP/DNS* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME} Пользовательские макросы [1] | [1] Для IP-адреса одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются |
| *Interface port* | Пользовательские макросы | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *SNMP community*<br>*SNMPv3 Context name, Security name, Authentication passphrase, Privacy passphrase*<br>*IPMI username, password* | Пользовательские макросы | |
| **Элемент данных, прототип элемента данных, правило LLD** | | |
| *Name* | Пользовательские макросы | Не поддерживается в именах правил LLD. |
| *Description* | Пользовательские макросы | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| *Key parameters* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | Поддерживаемые макросы {HOST.*} будут разрешаться в интерфейс, выбранный для элемента данных. При использовании в элементах данных без интерфейсов они будут разрешаться либо в интерфейс агента Zabbix, SNMP, JMX или IPMI узла сети в этом порядке приоритета, либо в 'UNKNOWN', если у узла сети нет ни одного интерфейса. |
| *Update interval*, *Custom intervals*, *Timeout (available for supported item types)*, *Store up to (for history/trends)* fields | Пользовательские макросы | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *Delete lost resources*, *Disable lost resources* fields | Пользовательские макросы | Эти поля поддерживаются только для правил LLD.<br><br>Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *Preprocessing step parameters* (including custom scripts), *custom error-handling parameters* (*Set value to* and *Set error to* fields) | Пользовательские макросы | |
| *Filter regular expressions* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | Эти поля поддерживаются только для правил LLD. |
| *Override filter regular expressions* | Пользовательские макросы | Эти поля поддерживаются только для правил LLD. |
| Override operations: *Update interval*, *History storage period*, *Trend storage period* (for item prototypes) | Пользовательские макросы | Эти поля поддерживаются только для правил LLD.<br><br>Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| **Типы элементов данных** | | |
| **Browser** элемент данных<br>*имена и значения параметров* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети<br>Макросы элемента данных: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}<br>Пользовательские макросы | |
| **Script** элемент данных<br>*имена и значения параметров* | | |
| **Calculated/aggregate** формула элемента данных (*константы выражения*, *параметры функции*, *параметры ключа элемента данных*, (только для агрегированного элемента данных) *условия фильтра* (имя группы узлов сети и имя тега)) | Пользовательские макросы | |
| **Database monitor** элемент данных<br>*поля Username, Password, SQL query* | Пользовательские макросы | |
| **HTTP agent** элемент данных<br>*поля URL, Query fields, Request body, Headers (names and values), SSL certificate file, SSL key file* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы элемента данных: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}<br>Пользовательские макросы | URL, содержащие секретный макрос, не будут работать, так как макрос в них будет разрешен как "******" |
| **HTTP agent** элемент данных<br>*поле HTTP Proxy* | Макросы элемента данных: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}<br>Пользовательские макросы | |
| **HTTP agent** элемент данных<br>*поля Required status codes, HTTP authentication username/password, SSL key password* | Пользовательские макросы | |
| **HTTP agent** элемент данных<br>*поле Allowed hosts* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | |
| **JMX agent** элемент данных<br>*поле Endpoint* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.PORT}<br>Пользовательские макросы | |
| **Trapper** элемент данных<br>*поле Allowed hosts* | | |
| **SNMP agent** элемент данных<br>*поле SNMP OID* | Пользовательские макросы | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| **SSH and Telnet** элемент данных *script* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | |
| **SSH agent** элемент данных *поля Username, Public key file, Private key file, Password* | Пользовательские макросы | |
| **Telnet agent** элемент данных *поля Username, Password* | Пользовательские макросы | |
| **Карта** | | |
| *URL, URL name* | Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>{HOSTGROUP.ID}<br>Макросы инвентаря узла сети<br>Макросы карты<br>{TRIGGER.ID} | |
| *Element label* | Макрос выражения: {?EXPRESSION} [1]<br>Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>{HOSTGROUP.ID}<br>Макросы инвентаря узла сети<br>Макросы карты<br>Макросы триггера: {TRIGGER.EVENTS.*}, {TRIGGER.ID}, {TRIGGER.PROBLEM.EVENTS.*}, {TRIGGERS.ACK}, {TRIGGERS.PROBLEM.ACK}, {TRIGGERS.PROBLEM.UNACK}, {TRIGGERS.UNACK} | [1] В этом макросе в данном расположении поддерживается только одна функция **avg**, **last**, **max** или **min** с параметром в секундах (можно использовать суффиксы времени).<br>В качестве узла сети в макросе можно использовать макрос {HOST.HOST<1-9>}.<br>Пример использования: {?avg(/{HOST.HOST}/item.key,1h)}<br>Неподдерживаемое использование: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)} |
| *Link label* | Макрос выражения: {?EXPRESSION} [1] | |
| *Shape label* | Макрос выражения: {?EXPRESSION} [1] | |
| *Text field in map shapes* | {MAP.NAME} | |
| **Сетевое обнаружение** | | |
| *Поле Update interval* | Пользовательские макросы (только глобальные) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *Поля SNMP community, SNMP OID*<br>*Поля SNMPv3 Context name, Security name, Authentication passphrase, Privacy passphrase* | Пользовательские макросы (только глобальные) | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| **Прокси** | | |
| *Address for active agents* > *Port* (если прокси принадлежит группе)<br>Для пассивного режима прокси: поля интерфейса *Address* и *Port*<br>Поля *Timeouts for item types* в режиме переопределения | Пользовательские макросы (только глобальные) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| **Группа прокси** | | |
| *Поля Failover period, Minimum number of proxies* | Пользовательские макросы (только глобальные) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| **Скрипты** | | |
| *Команды, текст подтверждения* (Тип: script, ручное действие **узла сети**) | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID} [1], {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети [2]<br>{MANUALINPUT}<br>Макросы Username<br>Пользовательские макросы | [1] Только текст подтверждения<br>[2] Поддерживается для Zabbix server и Zabbix proxy |
| *Команды* (Тип: script, ручное действие **события**) | Макросы Дата/время<br>Макросы События, макросы События причины/симптома, макросы События восстановления (если восстановление произошло)<br>Макросы обновления события: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS}<br>{FUNCTION.VALUE}<br>Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети<br>Макросы элемента данных (кроме {ITEM.STATE}, {ITEM.STATE.ERROR})<br>{MANUALINPUT}<br>Макросы прокси<br>Макросы триггера: {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}<br>Макросы Username<br>Пользовательские макросы | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| *Текст подтверждения* (Тип: script, ручное действие **события**) | Макросы события: {EVENT.ID}, {EVENT.NAME}, {EVENT.NSEVERITY}, {EVENT.SEVERITY}, {EVENT.STATUS}, {EVENT.VALUE}<br>Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети {MANUALINPUT}<br>Макросы Username<br>Пользовательские макросы | |
| *URL, текст подтверждения* (Тип: URL, ручное действие **узла сети/события**) | | |
| **Имя и значение тега** | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети<br>Макросы элемента данных: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*}<br>Макросы триггера: {TRIGGER.ID} (только значение тега триггера)<br>Пользовательские макросы | Макросы имени и значения тега разрешаются только в процессе генерации события. |
| **Триггер** | | |
| *Имя* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы элемента данных: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*}<br>Позиционные макросы/ссылки: $1...$9<br>Пользовательские макросы | |
| *Имя события* | Макросы Дата/время: {TIME}, {TIMESTAMP}<br>Макрос выражения: {?EXPRESSION}<br>Макросы функций<br>Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы элемента данных: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*}<br>Макросы триггера: {TRIGGER.EXPRESSION.EXPLAIN} | |
| *Оперативные данные* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы элемента данных: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*}<br>Пользовательские макросы | |
| *Выражение* | {TRIGGER.VALUE}<br>Пользовательские макросы [1] | [1] Только в константах и параметрах функций; секретные макросы не поддерживаются |
| *Поля Tag for matching, Menu entry name, Menu entry URL* | Пользовательские макросы | URL, содержащие секретный макрос, не будут работать, так как макрос в них будет разрешен как "******". |
| *Описание* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы элемента данных: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*}<br>Пользовательские макросы | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| *URL* | {EVENT.ID}<br>Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы элемента данных: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*}<br>{TRIGGER.ID} | |
| **Команды user parameter** | Позиционные макросы/ссылки: $1...$9 | |
| **Веб-сценарий** | | |
| *Поля Name, Agent, HTTP proxy* | Пользовательские макросы | |
| *Поле Update interval* | Пользовательские макросы | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *Поля Variables, Headers, SSL certificate, SSL key file* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | |
| *Поля User, Password, SSL key password* | Пользовательские макросы | |
| Шаг веб-сценария *поля Name, Variables (только значения)* | Пользовательские макросы | |
| Шаг веб-сценария *поля Timeout, Required status codes* | Пользовательские макросы | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| Шаг веб-сценария *поля URL, Headers (имена и значения), Required string* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | |
| Шаг веб-сценария *поле Post* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} | |
| **Тип медиа web-hook** | | |
| *Значения параметров* | Макросы Alert<br>Все макросы, поддерживаемые в уведомлениях о проблемах на основе триггера (см. выше) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *Скрипт* JavaScript, имя параметра | Пользовательские макросы (только глобальные) | |
| *Скрипт* JavaScript | | |
| *Поля Menu entry name, Menu entry URL* | {EVENT.TAGS.<tag name>} | |
| **Виджеты** | | |

| Расположение | Поддерживаемые макросы | Комментарии |
|---|---|---|
| **Gauge** виджет<br>*поле Description* | Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети<br>Макросы элемента данных (кроме {ITEM.STATE.ERROR})<br>Пользовательские макросы (только глобальные) | |
| **Honeycomb** виджет<br>*поле Primary/secondary label* | | |
| **Item value** виджет<br>*поле Description* | | |
| **Top hosts** виджет<br>столбец *Text data* | Макросы узла сети: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Макросы инвентаря узла сети<br>Пользовательские макросы (только глобальные) | |
| **URL** (Dynamic) виджет<br>*поле URL* | Макросы узла сети: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}<br>Пользовательские макросы | |
| **Другие расположения** | | |
| *Поле Working time* | Пользовательские макросы (только глобальные) | Одним макросом должно заполняться все поле; несколько макросов/смешивание с текстом не поддерживаются. |
| *Поле Timeouts for item types* (глобальная настройка)<br>Медиа пользователя<br>*поле When active* | | |
| Тип медиа Email<br>*поля Username, Password* | Пользовательские макросы (только глобальные) | Рекомендуются секретные макросы. |
| Запланированные отчеты<br>*поля Subject, Message* | Макросы Дата/время: {TIME}, {TIMESTAMP} | |

Сведения о макросах

Этот список содержит сведения о каждом встроенном макросе, сгруппированном по области применения:

- Макросы действий
- Макросы оповещений
- Макросы даты и времени
- Макросы обнаружения
- Макросы событий
  - Макросы обновления событий
  - Макросы событий причины/симптома
  - Макросы событий восстановления
- Макросы эскалации
- Макросы функций
- Макросы узла сети
  - Макросы целевого узла сети
- Макросы группы узлов сети
- Макросы инвентаря узла сети
- Макросы элемента данных
- Макросы уведомлений низкоуровневого обнаружения

- Макросы карты
- Макросы прокси
- Макросы скриптов
- Макросы сервисов
- Макросы триггера
- Макросы имени пользователя

Макросы действия

{ACTION.ID}

Числовой идентификатор сработавшего действия.<br>

{ACTION.NAME}

Имя сработавшего действия.<br>

Макросы оповещений

{ALERT.MESSAGE}

Значение *Сообщение по умолчанию* из конфигурации действия.<br>

{ALERT.SENDTO}

Значение *Send to* из конфигурации пользовательского носителя.<br>

{ALERT.SUBJECT}

Значение *Тема по умолчанию* из конфигурации действия.<br>

Макросы даты и времени

{DATE}

Текущая дата в формате `yyyy.mm.dd`.<br>

{TIME}

Текущее время в формате `hh:mm:ss`.<br>

{TIMESTAMP}

Текущее время в формате UNIX timestamp.<br>

Макросы обнаружения

{DISCOVERY.DEVICE.IPADDRESS}

IP-адрес обнаруженного устройства.<br> Доступно всегда, не зависит от того, добавлен ли узел сети.

{DISCOVERY.DEVICE.DNS}

DNS-имя обнаруженного устройства.<br> Доступно всегда, не зависит от того, добавлен ли узел сети.

{DISCOVERY.DEVICE.STATUS}

Статус обнаруженного устройства (UP/DOWN).<br>

{DISCOVERY.DEVICE.UPTIME}

Время, прошедшее с момента последнего изменения статуса обнаружения для конкретного устройства.<br> Например: 1h 29m 01s.<br> Для устройств со статусом DOWN это период их простоя.

{DISCOVERY.RULE.NAME}

Имя правила обнаружения, которое выявило наличие или отсутствие устройства/службы.

{DISCOVERY.SERVICE.NAME}

Имя обнаруженной службы.<br> Например: HTTP.

{DISCOVERY.SERVICE.PORT}

Порт обнаруженной службы.<br> Например: 80.

{DISCOVERY.SERVICE.STATUS}

Статус обнаруженной службы (UP/DOWN).<br>

{DISCOVERY.SERVICE.UPTIME}

Время, прошедшее с момента последнего изменения статуса обнаружения для определенной службы.<br> Например: 1h 29m 01s.<br> Для служб со статусом DOWN это период их простоя.

Макросы эскалации

{ESC.HISTORY}

Журнал ранее отправленных уведомлений, их шаг эскалации и статус (*sent*, *in progress* или *failed*).<br>

Макросы событий

{EVENT.ACK.STATUS}

Статус подтверждения события (Да/Нет).<br>

{EVENT.AGE}

Возраст события, которое вызвало действие, с точностью до секунды.<br> Полезно в эскалированных сообщениях. Например: 9m 13s.

{EVENT.DATE}

Дата события, которое вызвало действие, в формате `yyyy.mm.dd`. <br> Например: 2025.04.14.

{EVENT.DURATION}

Длительность события (разница во времени между событиями проблемы и восстановления) с точностью до секунды.<br> Полезно в сообщениях о восстановлении после проблемы.

{EVENT.ID}

Числовой идентификатор события, которое вызвало действие.<br>

{EVENT.NAME}

Имя события, которое вызвало действие.<br>

{EVENT.NSEVERITY}

Числовое значение серьезности события.<br> Возможные значения: *0* - Не классифицировано, *1* - Информация, *2* - Предупреждение, *3* - Средняя, *4* - Высокая, *5* - Катастрофа.

{EVENT.OBJECT}

Числовое значение объекта события.<br> Возможные значения: *0* - Триггер, *1* - Обнаруженный узел сети, *2* - Обнаруженная служба, *3* - Автоматическая регистрация, *4* - Элемент данных, *5* - Правило низкоуровневого обнаружения.

{EVENT.OPDATA}

Операционные данные базового триггера проблемы.<br>

{EVENT.SEVERITY}

Название серьезности события.<br>

{EVENT.SOURCE}

Числовое значение источника события.<br> Возможные значения: *0* - триггер, *1* - обнаружение, *2* - автоматическая регистрация, *3* - внутренний, *4* - сервис.

{EVENT.STATUS}

Словесное значение события, которое вызвало действие.<br>

{EVENT.TAGS}

Список тегов события, разделенных запятыми.<br>Расширяется в пустую строку, если теги отсутствуют.<br>

{EVENT.TAGSJSON}

JSON-массив, содержащий объекты тегов события.<br>Разворачивается в пустой массив, если теги отсутствуют.<br>

{EVENT.TAGS.<tag name>}

Значение тега события, на которое ссылается имя тега.<br> Имя тега, содержащее неалфавитно-цифровые символы (включая неанглийские многобайтовые UTF-символы), должно быть заключено в двойные кавычки. Кавычки и обратные слеши внутри имени тега в кавычках должны быть экранированы обратным слешем.

{EVENT.TIME}

Время события, которое вызвало действие, в формате `hh:mm:ss`.<br> Например: 12:57:53.

{EVENT.TIMESTAMP}

Метка времени UNIX события, которое вызвало действие.<br>

{EVENT.VALUE}

Числовое значение события, которое вызвало действие.<br> Возможные значения: *1* - проблема, *0* - восстановление.<br>

Макросы событий причины/симптома

{EVENT.CAUSE.ACK.STATUS}

Статус подтверждения события-причины (Да/Нет).<br> Используется в контексте события-симптома.

{EVENT.CAUSE.AGE}

Возраст события-источника с точностью до секунды.<br> Полезно в эскалированных сообщениях.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.DATE}

Дата события-причины в формате `yyyy.mm.dd`.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.DURATION}

Длительность события-источника (разница во времени между событиями проблемы и восстановления) с точностью до секунды.<br> Полезно в сообщениях о восстановлении после проблемы.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.ID}

Числовой ID события причины.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.NAME}

Имя события-причины.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.NSEVERITY}

Числовое значение серьезности события-причины.<br> Возможные значения: *0* - Не классифицировано, *1* - Информация, *2* - Предупреждение, *3* - Средняя, *4* - Высокая, *5* - Катастрофа.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.OBJECT}

Числовое значение объекта события причины.<br> Возможные значения: *0* - триггер, *1* - обнаруженный узел сети, *2* - обнаруженная служба, *3* - авторегистрация, *4* - элемент данных, *5* - правило низкоуровневого обнаружения.<br> Используется в контексте события симптома.

{EVENT.CAUSE.OPDATA}

Операционные данные базового триггера проблемы-источника.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.SEVERITY}

Название серьезности события-причины.<br> Возможные значения: *Not classified*, *Information*, *Warning*, *Average*, *High*, *Disaster*.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.SOURCE}

Числовое значение источника события причины.<br> Возможные значения: *0* - Trigger, *1* - Discovery, *2* - Autoregistration, *3* - Internal.<br> Используется в контексте события симптома.

{EVENT.CAUSE.STATUS}

Словесное значение события-источника.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.TAGS}

Список тегов события причины, разделенный запятыми.<br>Раскрывается в пустую строку, если теги отсутствуют.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.TAGSJSON}

JSON-массив, содержащий объекты тегов события-источника.<br>Разворачивается в пустой массив, если теги отсутствуют.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.TAGS.<tag name>}

Значение тега события-причины, на которое ссылается имя тега.<br> Имя тега, содержащее неалфавитно-цифровые символы (включая неанглийские многобайтовые UTF-символы), должно быть заключено в двойные кавычки. Кавычки

и обратные слэши внутри имени тега в кавычках должны быть экранированы обратным слэшем.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.TIME}

Время события-источника в формате `hh:mm:ss`.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.TIMESTAMP}

UNIX-метка времени события-причины.<br> Используется в контексте события-симптома.

{EVENT.CAUSE.UPDATE.HISTORY}

Журнал обновлений проблемы-источника (подтверждения и т. д.).<br> Используется в контексте события-симптома.

{EVENT.CAUSE.VALUE}

Числовое значение события причины<br> Возможные значения: *1* - Проблема, *0* - Восстановление.<br> Используется в контексте события-симптома.

{EVENT.SYMPTOMS}

Список событий симптомов. Включает следующие сведения: имя узла сети, имя события, уровень серьезности, возраст, теги сервиса и значения.<br> Этот макрос используется в контексте события причины и возвращает информацию о событиях симптомов.

Макросы события восстановления

{EVENT.RECOVERY.DATE}

Дата события восстановления в формате `yyyy.mm.dd`.<br>

{EVENT.RECOVERY.ID}

Числовой идентификатор события восстановления.<br>

{EVENT.RECOVERY.NAME}

Имя события восстановления.<br>

{EVENT.RECOVERY.STATUS}

Словесное значение события восстановления.<br>

{EVENT.RECOVERY.TAGS}

Список тегов события восстановления, разделенный запятыми. Если тегов нет, разворачивается в пустую строку.<br>

{EVENT.RECOVERY.TAGSJSON}

JSON-массив, содержащий объекты тегов события восстановления. Разворачивается в пустой массив, если теги отсутствуют.<br>

{EVENT.RECOVERY.TIME}

Время события восстановления в формате `hh:mm:ss`.<br>

{EVENT.RECOVERY.TIMESTAMP}

UNIX-метка времени события восстановления.<br>

{EVENT.RECOVERY.VALUE}

Числовое значение события восстановления.<br>

Макросы обновления события

{EVENT.UPDATE.ACTION}

Читаемое человеком название действия(й), выполненного(ых) во время обновления проблемы.<br> Принимает следующие значения: *acknowledged*, *unacknowledged*, *commented*, *changed severity from (original severity) to (updated severity)* и *closed* (в зависимости от того, сколько действий выполнено при одном обновлении).

{EVENT.UPDATE.ACTIONJSON}

Массив JSON, содержащий сведения о действии(ях), выполненных во время обновления проблемы.<br> Возможные значения свойств JSON:<br>- true (для свойств `acknowledge`, `unacknowledge`, `close`, `unsuppress`, `cause` и `symptom`);<br>- <строка сообщения> (для свойства `message`);<br>- timestamp (для свойств `suppress_until` и `timestamp`) или 0 (для свойства `suppress_until`, если подавление бессрочное);<br>- 0, 1, 2, 3, 4, 5 (для свойств `old` и `new` severity).<br><br> Например: `{"acknowledge":true,"message":"Monthly maintenance.","severity":{"old":2,"new":1},"suppress_until`

{EVENT.UPDATE.DATE}

Дата обновления события (подтверждение и т. д.) в формате yyyy.mm.dd.<br>{ACK.DATE} устарел и может не разрешаться в поддерживаемых местах.

{EVENT.UPDATE.HISTORY}

Журнал обновлений проблемы (подтверждения и т. д.).<br> {EVENT.ACK.HISTORY} устарел и может не разрешаться в поддерживаемых местах.

{EVENT.UPDATE.MESSAGE}

Сообщение обновления проблемы.<br> {ACK.MESSAGE} устарел и может не разрешаться в поддерживаемых местах.

{EVENT.UPDATE.NSEVERITY}

Числовое значение новой серьезности события, установленной во время операции обновления проблемы.<br>

{EVENT.UPDATE.SEVERITY}

Имя новой серьезности события, установленной во время операции обновления проблемы.<br>

{EVENT.UPDATE.STATUS}

Числовое значение статуса обновления проблемы.<br> Возможные значения: *0* - вебхук был вызван из-за события проблемы/восстановления, *1* - операция обновления.<br>

{EVENT.UPDATE.TIME}

Время обновления события (подтверждение и т. д.) в формате hh:mm:ss.<br>{ACK.TIME} устарел и может не разрешаться в поддерживаемых местах.

{EVENT.UPDATE.TIMESTAMP}

UNIX-метка времени обновления события (подтверждение и т. д.)<br>

Функциональные макросы

{FUNCTION.VALUE}

Значение N-й функции на основе элемента данных в выражении триггера на момент события.<br> Учитываются только функции, у которых в качестве первого параметра указан /host/key.<br><br> Этот макрос можно использовать с числовым индексом, например {FUNCTION.VALUE<1-9>}, чтобы указать на первую, вторую, третью и т. д. функцию в выражении триггера. См. indexed macros.

{FUNCTION.RECOVERY.VALUE}

Значение N-й функции на основе элемента данных в выражении восстановления на момент события.*<br> Учитываются только функции, у которых в качестве первого параметра указан /host/key.<br><br> Этот макрос можно использовать с числовым индексом, например {FUNCTION.RECOVERY.VALUE<1-9>}, чтобы указать на первую, вторую, третью и т. д. функцию в выражении триггера. См. indexed macros.

Макросы узла сети

{HOST.CONN}

IP-адрес узла сети или DNS-имя, в зависимости от настроек узла сети.<br> В глобальных скриптах, полях IP/DNS интерфейса и веб-сценариях макрос будет разрешаться в основной интерфейс агента. Если интерфейс агента не определен, будет использоваться основной интерфейс SNMP. Если интерфейс SNMP также не определен, будет использоваться основной интерфейс JMX. Если интерфейс JMX тоже не определен, будет использоваться основной интерфейс IPMI. Если у узла сети нет ни одного интерфейса, макрос разрешается в 'UNKNOWN'.<br><br> Этот макрос можно использовать с числовым индексом в виде {HOST.CONN<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{HOST.DESCRIPTION}

Описание узла сети.<br><br> Этот макрос можно использовать с числовым индексом в виде {HOST.DESCRIPTION<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{HOST.DNS}

DNS-имя узла сети.<br> В глобальных скриптах, полях IP/DNS интерфейса и веб-сценариях макрос будет разрешаться в основной интерфейс агента. Если интерфейс агента не определен, будет использоваться основной интерфейс SNMP. Если интерфейс SNMP также не определен, будет использоваться основной интерфейс JMX. Если интерфейс JMX тоже не определен, будет использоваться основной интерфейс IPMI. Если у узла сети нет ни одного интерфейса, макрос

разрешается в 'UNKNOWN'.<br><br> Этот макрос можно использовать с числовым индексом в виде {HOST.DNS<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{HOST.HOST}

Техническое имя узла сети.<br> Функции макросов не поддерживаются для этого макроса, если он используется как заполнитель в первом параметре функции истории, например, `last(/{HOST.HOST}/{ITEM.KEY})`.<br> {HOSTNAME<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос может использоваться с числовым индексом как {HOST.HOST<1-9>}, чтобы указывать на первый, второй, третий и т. д. узел сети в выражении триггера. См. индексированные макросы.

{HOST.ID}

ID узла сети.<br><br> Этот макрос можно использовать с числовым индексом в виде {HOST.ID<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{HOST.IP}

IP-адрес узла сети.<br> В глобальных скриптах, полях IP/DNS интерфейса и веб-сценариях макрос будет разрешаться в основной интерфейс агента. Если интерфейс агента не определен, будет использован основной интерфейс SNMP. Если интерфейс SNMP также не определен, будет использован основной интерфейс JMX. Если интерфейс JMX тоже не определен, будет использован основной интерфейс IPMI. Если у узла сети нет ни одного интерфейса, макрос разрешается в 'UNKNOWN'.<br> {IPADDRESS<1-9>} устарел и в поддерживаемых местах может не разрешаться.<br><br> Этот макрос можно использовать с числовым индексом в виде {HOST.IP<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{HOST.METADATA}

Метаданные узла сети.<br><br> Используется только для автоматической регистрации активного агента.

{HOST.NAME}

Отображаемое имя узла сети.<br><br> Этот макрос может использоваться с числовым индексом в виде {HOST.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{HOST.PORT}

Порт узла сети (агент).<br> В глобальных скриптах, полях IP/DNS интерфейса и веб-сценариях макрос будет разрешаться в основной интерфейс агента. Если интерфейс агента не определен, будет использоваться основной интерфейс SNMP. Если интерфейс SNMP также не определен, будет использоваться основной интерфейс JMX. Если интерфейс JMX тоже не определен, будет использоваться основной интерфейс IPMI. Если у узла сети нет ни одного интерфейса, макрос разрешается в 'UNKNOWN'.<br><br> Этот макрос можно использовать с числовым индексом в виде {HOST.PORT<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

Макросы целевого узла сети

{HOST.TARGET.CONN}

IP-адрес или DNS-имя целевого узла сети.

{HOST.TARGET.DNS}

DNS-имя целевого узла сети.

{HOST.TARGET.HOST}

Техническое имя целевого узла сети.

{HOST.TARGET.IP}

IP-адрес целевого узла сети.

{HOST.TARGET.NAME}

Отображаемое имя целевого узла сети.

Макросы группы узлов сети

{HOSTGROUP.ID}

Идентификатор группы узлов сети.<br><br>

Макросы инвентаризации узла сети

{INVENTORY.ALIAS}

Поле *Alias* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.ALIAS<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.ASSET.TAG}

Поле *Asset tag* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.ASSET.TAG<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.CHASSIS}

Поле *Chassis* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.CHASSIS<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.CONTACT}

Поле *Contact* в инвентаре узла сети.<br> {PROFILE.CONTACT<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.CONTACT<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.CONTRACT.NUMBER}

Поле *Номер договора* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.CONTRACT.NUMBER<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.DEPLOYMENT.STATUS}

Поле *Deployment status* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.DEPLOYMENT.STATUS<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HARDWARE}

Поле *Hardware* в инвентаре узла сети.<br> {PROFILE.HARDWARE<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HARDWARE<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HARDWARE.FULL}

Поле *Hardware (Full details)* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HARDWARE.FULL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HOST.NETMASK}

Поле *Маска подсети узла сети* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HOST.NETMASK<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HOST.NETWORKS}

Поле *Host networks* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HOST.NETWORKS<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HOST.ROUTER}

Поле *маршрутизатор узла сети* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HOST.ROUTER<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HW.ARCH}

Поле *Аппаратная архитектура* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HW.ARCH<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HW.DATE.DECOMM}

Поле *Дата вывода оборудования из эксплуатации* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HW.DATE.DECOMM<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HW.DATE.EXPIRY}

Поле *Дата окончания срока аппаратного обслуживания* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HW.DATE.EXPIRY<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HW.DATE.INSTALL}

Поле *Дата установки оборудования* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HW.DATE.INSTALL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.HW.DATE.PURCHASE}

Поле *Дата покупки оборудования* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.HW.DATE.PURCHASE<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.INSTALLER.NAME}

Поле *Installer name* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.INSTALLER.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.LOCATION}

Поле *Location* в инвентаре узла сети.<br> {PROFILE.LOCATION<1-9>} устарел и может не разрешаться в поддерживаемых locations.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.LOCATION<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.LOCATION.LAT}

Поле *Широта местоположения* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.LOCATION.LAT<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.LOCATION.LON}

Поле *Долгота местоположения* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.LOCATION.LON<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. индексированные макросы.

{INVENTORY.MACADDRESS.A}

Поле *MAC address A* в инвентарных данных узла сети.<br> {PROFILE.MACADDRESS<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.MACADDRESS.A<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.MACADDRESS.B}

Поле *MAC address B* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.MACADDRESS.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.MODEL}

Поле *Model* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.MODEL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.NAME}

Поле *Name* в инвентаре узла сети.<br> {PROFILE.NAME<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.NOTES}

Поле *Notes* в инвентаре узла сети.<br> {PROFILE.NOTES<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.NOTES<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.OOB.IP}

Поле *OOB IP address* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.OOB.IP<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.OOB.NETMASK}

Поле *маска подсети OOB* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.OOB.NETMASK<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.OOB.ROUTER}

Поле *OOB router* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.OOB.ROUTER<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.OS}

Поле *OS* в инвентаре узла сети.<br> {PROFILE.OS<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.OS<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.OS.FULL}

Поле *OS (Full details)* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.OS.FULL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.OS.SHORT}

Поле *OS (Short)* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.OS.SHORT<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.PRIMARY.CELL}

Поле *Основная ячейка POC* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.PRIMARY.CELL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. индексированные макросы.

{INVENTORY.POC.PRIMARY.EMAIL}

Поле *Основной email POC* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.PRIMARY.EMAIL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.PRIMARY.NAME}

Поле *Имя основного контактного лица* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.PRIMARY.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.PRIMARY.NOTES}

Поле *Примечания основного контактного лица* в инвентаре узла сети.<br><br> Этот макрос может использоваться с числовым индексом, например {INVENTORY.POC.PRIMARY.NOTES<1-9>}, чтобы указывать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.PRIMARY.PHONE.A}

Поле *Primary POC phone A* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.PRIMARY.PHONE.A<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.PRIMARY.PHONE.B}

Поле *Основной телефон POC B* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.PRIMARY.PHONE.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.PRIMARY.SCREEN}

Поле *Имя экрана основного контактного лица* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.PRIMARY.SCREEN<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.CELL}

Поле *Secondary POC cell* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.CELL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.EMAIL}

Поле *Secondary POC email* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.EMAIL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.NAME}

Поле *Имя второго контактного лица* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.NOTES}

Поле *Примечания вторичного контактного лица* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.NOTES<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.PHONE.A}

Поле *Secondary POC phone A* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.PHONE.A<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.PHONE.B}

Поле *Secondary POC phone B* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.PHONE.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.POC.SECONDARY.SCREEN}

Поле *Secondary POC screen name* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.POC.SECONDARY.SCREEN<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SERIALNO.A}

Поле *Серийный номер A* в инвентарных данных узла сети.<br> {PROFILE.SERIALNO<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SERIALNO.A<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SERIALNO.B}

Поле *Серийный номер B* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SERIALNO.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.ADDRESS.A}

Поле *Site address A* в инвентаре узла сети.<br><br> Этот макрос может использоваться с числовым индексом, например {INVENTORY.SITE.ADDRESS.A<1-9>}, чтобы указывать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.ADDRESS.B}

Поле *Адрес сайта B* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.ADDRESS.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.ADDRESS.C}

Поле *Site address C* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.ADDRESS.C<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.CITY}

Поле *Город местоположения* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.CITY<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.COUNTRY}

Поле *Страна расположения* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.COUNTRY<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.NOTES}

Поле *Примечания к сайту* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.NOTES<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.RACK}

Поле *Расположение стойки на площадке* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.RACK<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.STATE}

Поле *State/Province* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.STATE<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SITE.ZIP}

Поле *Почтовый индекс/ZIP* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SITE.ZIP<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE}

Поле *Software* в инвентаре узла сети.<br> {PROFILE.SOFTWARE<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE.APP.A}

Поле *Software application A* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE.APP.A<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE.APP.B}

Поле *Software application B* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE.APP.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE.APP.C}

Поле *Software application C* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE.APP.C<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE.APP.D}

Поле *Software application D* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE.APP.D<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE.APP.E}

Поле *Software application E* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE.APP.E<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.SOFTWARE.FULL}

Поле *Программное обеспечение (полные сведения)* в инвентарных данных узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.SOFTWARE.FULL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.TAG}

Поле *Tag* в инвентаре узла сети.<br> {PROFILE.TAG<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос может использоваться с числовым индексом, например {INVENTORY.TAG<1-9>}, чтобы указывать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.TYPE}

Поле *Type* в инвентаре узла сети.<br> {PROFILE.DEVICETYPE<1-9>} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.TYPE<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.TYPE.FULL}

Поле *Тип (полные сведения)* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.TYPE.FULL<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. индексированные макросы.

{INVENTORY.URL.A}

Поле *URL A* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.URL.A<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.URL.B}

Поле *URL B* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.URL.B<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.URL.C}

Поле *URL C* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.URL.C<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{INVENTORY.VENDOR}

Поле *Vendor* в инвентаре узла сети.<br><br> Этот макрос можно использовать с числовым индексом, например {INVENTORY.VENDOR<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

Макросы элемента данных

{ITEM.DESCRIPTION}

Описание N-го элемента данных в выражении триггера, которое вызвало уведомление.<br><br> Этот макрос можно использовать с числовым индексом в виде {ITEM.DESCRIPTION<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{ITEM.DESCRIPTION.ORIG}

Исходное описание (с неразрешенными макросами) N-го элемента данных в выражении триггера, которое вызвало уведомление.<br><br> Этот макрос можно использовать с числовым индексом в виде {ITEM.DESCRIPTION.ORIG<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{ITEM.ID}

Числовой ID N-го элемента данных в выражении триггера, которое вызвало уведомление.<br><br> Этот макрос можно использовать с числовым индексом в виде {ITEM.ID<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{ITEM.KEY}

Ключ N-го элемента данных в выражении триггера, которое вызвало уведомление.<br> Функции макросов не поддерживаются для этого макроса, если он используется как заполнитель в первом параметре функции истории, например, last(/{HOST.HOST}/{ITEM.KEY}).<br> {TRIGGER.KEY} устарел и может не разрешаться в поддерживаемых местах.<br><br> Этот макрос можно использовать с числовым индексом как {ITEM.KEY<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. индексированные макросы.

{ITEM.KEY.ORIG}

Исходный ключ (с неразвернутыми макросами) N-го элемента данных в выражении триггера, который вызвал уведомление.<br><br> Этот макрос можно использовать с числовым индексом в виде {ITEM.KEY.ORIG<1-9>}, чтобы указать на первый, второй, третий и т. д. узел сети в выражении триггера. См. indexed macros.

{ITEM.LASTVALUE}

Последнее значение N-го элемента данных в выражении триггера, которое вызвало уведомление.<br> В веб-интерфейсе оно будет разрешаться в *UNKNOWN*, если последнее значение из истории было собрано более чем *Max history display period* назад (задается в разделе меню Administration → General).<br> При использовании в имени проблемы макрос не будет подставлять последнее значение элемента данных при просмотре событий проблемы; вместо этого он сохранит значение элемента данных на момент возникновения проблемы.<br>При использовании в уведомлениях в некоторых случаях макрос может не подставлять последнее значение элемента данных в момент срабатывания триггера. Например, если элемент данных быстро получает два значения, "A" и "B", и триггер срабатывает для "A", в уведомлениях может отображаться "B" как последнее значение из-за небольшой задержки обработки - последнее значение элемента данных изменилось между моментом срабатывания триггера и моментом создания уведомления. Чтобы избежать этого, можно использовать макрос {ITEM.VALUE}, который подставляет значение на момент срабатывания триггера, обеспечивая использование правильного значения в уведомлении.<br> Это псевдоним для `last(/{HOST.HOST}/{ITEM.KEY})`.<br> Подставленное значение для текстовых/log-элементов данных обрезается до 20 символов веб-интерфейсом в следующих местах:<br>- Operational data;<br>- Trigger description;<br>- Trigger URLs;<br>- Trigger URL labels;<br>- Description of the item value widget.<br>Чтобы получить полное значение, можно использовать функции макросов, поскольку сервер не обрезает значения. Например: `{{ITEM.LASTVALUE}.regsub("(.*)", \1)}`<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LASTVALUE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LASTVALUE.AGE}

Время, прошедшее между последним сбором значения элемента данных и вычислением макроса.<br> Отображается в удобочитаемом формате (например, 1m 45s).<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LASTVALUE.AGE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LASTVALUE.DATE}

Дата, когда было собрано последнее значение элемента данных.<br> Отображается в формате `YYYYMMDD`.<br> Часовой пояс сервера используется в именах триггеров (в Monitoring -> Problems list), именах событий, а также в именах и значениях тегов. Во всех остальных случаях используется часовой пояс пользователя.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LASTVALUE.DATE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LASTVALUE.TIME}

Время, когда было собрано последнее значение элемента данных.<br> Отображается в формате `HHMMSS`.<br> Часовой пояс сервера используется в именах триггеров (в списке Monitoring -> Problems), именах событий, а также в именах и значениях тегов. Во всех остальных случаях используется часовой пояс пользователя.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LASTVALUE.TIME<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LASTVALUE.TIMESTAMP}

UNIX-метка времени, когда было собрано последнее значение элемента данных.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LASTVALUE.TIMESTAMP<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.AGE}

Возраст события журнала.<br>С точностью до секунды.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.AGE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.DATE}

Дата, когда запись журнала была записана в журнал.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.DATE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.EVENTID}

ID события в журнале событий.<br> Только для мониторинга журнала событий Windows.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.EVENTID<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.NSEVERITY}

Числовая степень серьезности события в журнале событий.<br> Только для мониторинга журнала событий Windows.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.NSEVERITY<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.SEVERITY}

Словесная серьезность события в журнале событий.<br> Только для мониторинга журнала событий Windows.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.SEVERITY<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.SOURCE}

Источник события в журнале событий.<br> Только для мониторинга журнала событий Windows.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.SOURCE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент в выражении триггера. См. indexed macros.

{ITEM.LOG.TIME}

Время, когда запись журнала была записана в журнал.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.TIME<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.LOG.TIMESTAMP}

UNIX-метка времени, когда запись журнала была записана в журнал.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.LOG.TIMESTAMP<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.NAME}

Имя элемента данных со всеми раскрытыми макросами.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.NAME.ORIG}

Исходное имя элемента данных (с неразрешенными макросами).<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.NAME.ORIG<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.STATE}

Последнее состояние N-го элемента данных в выражении триггера, которое вызвало уведомление.<br> Возможные значения: *Not supported*, *Normal*.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.STATE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.STATE.ERROR}

Сообщение об ошибке с подробностями о том, почему элемент данных стал неподдерживаемым.<br> Если элемент данных переходит в состояние неподдерживаемого и затем сразу снова становится поддерживаемым, поле ошибки может быть пустым.

{ITEM.VALUE}

Разрешается в одно из следующих значений:

- Историческое значение N-го элемента данных в выражении триггера на момент события, если используется в контексте изменения состояния триггера, например при отображении событий или отправке уведомлений;
- Последнее значение N-го элемента данных в выражении триггера, если используется вне контекста изменения состояния триггера, например при отображении списка триггеров во всплывающем окне выбора. В этом случае работает так же, как {ITEM.LASTVALUE}.

В первом случае будет разрешено в *UNKNOWN*, если историческое значение уже было удалено или никогда не сохранялось.<br>Во втором случае, и только на веб-интерфейсе, будет разрешено в *UNKNOWN*, если последнее значение из истории было собрано более чем *Max history display period* назад (задается в разделе меню Administration→General).<br>Разрешенное значение для текстовых/лог-элементов данных обрезается веб-интерфейсом до 20 символов в следующих местах:<br>- Оперативные данные;<br>- Описание триггера;<br>- URL триггера;<br>- Метки URL триггера;<br>- Описание виджета значения элемента данных.<br>Чтобы получить полное значение, можно использовать функции макросов, поскольку сервер не обрезает значения. Например: `{{ITEM.VALUE}.regsub("(.*)", \1)}`<br><br>Этот макрос можно использовать с числовым индексом, например {ITEM.VALUE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.VALUE.AGE}

Время, прошедшее между сбором значения элемента данных и вычислением макроса.<br> Отображается в удобочитаемом формате (например, 1m 45s).<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.VALUE.AGE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент в выражении триггера. См. indexed macros.

{ITEM.VALUE.DATE}

Дата, когда было собрано значение элемента данных.<br> Отображается в формате `YYYYMMDD`.<br> Часовой пояс сервера используется в названиях триггеров (в списке Monitoring -> Problems), названиях событий, а также в названиях и значениях тегов. Во всех остальных случаях используется часовой пояс пользователя.<br><br>Этот макрос может использоваться с числовым индексом, например {ITEM.VALUE.DATE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.VALUE.TIME}

Время, когда было собрано значение элемента данных.<br> Отображается в формате `HHMMSS`.<br> Часовой пояс сервера используется в названиях триггеров (в списке Monitoring -> Problems), названиях событий, а также в именах и значениях тегов. Во всех остальных случаях используется часовой пояс пользователя.<br><br>Этот макрос может использоваться с числовым индексом, например {ITEM.VALUE.TIME<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.VALUE.TIMESTAMP}

Метка времени UNIX, когда было собрано значение элемента данных.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.VALUE.TIMESTAMP<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

{ITEM.VALUETYPE}

Тип значения N-го элемента данных в выражении триггера, которое вызвало уведомление.<br> Возможные значения: *0* - Numeric float, *1* - Character, *2* - Log, *3* - Numeric unsigned, *4* - Text.<br><br> Этот макрос можно использовать с числовым индексом, например {ITEM.VALUETYPE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент данных в выражении триггера. См. indexed macros.

Макросы уведомлений низкоуровневого обнаружения

{LLDRULE.DESCRIPTION}

Описание правила низкоуровневого обнаружения, которое вызвало уведомление.

{LLDRULE.DESCRIPTION.ORIG}

Исходное описание (с неразрешенными макросами) правила низкоуровневого обнаружения, которое вызвало уведомление.

{LLDRULE.ID}

Числовой ID правила обнаружения низкого уровня, которое вызвало уведомление.

{LLDRULE.KEY}

Ключ правила обнаружения низкого уровня, которое вызвало уведомление.

{LLDRULE.KEY.ORIG}

Исходный ключ (с неразрешенными макросами) правила низкоуровневого обнаружения, которое вызвало уведомление.

{LLDRULE.NAME}

Имя правила обнаружения низкого уровня, которое вызвало уведомление.

{LLDRULE.NAME.ORIG}

Исходное имя (с неразрешенными макросами) правила низкоуровневого обнаружения, которое вызвало уведомление.

{LLDRULE.STATE}

Последнее состояние правила низкоуровневого обнаружения.<br> Возможные значения: *Not supported*, *Normal*.<br><br> Этот макрос можно использовать с числовым индексом, например {LLDRULE.STATE<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент в выражении триггера. См. indexed macros.

{LLDRULE.STATE.ERROR}

Сообщение об ошибке с подробностями о том, почему правило низкоуровневого обнаружения стало неподдерживаемым.<br> Если правило низкоуровневого обнаружения переходит в состояние неподдерживаемого, а затем сразу снова становится поддерживаемым, поле ошибки может быть пустым.

Макросы карты

{MAP.ID}

Идентификатор карты сети.

{MAP.NAME}

Имя карты сети.

Макросы прокси

{PROXY.DESCRIPTION}

Описание прокси.<br> Разрешается в одно из следующих значений: - Прокси N-го элемента данных в выражении триггера (в уведомлениях на основе триггеров). Здесь можно использовать индексированные макросы; - Прокси, который выполнил обнаружение (в уведомлениях обнаружения). Используйте здесь {PROXY.DESCRIPTION} без индексации; - Прокси, к которому зарегистрировался активный агент (в уведомлениях авто-регистрации). Используйте здесь {PROXY.DESCRIPTION} без индексации.

Этот макрос можно использовать с числовым индексом, например {PROXY.DESCRIPTION<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент в выражении триггера. См. индексированные макросы.

{PROXY.NAME}

Имя прокси.<br> Разрешается в одно из следующих значений: - Прокси N-го элемента данных в выражении триггера (в уведомлениях на основе триггеров). Здесь можно использовать индексированные макросы; - Прокси, который выполнил обнаружение (в уведомлениях обнаружения). Используйте здесь {PROXY.NAME} без индексации; - Прокси, к которому зарегистрировался активный агент (в уведомлениях авто-регистрации). Используйте здесь {PROXY.NAME} без индексации.

Этот макрос можно использовать с числовым индексом, например {PROXY.NAME<1-9>}, чтобы указать на первый, второй, третий и т. д. элемент в выражении триггера. См. индексированные макросы.

Макросы скриптов

{MANUALINPUT}

Значение ручного ввода, указанное пользователем во время выполнения скрипта.<br>

Макросы сервиса

{SERVICE.DESCRIPTION}

Описание сервиса с подставленными макросами.<br>

{SERVICE.ID}

Числовой идентификатор сервиса, который вызвал действие.<br>

{SERVICE.NAME}

Имя сервиса с подставленными макросами.<br>

{SERVICE.ROOTCAUSE}

Список событий проблем триггеров, которые привели к сбою службы, отсортированный по степени серьезности и имени узла сети.<br> Включает следующие сведения: имя узла сети, имя события, степень серьезности, возраст, теги службы и значения.

{SERVICE.TAGS}

Список тегов событий сервиса, разделенных запятыми.<br> Теги событий сервиса можно определить в разделе тегов конфигурации сервиса. Если тегов нет, расширяется в пустую строку.<br>

{SERVICE.TAGSJSON}

Массив JSON, содержащий объекты тегов событий службы.<br> Теги событий службы можно определить в разделе тегов конфигурации службы. Если тегов нет, разворачивается в пустой массив.

{SERVICE.TAGS.<tag name>}

Значение тега события сервиса, на которое ссылается имя тега.<br> Теги события сервиса можно определить в разделе тегов конфигурации сервиса.<br> Имя тега, содержащее неалфавитно-цифровые символы (включая неанглийские многобайтовые UTF-символы), должно быть заключено в двойные кавычки. Кавычки и обратные слэши внутри имени тега в кавычках должны быть экранированы обратным слэшем.

Макросы триггеров

{TRIGGER.DESCRIPTION}

Описание триггера.<br> Все макросы, поддерживаемые в описании триггера, будут подставлены, если {TRIGGER.DESCRIPTION} используется в тексте уведомления.<br> {TRIGGER.COMMENT} устарел и может не разрешаться в поддерживаемых местах.

{TRIGGER.EVENTS.ACK}

Количество подтвержденных событий для элемента карты в картах или для триггера, который сгенерировал текущее событие в уведомлениях.

{TRIGGER.EVENTS.PROBLEM.ACK}

Количество подтвержденных событий проблемы для всех триггеров без учета их состояния.

{TRIGGER.EVENTS.PROBLEM.UNACK}

Количество непризнанных событий проблемы для всех триггеров, независимо от их состояния.

{TRIGGER.EVENTS.UNACK}

Количество не подтвержденных событий для элемента карты на картах или для триггера, который сгенерировал текущее событие в уведомлениях.

{TRIGGER.EXPRESSION}

Выражение триггера.<br>

{TRIGGER.EXPRESSION.EXPLAIN}

Частично вычисленное выражение триггера.<br> Функции, основанные на элементах данных, вычисляются и заменяются результатами в момент генерации события, тогда как все остальные функции отображаются в том виде, в котором они записаны в выражении. Может использоваться для отладки выражений триггера.

{TRIGGER.EXPRESSION.RECOVERY}

Выражение восстановления триггера, если для *OK event generation* в настройке триггера установлено значение "Recovery expression"; в противном случае возвращается пустая строка.<br>

{TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}

Частично вычисленное выражение восстановления триггера.<br> Функции, основанные на элементах данных, вычисляются и заменяются результатами в момент генерации события, тогда как все остальные функции отображаются в том виде, в котором они записаны в выражении. Может использоваться для отладки выражений восстановления триггера.

{TRIGGER.HOSTGROUP.NAME}

Отсортированный (по SQL-запросу) список групп узлов сети, в которых определен триггер, разделенный запятой и пробелом.

{TRIGGER.ID}

Числовой ID триггера, который вызвал это действие.<br>

{TRIGGER.NAME}

Имя триггера с подставленными значениями макросов.<br> Обратите внимание, что {EVENT.NAME} можно использовать в действиях для отображения имени сработавшего события/проблемы с подставленными значениями макросов.

{TRIGGER.NAME.ORIG}

Исходное имя триггера (без раскрытия макросов).<br>

{TRIGGER.NSEVERITY}

Числовая серьезность триггера.<br> Возможные значения: *0* - Не классифицировано, *1* - Информация, *2* - Предупреждение, *3* - Средняя, *4* - Высокая, *5* - Катастрофа.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.ACK}

Количество подтвержденных событий проблем для триггеров в состоянии проблемы.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.UNACK}

Количество не подтвержденных событий проблем для триггеров в состоянии проблемы.

{TRIGGER.SEVERITY}

Имя уровня серьезности триггера.<br> Можно задать в *Administration > General > Trigger displaying options*.

{TRIGGER.STATE}

Последнее состояние триггера.<br> Возможные значения: *Unknown*, *Normal*.

{TRIGGER.STATE.ERROR}

Сообщение об ошибке с подробностями о том, почему триггер стал неподдерживаемым.<br> Если триггер переходит в неподдерживаемое состояние, а затем сразу снова становится поддерживаемым, поле ошибки может быть пустым.

{TRIGGER.STATUS}

Значение триггера на момент выполнения шага операции.<br> Возможные значения: *Problem*, *OK*.<br> {STATUS} устарел и может не разрешаться в поддерживаемых местах.

{TRIGGER.TEMPLATE.NAME}

Отсортированный (по SQL-запросу) список шаблонов, в которых определен триггер, разделенный запятыми и пробелами, или *UNKNOWN*, если триггер определен в узле сети.

{TRIGGER.URL}

URL триггера.<br>

{TRIGGER.URL.NAME}

Метка для URL триггера.<br>

{TRIGGER.VALUE}

Текущее числовое значение триггера.<br> Возможные значения: *0* - триггер находится в состоянии OK, *1* - триггер находится в состоянии Problem.

{TRIGGERS.ACK}

Количество подтвержденных триггеров для элемента карты, без учета состояния триггера.<br> Триггер считается подтвержденным, если подтверждены все его события проблемы.

{TRIGGERS.PROBLEM.ACK}

Количество подтвержденных триггеров с проблемой для элемента карты.<br> Триггер считается подтвержденным, если все его события проблемы подтверждены.

{TRIGGERS.PROBLEM.UNACK}

Количество непризнанных триггеров с проблемами для элемента карты.<br> Триггер считается непризнанным, если хотя бы одно из его событий проблемы не признано.

{TRIGGERS.UNACK}

Количество неподтвержденных триггеров для элемента карты, без учета состояния триггера.<br> Триггер считается неподтвержденным, если хотя бы одно из его событий проблемы не подтверждено.

Макросы имени пользователя

{USER.FULLNAME}

Имя, фамилия и имя пользователя, который добавил подтверждение события или запустил скрипт.<br>

{USER.NAME}

Имя пользователя, который запустил скрипт.<br>

{USER.SURNAME}

Фамилия пользователя, который запустил скрипт.<br>

{USER.USERNAME}

Имя пользователя, который запустил скрипт.<br> {USER.ALIAS} теперь устарел и может не разрешаться в поддерживаемых местах.

Нумерованные макросы

Синтаксис нумерованых макросов **{МАКРОС<1-9>}** может ссылаться только на N-й элемент данных, функцию или узел сети в поле *Выражение (Expression)* триггера:

- **{HOST.IP1}, {HOST.IP2}, {HOST.IP3}** раскрываются в IP адреса первого, второго и третьего узла сети из выражения триггера (если только оно содержит эти узлы сети).

- **{ITEM.VALUE1}, {ITEM.VALUE2}, {ITEM.VALUE3}** раскрываются в значение первого, второго и третьего элемента данных из выражения триггера на момент события (если только выражение триггера содержит эти элементы данных).
- **{FUNCTION.VALUE1}, {FUNCTION.VALUE2}, {FUNCTION.VALUE3}** раскрываются в значение первой, второй и третьей функции, относящейся к элементам данных, из выражения триггера на момент события (если только выражение триггера содержит эти функции).

В контексте триггеров нумерованые макросы всегда ссылаются на поле *Выражение (Expression)* конфигурации триггера, а не на *Выражение восстановления (Recovery expression)*. Например, в событии восстановления **{ITEM.VALUE2}** будет раскрыто в значение второго элемента данных из выражения проблемы на момент восстановления.

Макрос {HOST.HOST<1-9>} также поддерживается внутри макроса выражений {? (/ _ / , )} в **именах графиков**. Например, {? (/{HOST.HOST2}/ , )} в имени графика будет раскрываться в узел сети второго элемента данных графика.

**Warning:**
Нумерованные макросы не раскрываются в любом другом контексте, кроме упомянутых здесь случаев. В остальных случаях используйте макросы без номера (**{HOST.HOST}, {HOST.IP}** и т.п.).

#### 2. Пользовательские макросы, поддерживаемые по местоположению

Обзор

В этом разделе представлен список мест, где поддерживаются определяемые пользователем макросы.

**Note:**
Для *Действий*, *Сетевого обнаружения*, *Прокси* и всех мест, перечисленных в разделе *Другие места* этой страницы, поддерживаются только пользовательские макросы, определённые на глобальном уровне. В упомянутых местах макросы, определённые на уровне узла сети и на уровне шаблона, раскрываться не будут.

**Note:**
Чтобы настроить значения макросов (например, сократить или извлечь определённые подстроки), можно использовать функции макросов.

Действия

При настройке действий пользовательские макросы могут быть использованы в следующих полях:

| Место | | Несколько макросов/вперемешку с |
|---|---|---|
| Оповещения и команды на события триггеров | | да |
| Внутренние оповещения на события триггеров | | да |
| Оповещения об обновлении проблем | | да |
| Оповещения и команды на события услуг | | да |
| Оповещения об обновлении услуг | | да |
| Условие периода времени | | нет |
| *Операции* | | |
| | Длительность шага операции по умолчанию | нет |
| | Длительность шага | нет |

Узлы сети / прототипы узлов сети

При настройке узла сети и прототипа узла сети пользовательские макросы могут использоваться в следующих полях:

| Место | | Несколько макросов/вперемешку с текстом[1] |
|---|---|---|
| IP адрес / DNS имя интерфейса | | только DNS |
| Порт интерфейса | | нет |
| *SNMP v1, v2* | | |
| | SNMP community | да |
| *SNMP v3* | | |
| | Имя контекста | да |
| | Имя безопасности | да |

| Место | | Несколько макросов/вперемешку с текстом[1] |
|---|---|---|
| | Пароль аутентификации | да |
| | Ключевая фраза безопасности | да |
| *IPMI* | | |
| | Имя пользователя | да |
| | Пароль | да |
| *Теги*[2] | | |
| | Имена тегов | да |
| | Значения тегов | да |

Элементы данных / прототипы элементов данных

При настройке элементов данных или прототипов элементов данных пользовательские макросы могут использоваться в следующих полях:

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Имя элемента данных | да |
| Параметры ключа элемента данных | да |
| Интервал обновления | нет |
| Пользовательские интервалы | нет |
| Время ожидания (доступно для поддерживаемых типов элементов данных) | нет |
| Хранить до (для истории и динамики изменений) | нет |
| Описание | да |
| *Вычисляемый/агрегируемый элемент данных* | |
| Формула (константы выражения и параметры функции; параметры ключа элемента данных; (*только агрегируемый элемент данных*) условия фильтра (имя группы узлов сети и имя тега)) | да |
| *Монитор баз данных* | |
| Имя пользователя | да |
| Пароль | да |
| SQL запрос | да |
| *HTTP агент* | |
| URL[3] | да |
| Поля запроса | да |
| Тело запроса | да |

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Заголовки (имена и значения) | да |
| Требуемые коды состояния | да |
| HTTP прокси | да |
| Имя пользователя для HTTP аутентификации | да |
| Пароль для HTTP аутентификации | да |
| Файл SSL сертификата | да |
| Файл SSL ключа | да |
| Пароль к SSL ключу | да |
| Разрешённые хосты | да |
| *JMX агент* | |
| JMX endpoint | да |
| *Скриптовый элемент данных* | |
| Имена и значения параметров | да |
| *Элемент данных типа «Браузер»* | |
| Имена и значения параметров | да |
| *SNMP агент* | |
| SNMP OID | да |
| *SSH агент* | |
| Имя пользователя | да |
| Файл публичного ключа | да |
| Файл приватного ключа | да |
| Пароль | да |
| Выполняемый скрипт | да |
| *TELNET агент* | |
| Имя пользователя | да |
| Пароль | да |
| Выполняемый скрипт | да |
| *Zabbix траппер* | |
| Разрешённые хосты | да |
| *Теги*[2] | |
| Имена тегов | да |
| Значения тегов | да |
| *Шаги предобработки* | |
| Параметры (включая пользовательские скрипты) | да |
| Пользовательские параметры обработки ошибок (поля *Задать значением (Set value to)* и *Задать ошибку в (Set error to)*) | да |

Низкоуровневое обнаружение

При настройке правил низкоуровневого обнаружения пользовательские макросы могут использоваться в следующих полях:

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Параметры ключа | да |

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Интервал обновления | нет |
| Пользовательские интервалы | нет |
| Время ожидания (доступно для поддерживаемых типов элементов данных) | нет |
| Удаление потерянных ресурсов | нет |
| Деактивация потерянных ресурсов | нет |
| Описание | да |
| *SNMP агент* | |
| SNMP OID | да |
| *SSH агент* | |
| Имя пользователя | да |
| Файл публичного ключа | да |
| Файл приватного ключа | да |
| Пароль | да |
| Выполняемый скрипт | да |
| *TELNET агент* | |
| Имя пользователя | да |
| Пароль | да |
| Выполняемый скрипт | да |
| *Zabbix траппер* | |
| Разрешённые хосты | да |
| *Монитор баз данных* | |
| Имя пользователя | да |
| Пароль | да |
| SQL запрос | да |
| *JMX агент* | |
| JMX endpoint | да |
| *HTTP агент* | |
| URL[3] | да |
| Поля запроса | да |
| Тело запроса | да |
| Заголовки (имена и значения) | да |
| Требуемые коды состояния | да |
| Имя пользователя для HTTP аутентификации | да |
| Пароль для HTTP аутентификации | да |
| *Фильтры* | |
| Регулярное выражение | да |
| *Замещения* | |
| Фильтры: регулярное выражение | да |
| Операции: интервал обновления (для прототипов элементов данных) | нет |

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Операции: период хранения истории (для прототипов элементов данных) | нет |
| Операции: период хранения динамики изменений (для прототипов элементов данных) | нет |

Сетевое обнаружение

При настройке правил сетевого обнаружения пользовательские макросы могут использоваться в следующих полях:

| Место | | Несколько макросов/вперемешку с текстом[1] |
|---|---|---|
| Интервал обновления | | нет |
| *SNMP v1, v2* | | |
| | SNMP community | да |
| | SNMP OID | да |
| *SNMP v3* | | |
| | Имя контекста | да |
| | Имя безопасности | да |
| | Пароль аутентификации | да |
| | Ключевая фраза безопасности | да |
| | SNMP OID | да |

Прокси серверы

При настройке прокси серверов пользовательские макросы могут использоваться в следующих полях:

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| *Адрес для активных агентов* > *Порт* (если прокси принадлежит группе) | нет |
| Адрес интерфейса и порт (для пассивных прокси) | нет |
| Время ожидания по типам элементов данных | нет |

Группы прокси

При настройке групп прокси пользовательские макросы могут использоваться в следующих полях:

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Период аварийного переключения (Failover period) | нет |
| Минимальное количество прокси (Minimum number of proxies) | нет |

Шаблоны

При настройке шаблонов пользовательские макросы могут использоваться в следующих полях:

| Место | | Несколько макросов/вперемешку с текстом[1] |
|---|---|---|
| *Теги*[2] | | |
| | Имена тегов | да |
| | Значения тегов | да |

Триггеры

При настройке триггеров пользовательские макросы могут использоваться в следующих полях:

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Имя | да |

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Оперативные данные | да |
| Выражение (только в константах и параметрах функций; секретные макросы не поддерживаются) | да |
| Теги для совпадения | да |
| Имя записи в меню | да |
| URL записи в меню[3] | да |
| Описание | да |
| *Теги*[2] | |
| Имена тегов | да |
| Значения тегов | да |

Веб-сценарии

При настройке веб-сценариев пользовательские макросы могут использоваться в следующих полях:

| Место | | Несколько макросов/вперемешку с текстом[1] |
|---|---|---|
| Имя | | да |
| Интервал обновления | | нет |
| Агент | | да |
| HTTP прокси | | да |
| Переменные (только значения) | | да |
| Заголовки (имена и значения) | | да |
| *Шаги* | | |
| | Имя | да |
| | URL[3] | да |
| | Переменные (только значения) | да |
| | Заголовки (имена и значения) | да |
| | Время ожидания | нет |
| | Требуемая строка | да |
| | Требуемые коды состояния | нет |
| *Аутентификация* | | |
| | Пользователь | да |
| | Пароль | да |
| | Файл SSL сертификата | да |
| | Файл SSL ключа | да |
| | Пароль к SSL ключу | да |
| *Теги*[2] | | |
| | Имена тегов | да |
| | Значения тегов | да |

Другие места

Помимо перечисленных здесь мест, пользовательские макросы могут использоваться в следующих полях:

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| Глобальные скрипты (URL, скрипт, SSH, Tel-net, IPMI), включая текст подтверждения | да |
| Вебхуки | |
| Скрипт JavaScript | нет |
| Имя параметра скрипта JavaScript | нет |
| Значение параметра скрипта JavaScript | да |
| *Панели* | |
| Столбец данных типа *Текст* в виджете панели *Топ узлов сети* | да |
| Параметр *Описание* в виджетах панели *Значение элемента данных* и *Радиальный датчик* | да |
| Параметр *Текст* Первичной/Вторичной метки в виджете панели *Сотовый график* | да |
| Параметр *URL*[3] в виджете панели *URL* | да |
| Пользователи → Пользователи → Оповещения* | |
| Когда активен | нет |
| *Администрирование* → *Общие* → *Веб-интерфейс* | |
| Рабочее время | нет |
| *Администрирование* → *Общие* → *Время ожидания* | |
| Время ожидания по типам элементов данных | нет |
| *Администрирование* → *Общие* → *Коннекторы* | |
| URL | да |
| Имя пользователя | да |
| Пароль | да |
| Токен на предъявителя (Bearer token) | да |
| Время ожидания (Timeout) | нет |
| HTTP прокси | да |
| Файл SSL сертификата | да |
| Файл SSL ключа | да |
| Пароль к SSL ключу | да |

| Место | Несколько макросов/вперемешку с текстом[1] |
|---|---|
| *Оповещения* → *Способы оповещений* → *Шаблоны сообщений* | |
| Тема | да |
| Сообщение | да |
| *Оповещения* → *Способы оповещений* → *Скрипты* | |
| Параметры скрипта | да |
| *Оповещения* → *Способы оповещений* → *Способ оповещений* | |
| Поля *Имя пользователя* и *Пароль* для способа оповещений *Email* (когда параметр *Аутентификация* имеет значение "Имя пользователя и пароль"; рекомендуются секретные макросы) | да |

Полный список всех макросов, поддерживаемых в Zabbix, смотрите в разделе поддерживаемые макросы.

Примечания

[1] Если несколько макросов в поле или макросы вперемешку с текстом не поддерживаются для данного места, то один макрос должен заполнять целиком всё поле.

[2] Макросы, которые используются в именах и значениях тегов, раскрываются только во время процесса генерации событий.

[3] URL-ы, которые содержат секретные макросы, работать не будут, поскольку макросы в них будут раскрыты как «******».

### 7. Символы единиц измерения

Обзор

Работа с большими значениями, такими как «86400», «104857600» или «1000000», может быть сложной и может приводить к ошибкам. Для упрощения настройки и улучшения читабельности Zabbix поддерживает символы единиц измерения (суффиксы), которые выполняют функцию множителей значений.

Использование суффиксов может упростить, например, настройку выражений триггеров, делая их более простыми для понимания и поддержки.

Выражения триггера без суффиксов:

```
last(/host/system.uptime)<86400
avg(/host/system.cpu.load,600s)<10
last(/host/vm.memory.size[available])<20971520
```

Выражения триггера с суффиксами:

```
last(/host/system.uptime)<1d
avg(/host/system.cpu.load,10m)<10
last(/host/vm.memory.size[available])<20M
```

Суффиксы также могут упростить настройку других объектов — таких как ключи элементов данных, виджеты и т. д, и помочь отобразить значения элементов данных в удобном для восприятия формате.

> **Note:**
> Чтобы узнать, поддерживает ли поле конфигурации объекта суффиксы, всегда смотрите соответствующую страницу для настраиваемого оъекта.

Суффиксы времени

Веб-интерфейс Zabbix поддерживает следующие суффиксы времени в конфигурации сущностей:<br> **s** - секунды *(при использовании работает так же, как и необработанное значение)*;<br> **m** - минуты;<br> **h** - часы;<br> **d** - дни;<br> **w** - недели;<br> **M** - месяцы *(только для функций трендов)*;<br> **y** - годы *(только для функций трендов)*.

Суффиксы времени поддерживаются только для целых чисел. Например, 1h поддерживается, но 1,5h или 1.5h не поддерживается; вместо этого используйте 90m.

Суффиксы размера памяти

Zabbix поддерживает следующие суффиксы размера памяти:<br> **K** - килобайты<br> **M** - мегабайты<br> **G** - гигабайты<br> **T** - терабайты

Суффиксы значений элементов данных

Суффиксы также можно использовать для отображения числовых значений элементов данных в удобочитаемом формате.

Чтобы включить это, используйте один из следующих суффиксов в поле *Units* при настройке элемента данных:<br> **B** - байты;<br> **Bps** - байты в секунду;<br> **s** - секунды, отображаемые с использованием до трех наибольших ненулевых единиц времени;<br> **uptime** - прошедшее время в формате hh:mm:ss или N days, hh:mm:ss;<br> **unixtime** - метка времени Unix, формат yyyy.mm.dd hh:mm:ss.

К способу интерпретации и отображения этих суффиксов применяются следующие дополнительные правила:

- Для B и Bps Zabbix использует преобразование по основанию 2 (1K = 1024B) в соответствии со стандартом JEDEC.
- Для других единиц (например, Hz, W и т. д.) Zabbix использует преобразование по основанию 10 (1K = 1000).
- Для s (секунды):
  - Формат включает yyy mmm ddd hhh mmm sss ms; отображаются только до трех наибольших ненулевых единиц времени (например, 1M 10d 4h).
  - Если единица равна нулю и находится между двумя ненулевыми единицами, она опускается (например, 10d 56m вместо 10d 0h 56m).

При использовании *Units* к значениям элементов данных автоматически применяются следующие суффиксы-множители:<br> **K**, **M**, **G**, **T** - кило, мега, гига, тера;<br> **P**, **E**, **Z**, **Y** - пета, экса, зетта, йотта *(они применяются только во веб-интерфейсе)*.

Чтобы предотвратить преобразование единиц, используйте префикс ! (например, !B или !s).

Следующие примеры показывают, как полученные значения элементов данных преобразуются на основе указанных единиц:

```
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:**
До Zabbix 4.0 существовал жестко заданный список исключений для единиц, состоящий из `ms`, `rpm`, `RPM`, `%`. Этот список исключений устарел, поэтому правильный способ предотвратить преобразование таких единиц — использовать `!ms`, `!rpm`, `!RPM`, `!%`.

#### 8. Настройка периодов времени

Обзор

Чтобы указать период времени, должен использоваться следующий формат:

`d-d,hh:mm-hh:mm`

где эти символы означают следующее:

| Символ | Описание |
|---|---|
| *d* | День недели: 1 — понедельник, 2 — вторник ,... , 7 — воскресенье |
| *hh* | Часы: 00-24 |
| *mm* | Минуты: 00-59 |

Вы можете указать более одного периода времени, используя разделитель в виде точки с запятой (;):

`d-d,hh:mm-hh:mm;d-d,hh:mm-hh:mm...`

Если оставить период времени пустым, это будет эквивалентно 1-7,00:00-24:00, что является значением по умолчанию.

**Attention:**
Верхняя граница периодов времени не включается. Таким образом, если вы укажете период 09:00-18:00, то последней секундой в периоде будет 17:59:59.

Примеры

Рабочее время. С понедельника по пятницу с 9:00 до 18:00:

`1-5,09:00-18:00`

Рабочие часы плюс выходные. С понедельника по пятницу с 9:00 до 18:00, а также суббота и воскресенье с 10:00 до 16:00:

`1-5,09:00-18:00;6-7,10:00-16:00`

#### 9. Выполнение команд

Zabbix использует единый функционал для внешних проверок, пользовательских параметров, элементов данных system.run, пользовательских скриптов оповещений, удалённых команд и глобальных скриптов.

Этапы выполнения

**Note:**
По умолчанию, все скрипты в Zabbix выполняются с имольвованием оболочки *sh*, и оболочку по умолчанию изменить нельзя. Чтобы использовать другую оболочку, можно использовать следующий приём: создать файл скрипта и вызвать этот скрипт при выполнении команды.

Команда/скрипт выполняется одинаково как на Unix, так и на Windows платформах:

1. Zabbix (родительский процесс) создаёт конвейер для связи
2. Zabbix настраивает данный конвейер для вывода данных от создаваемого дочернего процесса
3. Zabbix создаёт дочерний процесс (выполняет команду/скрипт)
4. Создается новая группа процесса (в Unix) или задача (в Windows) для дочернего процесса
5. Zabbix считывает из конвейера данные до тех пор, пока не истечёт установленное время ожидания или до момента, когда прекратится запись с другой стороны (закрыты ВСЕ обработчики/файловые дескрипторы). Обратите внимание, что дочерний процесс может создать еще некоторое количество процессов и выйти до того, как эти процессы завершатся или будет закрыт обработчик/файловый дескриптор.
6. Если установленное время ожидания не было достигнуто, Zabbix ждёт завершения начального дочернего процесса или ждёт достижения установленного времени ожидания.
7. Если начальный дочерний процесс завершил свою работу и время ожидания не вышло, тогда Zabbix проверяет код завершения начального дочернего процесса и сравнивает его с 0 (ненулевое значение считается ошибкой выполнения, только для пользовательских скриптов оповещений, удалённых команд и пользовательских скриптов, выполняемых на Zabbix сервере и Zabbix прокси)
8. На этот момент подразумевается, что всё выполнено, и всё дерево процессов (т.е. группа процессов или задача) завершается.

**Attention:**
Zabbix предполагает, что команда/скрипт завершили обработку, в тот момент, когда завершился изначальный дочерний процесс И никакие другие процессы всё ещё не держат открытым обработчик/файловый дескриптор вывода. Когда обработка завершена, ВСЕ созданные процессы завершаются.

Все двойные кавычки и обратная косая черта в команде экранируются обратной косой чертой, и вся команда заключается в двойные кавычки.

Проверка кода завершения

Код завершения проверяется на соответствие следующим условиям:

- Только для пользовательских скриптов оповещений, удалённых команд и пользовательских скриптов, выполняемых на Zabbix сервере и Zabbix прокси.
- Любой код завершения, отличный от 0, считается ошибкой выполнения.
- Содержимое стандарного вывода ошибки и стандартного вывода собирается при ошибочных выполнениях и доступно в веб-интерфейсе (где отображается результат выполнения).
- Для удалённых команд, выполняемых на Zabbix агенте/прокси, дополнительная запись в журнале может создаваться, используя параметр LogRemoteCommands на агенте/прокси.

Возможные сообщения в веб-интерфейсе и записи в журналах при ошибочных командах/скриптах:

- Содержимое стандарного вывода ошибки и стандартного вывода при ошибочных выполнениях (если имеется).
- «Process exited with code: N». (при пустом выводе и отличном от 0 коде завершения).
- «Process killed by signal: N». (при завершении процесса сигналом, только на Linux).
- «Process terminated unexpectedly». (при завершении процесса по неизвестным причинам).

Смотрите также

- Внешние проверки
- Пользовательские параметры
- Элементы данных system.run
- Пользовательские скрипты оповещений
- Удалённые команды
- Глобальные скрипты

### 10. Совместимость версий

Поддерживаемые агенты

Чтобы быть совместимым с Zabbix 7.4, агент Zabbix не должен быть старее версии 1.4 и не должен быть новее 7.4.

Возможно, потребуется проверить конфигурацию более старых агентов, так как некоторые параметры были изменены, например параметры, связанные с журналированием для версий до 3.0.

Чтобы в полной мере использовать преимущества новейшей функциональности, метрик, улучшенной производительности и сниженного потребления памяти, используйте последнюю поддерживаемую версию агента.

**Примечания для Windows XP**

- В 32-разрядной версии Windows XP не используйте агенты Zabbix новее 6.0.x;
- В Windows XP/Server 2003 не используйте шаблоны агента новее Zabbix 4.0.x. Более новые шаблоны используют англоязычные счетчики производительности, которые поддерживаются только начиная с Windows Vista/Server 2008.

Поддерживаемые агенты 2

Более старые версии Zabbix agent 2 начиная с версии 4.4 совместимы с Zabbix 7.4; Zabbix agent 2 не должен быть новее 7.4.

Обратите внимание, что при использовании Zabbix agent 2 версий 4.4 и 5.0 для обновления неподдерживаемых элементов данных используется интервал по умолчанию в 10 минут.

Чтобы в полной мере воспользоваться последними возможностями, метриками, улучшенной производительностью и сниженным потреблением памяти, используйте последнюю поддерживаемую версию agent 2.

Поддерживаемые прокси Zabbix

Для полной совместимости с Zabbix 7.4 прокси должны соответствовать основной версии сервера. Только прокси Zabbix 7.4.x полностью совместимы с сервером Zabbix 7.4.x.

Устаревшие прокси поддерживаются частично: они по-прежнему могут собирать данные и выполнять скрипты, но не могут получать обновления конфигурации, такие как новые элементы данных.

В отношении сервера Zabbix прокси могут быть:

- *Текущими* (прокси и сервер имеют одну и ту же основную версию);
- *Устаревшими* (версия прокси старее версии сервера, но поддерживается частично);
- *Неподдерживаемыми* (версия прокси старее предыдущего LTS-релиза сервера *или* версия прокси новее основной версии сервера).

Примеры:

| Версия сервера | *Текущая* версия прокси | *Устаревшая* версия прокси | *Неподдерживаемая* версия прокси |
|---|---|---|---|
| 6.4 | 6.4 | 6.0, 6.2 | Старее 6.0; новее 6.4 |
| 7.0 | 7.0 | 6.0, 6.2, 6.4 | Старее 6.0; новее 7.0 |
| 7.2 | 7.2 | 7.0 | Старее 7.0; новее 7.2 |
| 7.4 | 7.4 | 7.0 | Старее 7.0; новее 7.4 |

Функциональность, поддерживаемая прокси:

| Версия прокси | Обновление данных | Обновление настроек | Задания |
|---|---|---|---|
| *Текущая* | Да | Да | Да |
| *Устаревшая* | Да | Нет | Удалённые команды (например, скрипты оболочки)<br>Незамедлительные проверки значений элементов данных (т.е. *Выполнить сейчас*)<br>Примечание: тестирование реального значения для предобработки не поддерживается. |
| *Неподдерживаемая* | Нет | Нет | Нет |

Оповещения об использовании несовместимых версий демона Zabbix будут журналироваться.

Поддерживаемые XML-файлы

Для импорта в Zabbix 7.4 поддерживаются XML-файлы не старше версии 1.8.

**Attention:**
В формате XML-экспорта зависимости триггеров хранятся только по имени. Если существует несколько триггеров с одинаковым именем (например, с разными уровнями серьезности и выражениями), между которыми определена зависимость, импортировать их невозможно. Такие зависимости необходимо вручную удалить из XML-файла и добавить снова после импорта.

## 11 Динамическая библиотека Zabbix sender для Windows

Обзор

В среде Windows приложения могут отправлять данные на сервер/прокси Zabbix, используя динамическую библиотеку Zabbix sender (zabbix_sender.dll) вместо запуска внешнего процесса (zabbix_sender.exe).

Для компиляции пользовательских приложений с zabbix_sender.dll требуются zabbix_sender.h и zabbix_sender.lib.

Получение

Есть два способа получить zabbix_sender.dll.

**1.** Скачайте файлы zabbix_sender.h, zabbix_sender.lib и zabbix_sender.dll в виде ZIP-архива.

При выборе параметров загрузки обязательно укажите "No encryption" в разделе *Encryption* и "Archive" в разделе *Packaging*. Затем скачайте Zabbix agent (не Zabbix agent 2).

Файлы zabbix_sender.h, zabbix_sender.lib и zabbix_sender.dll будут находиться в загруженном ZIP-архиве в каталоге `bin\dev`. Распакуйте файлы в нужное место.

**2.** Соберите zabbix_sender.dll из исходного кода (см. инструкции).

Динамическая библиотека с файлами для разработки будет находиться в каталоге `bin\winXX\dev`. Чтобы использовать ее, подключите заголовочный файл zabbix_sender.h и выполните линковку с библиотекой zabbix_sender.lib.

См. также

- пример простой утилиты Zabbix sender, реализованной с использованием динамической библиотеки Zabbix sender, чтобы продемонстрировать использование библиотеки;
- файл zabbix_sender.h с интерфейсными функциями динамической библиотеки Zabbix sender. Этот файл содержит документацию, объясняющую назначение каждой интерфейсной функции, ее аргументы и возвращаемое значение.

## 12. Обновление мониторинга услуг

**Обзор** В Zabbix 6.0 функционал мониторинга услуг была значительно переработан (см. Что нового в Zabbix 6.0.0 для просмотра списка изменений).

На этой странице описывается, как услуги и соглашения об уровне обслуживания (SLA), определённые в более ранних версиях Zabbix, изменяются во время обновления до Zabbix 6.0 или новее.

**Услуги** В более старых версиях Zabbix услуги имели два типа зависимостей: мягкие и жёсткие. После обновления все зависимости станут одинаковыми.

Если служба «Дочерняя служба» ранее была связана с «Родительской службой 1» через жёсткую зависимость и дополнительно с «Родительской службой 2» через мягкую зависимость, то после обновления «Дочерняя служба» будет иметь две родительские службы «Родительская служба 1» и «Родительская служба 2».

Сопоставление проблем и услуг на основе триггеров было заменено сопоставлением на основе тегов. В Zabbix 6.0 и новее диалог настройки услуги имеет новый параметр *Теги проблем*, который позволяет указать одну или несколько пар имени и значения тега для сопоставления проблем. Триггеры, ранее связанные с услугой, получат новый тег `ServiceLink`: `<  >:<  >` (значение тега будет усечено до 32 символов). Связанные услуги получат тег проблемы `ServiceLink` с тем же значением.

Правила вычисления состояния

«Алгоритм вычисления состояния» будет обновлён с использованием следующих правил:

- Без вычисления → Задать состояние в OK
- Проблема, если хотя бы одна дочерняя услуга в состоянии проблема → Самое критичное из дочерних услуг
- Проблема, если все дочерние услуги в состоянии проблема → Самое критичное, если все дочерние имеют проблемы

**Соглашения об уровне обслуживания (SLA )** Ранее целевые показатели SLA требовалось определять для каждой услуги отдельно. Начиная с Zabbix 6.0, SLA стало отдельным объектом, который содержит информацию о графике обслуживания, ожидаемом целевом уровне обслуживания (service level objective, SLO) и периодах простоя, которые необходимо исключать из расчёта. После настройки соглашение об уровне обслуживания можно назначить нескольким услугам с помощью тегов услуг.

Во время обновления:

- Идентичные SLA, определённые для каждой услуги, будут сгруппированы, и для каждой группы будет создан один объект SLA.
- Каждая затронутая услуга получит специальный тег SLA:<ID>, и тот же тег будет указан в параметре *Service tags* соответствующего объекта SLA.
- Время создания услуги (новый показатель в отчётах SLA) для существующих служб будет установлено в 01.01.2000 00:00.

### 13. Другие проблемы

Вход в систему и systemd

Мы рекомендуем создать пользователя *zabbix* как системного, то есть, без возможности входа в систему. Некоторые пользователи игнорируют эту рекомендацию и используют ту же самую учётную запись для входа в систему (например, используя SSH) на хост с работающим Zabbix. Это может привести к аварийному останову демона Zabbix при выходе из системы. В этом случае в журнале Zabbix сервера вы получите что-то наподобие следующего:

```
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
```

и в журнале Zabbix агента:

```
zabbix_agentd [27796]: [file:'log.c',line:238] lock failed: [22] Invalid argument
```

Это происходит из-за того, что по умолчанию настройка systemd в файле /etc/systemd/logind.conf равна RemoveIPC=yes. Когда вы выполняете выход из системы, ранее созданные компонентами Zabbix семафоры удаляются, что приводит к сбою.

Выдержка из документации по 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".
```

Имеется 2 способа решения этой проблемы:

1. (рекомендуемый) Перестаньте использовать аккаунт *zabbix* для чего-либо кроме процессов Zabbix, выделите отдельную учётную запись для остальных потребностей.
2. (не рекомендуемый) Задайте RemoveIPC=no в /etc/systemd/logind.conf и перезагрузите систему. Обратите внимание, что RemoveIPC является общесистемным параметром, его изменение повлияет на всю систему.

Использование веб-интерфейса Zabbix за прокси

Если веб-интерфейс Zabbix работает за прокси-сервером, cookie path (путь cookie) в файле конфигурации прокси необходимо переписать, чтобы он соответствовал reverse-proxied path (обратному пути прокси). См. примеры ниже. Если путь cookie не переписан, пользователи могут столкнуться с проблемами авторизации при попытке войти в веб-интерфейс Zabbix.

Пример конфигурации для nginx

```
# ..
location / {
# ..
proxy_cookie_path /zabbix /;
proxy_pass http://192.168.0.94/zabbix/;
# ..
```

Пример конфигурации для Apache

```
# ..
ProxyPass "/" http://host/zabbix/
ProxyPassReverse "/" http://host/zabbix/
ProxyPassReverseCookiePath /zabbix /
ProxyPassReverseCookieDomain host zabbix.example.com
# ..
```

### 14. Отличия между Zabbix агентом и Zabbix агентом 2

В данном разделе описываются различия между Zabbix агентом и Zabbix агентом 2.

| Параметр | Zabbix агент | Zabbix агент 2 |
|---|---|---|
| Язык программирования | C | Go с некоторыми частями на C |
| Демонизация | да | только с помощью systemd (да на Windows) |
| Поддерживаемые расширения | Пользовательские загружаемые модули на C. | Пользовательские плагины на Go. |
| *Требования* | | |
| Поддерживаемые платформы | Linux, IBM AIX, FreeBSD, NetBSD, OpenBSD, HP-UX, Mac OS X, Solaris: 9, 10, 11, Windows: все настольные и серверные версии, начиная с XP | Linux, Windows: все настольные и серверные версии, на которых может быть установлена поддерживаемая версия Go. |
| Поддерживаемые криптобиблиотеки | GnuTLS 3.1.18 и выше<br>OpenSSL 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x.<br>LibreSSL — протестирована с версиями 2.7.4, 2.8.2 (применяются определённые ограничения, см. страницу Шифрование). | Linux: OpenSSL 1.0.1 и выше.<br>MS Windows: OpenSSL 1.1.1 или выше.<br>В библиотеке OpenSSL должны быть включена поддержка PSK. LibreSSL не поддерживается. |
| *Процессы мониторинга* | | |
| Процессы | Отдельный процесс активных проверок для каждой записи сервера/прокси. | Один процесс с автоматически созданными потоками.<br>Максимальное количество потоков определяется переменной окружения GOMAXPROCS. |
| Метрики | **UNIX**: см. список поддерживаемых элементов данных.<br><br>**Windows**: см. список дополнительных элементов данных, специфичных для Windows. | **UNIX**: все метрики, поддерживаемые Zabbix агентом.<br>Кроме того, агент 2 предоставляет нативное решение для мониторинга с помощью Zabbix: Docker, Memcached, MySQL, PostgreSQL, Redis, systemd и других объектов мониторинга — см. полный список элементов данных, специфичных для агента 2.<br><br>**Windows**: все метрики, поддерживаемые Zabbix агентом, а также net.tcp.service* проверки для HTTPS, LDAP.<br>Кроме того, агент 2 предоставляет нативное решение для мониторинга с помощью Zabbix: PostgreSQL, Redis. |
| Параллелизм | Активные проверки для одного сервера выполняются последовательно. | Проверки из разных плагинов или множественные проверки в одном плагине могут выполняться параллельно. |
| Сторонние трапы | нет | да |
| *Дополнительный функционал* | | |
| Постоянное хранилище | нет | да |

| Параметр | Zabbix агент | Zabbix агент 2 |
|---|---|---|
| Постоянные файлы для элементов данных log*[] | да (только на Unix) | нет |
| Передача данных журнала | Может выполняться во время сбора данных журнала, чтобы освободить буфер. | Сбор данных журнала останавливается при заполнении буфера, поэтому параметр BufferSize должен быть равен как минимум MaxLinesPerSecond x 2. |
| Смена пользователя во время выполнения | да (только Unix-подобные системы) | нет (управляется systemd) |
| Настраиваемые пользователем шифры | да | нет |

**См. также:**

- *Описание процессов Zabbix*: Zabbix агент, Zabbix агент 2
- *Параметры настройки*: Zabbix агент UNIX / Windows, Zabbix агент 2 UNIX / Windows

### 15. Примеры экранирования

Обзор

На этой странице представлены примеры правильного экранирования при использовании регулярных выражений в различных контекстах.

> **Note:**
> При использовании конструктора триггерных выражений правильное экранирование в регулярных выражениях добавляется автоматически.

Примеры

**Пользовательский макрос с контекстом**

Регулярное выражение: `\.+\"[a-z]+<br>` Пользовательский макрос с контекстом: `{$MACRO:regex:"\.+\\"[a-z]+"}`

Замечания:

- обратные слэши не экранируются;
- кавычки экранируются.

**Функция макросов в параметре ключа элемента данных**

Регулярное выражение: `.+:(\d+)$<br>` Ключ элемента данных: `net.tcp.service[tcp,,"{{$ENDPOINT}.regsub(\".+:(\d+)$"`

Замечания: - регулярное выражение внутри функции макросов `regsub` заключается в двойные кавычки (из-за того, что содержит закрывающие скобки); - Символы кавычек, обрамляющие регулярное выражение, экранируются (из-за того, что весь третий параметр элемента данных заключается в кавычки); - третий параметр ключа элемента данных заключается в кавычки, поскольку он содержит запятую.

**Функция макросов низкоуровневого обнаружения (LLD)**

Регулярное выражение: `\.+\"([a-z]+)<br>` Макрос LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`

Замечания:

- обратные слэши не экранируются;
- кавычки экранируются.

**Функция макросов LLD внутри контекста пользовательского макроса**

Регулярное выражение: `\.+\"([a-z]+)<br>` Макрос LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Пользовательский макрос с контекстом: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"[a-z]+\", \1)}"}`

Замечания:

- экранирование обратных слэшей для LLD не меняется;
- при вставке макроса LLD в контекст пользовательского макроса нам нужно поместить это в строку:

1. Выражение макроса заключается в кавычки;
2. Кавычки экранируются; всего добавляется 3 новых обратных слэша.

**Строковый параметр функции (любой)**

В качестве примера используется `concat`.

Содержимое строки: `\.+\"[a-z]+<br>` Выражение: `concat("abc", "\\.\\\"[a-z]+")`

Замечание:

- Строковый параметр требует экранирования как для обратных слэшей, так и для кавычек.

**Функция макросов LLD внутри строкового параметра функции**

Регулярное выражение: `\.+\"([a-z]+)<br>` Макрос LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Выражение: `concat("abc, "{{#MACRO}.iregsub(\"\\.+\\\\\"([a-z]+)\", \\1)}")`

Замечания:

- Строковые параметры требуют экранирования как для обратных слэшей, так и для кавычек;
- Добавляется ещё один уровень экранирования, так как макрос будет раскрыт только после раскрытия кавычек в строке;

**Пользовательский макрос с контекстом внутри строкового параметра функции**

Регулярное выражение: `\.+\"[a-z]+<br>` Пользовательский макрос с контекстом: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Выражение: `concat("abc, "{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}")`

Замечания:

- Как и в предыдущем примере, необходим дополнительный уровень экранирования;
- Обратные слэши и кавычки экранируются только для экранирования верхнего уровня (поскольку это строковый параметр).

**Функция макросов LLD внутри контекста пользовательского макроса внутри функции**

Регулярное выражение: `\.+\"([a-z])+<br>` Макрос LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Пользовательский макрос с контекстом: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"([a-z]+)\", \1)}"}<br>` Выражение: `concat("abc, "{$MACRO:\"{{#MACRO}.iregsub(\\\"\.+\\\\\\\"([a-z]+)\\\", \\1)}\"}")`

Замечания по трём уровням экранирования:

1. Для функции макросов LLD, без экранирования обратных слэшей;
2. Для пользовательского макроса с контекстом, без экранирования обратных слэшей;
3. Для строкового параметра функции, с экранированием обратных слэшей.

**Пользовательский макрос с контекстом просто внутри строки**

Регулярное выражение: `\.+\"[a-z]+<br>` Пользовательский макрос с контекстом: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Внутри строки некоторого выражения, например: `func(arg1, arg2, arg3)="{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}"`

Замечания:

- Строки также требуют экранирования обратных слэшей;
- Строки также требуют экранирования кавычек;
- Снова случай с двумя уровнями экранирования:

1. Экранирование для контекста пользовательского макроса — без экранирования обратных слэшей;
2. Экранирование, поскольку это строка — с экранированием обратных слэшей.

## 23 Краткие справочные руководства

### Обзор

Этот раздел документации содержит краткие инструкции по настройке Zabbix для некоторых часто требуемых целей мониторинга.

Он предназначен для новых пользователей Zabbix и может использоваться как навигатор по другим разделам документации, которые содержат информацию, необходимую для решения задачи.

Доступны следующие краткие руководства:

- Мониторинг Linux с помощью агента Zabbix
- Мониторинг Windows с помощью агента Zabbix
- Мониторинг Apache через HTTP
- Мониторинг MySQL с помощью агента Zabbix 2
- Мониторинг VMware с помощью Zabbix
- Мониторинг сетевого трафика с помощью Zabbix
- Мониторинг сетевого трафика с помощью Zabbix с использованием активных проверок
- Мониторинг веб-сайтов с помощью элементов данных Browser
- Мониторинг сертификатов веб-сайтов с помощью агента Zabbix 2 (пассивно)
- Мониторинг сетевого коммутатора или маршрутизатора с помощью Zabbix
- Мониторинг журнала событий Windows с использованием активных проверок

### 1 Мониторинг Linux с помощью агента Zabbix

**Introduction** This page walks you through the steps required to start basic monitoring of Linux machines with Zabbix. The steps described in this tutorial can be applied to any Linux-based operating system.

**Who this guide is for**

This guide is designed for new Zabbix users and contains the minimum set of steps required to enable basic monitoring of your Linux machine. If you are looking for deep customization options or require more advanced configuration, see Configuration section of Zabbix manual.

**Prerequisites**

Before proceeding with this guide, you need to download and install Zabbix server and Zabbix frontend according to the instructions for your OS.

**Установка Zabbix agent** Zabbix agent — это процесс, отвечающий за сбор данных.

Проверьте версию вашего сервера Zabbix:

```
zabbix_server -V
```

Установите Zabbix agent той же версии (рекомендуется) на Linux-машину, которую вы хотите мониторить. В зависимости от ваших потребностей в мониторинге это может быть та же машина, на которой установлен сервер Zabbix, или совершенно другая машина.

Выберите наиболее подходящий способ установки:

- Запуск в виде Docker-контейнера — см. список доступных образов в репозитории Zabbix Docker.
- Установка из пакетов Zabbix (доступно для Alma Linux, CentOS, Debian, Oracle Linux, Raspberry Pi OS, RHEL, Rocky Linux, SUSE Linux Enterprise Server, Ubuntu).
- Сборка из исходных кодов.

**Настройка Zabbix для мониторинга** Агент Zabbix может собирать метрики в активном или пассивном режиме (одновременно).

> **Note:**
> Пассивная проверка — это простой запрос данных. Сервер Zabbix или прокси запрашивает некоторые данные (например, загрузку CPU), а агент Zabbix отправляет результат обратно на сервер. Активные проверки требуют более сложной обработки. Сначала агент должен получить от сервера(ов) список элементов данных для независимой обработки, а затем пакетно отправить данные обратно. Дополнительные сведения см. в разделе Пассивные и активные проверки агента.

Шаблоны мониторинга, предоставляемые Zabbix, обычно предлагают два варианта — шаблон для агента Zabbix и шаблон для агента Zabbix (active). В первом случае агент будет собирать метрики в пассивном режиме. Такие шаблоны обеспечивают одинаковые результаты мониторинга, но используют разные протоколы связи.

Дальнейшая настройка Zabbix зависит от того, выбираете ли вы шаблон для активных или пассивных проверок агента Zabbix.

**Passive checks** Агент Zabbix

1. Откройте файл конфигурации агента на машине, где установлен агент.

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

2. Добавьте IP-адрес или DNS-имя (и необязательный порт) вашего сервера Zabbix в параметр server. Например:

```
Server=192.0.2.1:10051
```

Агент Zabbix будет использовать этот адрес, чтобы принимать входящие подключения только от указанных серверов Zabbix или прокси для получения данных.

3. Перезапустите агент Zabbix.

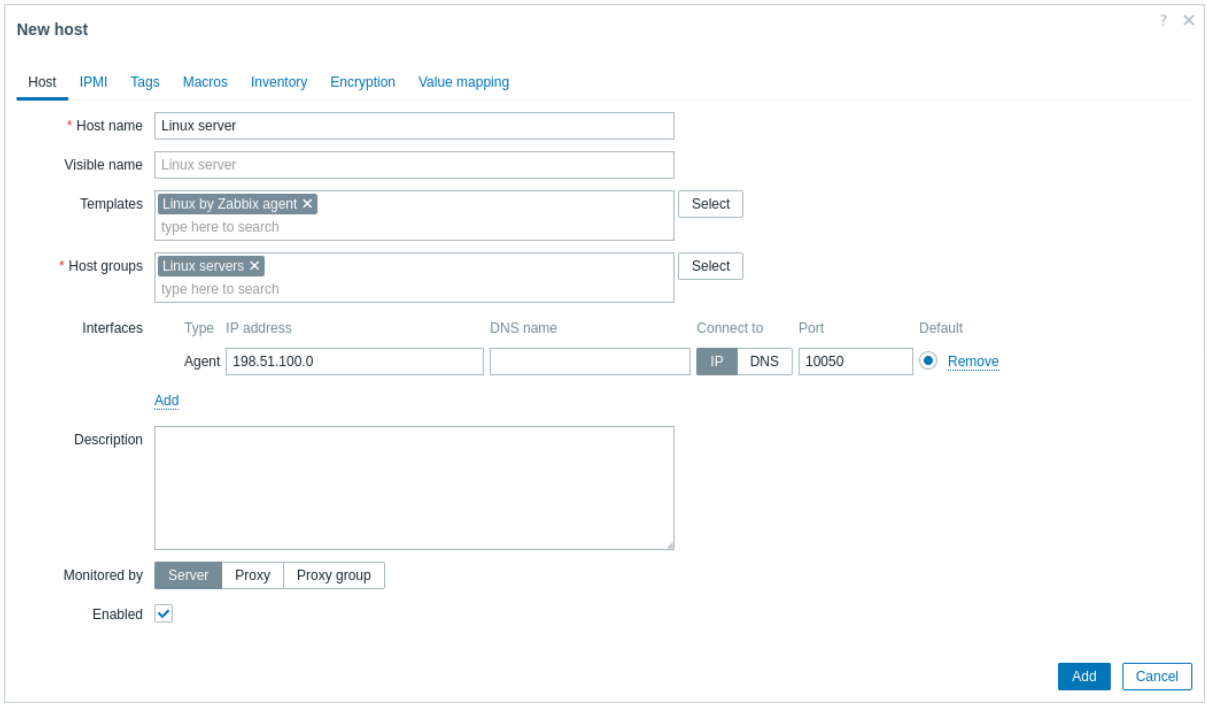
```
systemctl restart zabbix-agent
```

Веб-интерфейс Zabbix

1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix.

- В поле *Host name* введите имя узла сети (например, ”Linux server”).
- В поле *Templates* введите или выберите шаблон ”Linux by Zabbix agent”, который будет связан с узлом сети.
- В поле *Host groups* введите или выберите группу узлов сети (например, ”Linux servers”).
- В параметре *Interfaces* добавьте интерфейс *Agent* и укажите IP-адрес или DNS-имя Linux-машины, на которой установлен агент.




3. Нажмите *Add*, чтобы добавить узел сети. Этот узел сети будет представлять контролируемую Linux-машину.

**Active checks** Агент Zabbix

1. Откройте файл конфигурации агента на машине, где установлен агент.

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

2. Добавьте IP-адрес или DNS-имя (и необязательный порт) вашего сервера Zabbix в параметр ServerActive. Например:

```
ServerActive=192.0.2.1:10051
```

Агент Zabbix будет использовать этот адрес для подключения к trapper-порту сервера Zabbix (по умолчанию: 10051) и запроса данных конфигурации для активных проверок.

3. Укажите параметр Hostname, который должен совпадать с именем узла сети, заданным в веб-интерфейсе Zabbix. В нашем примере это:

```
Hostname=Linux server
```

Значение Hostname должно совпадать, поскольку для активных проверок агент Zabbix использует его для получения корректной конфигурации узла сети с сервера. В частности, агент инициирует подключение к серверу и идентифицирует себя с помощью значения Hostname. Затем сервер предоставляет конфигурацию мониторинга для этого узла сети. Если эти значения различаются, агент не получит соответствующую конфигурацию, что приведет к отсутствию метрик или проблемам с мониторингом.

4. Перезапустите агент Zabbix.

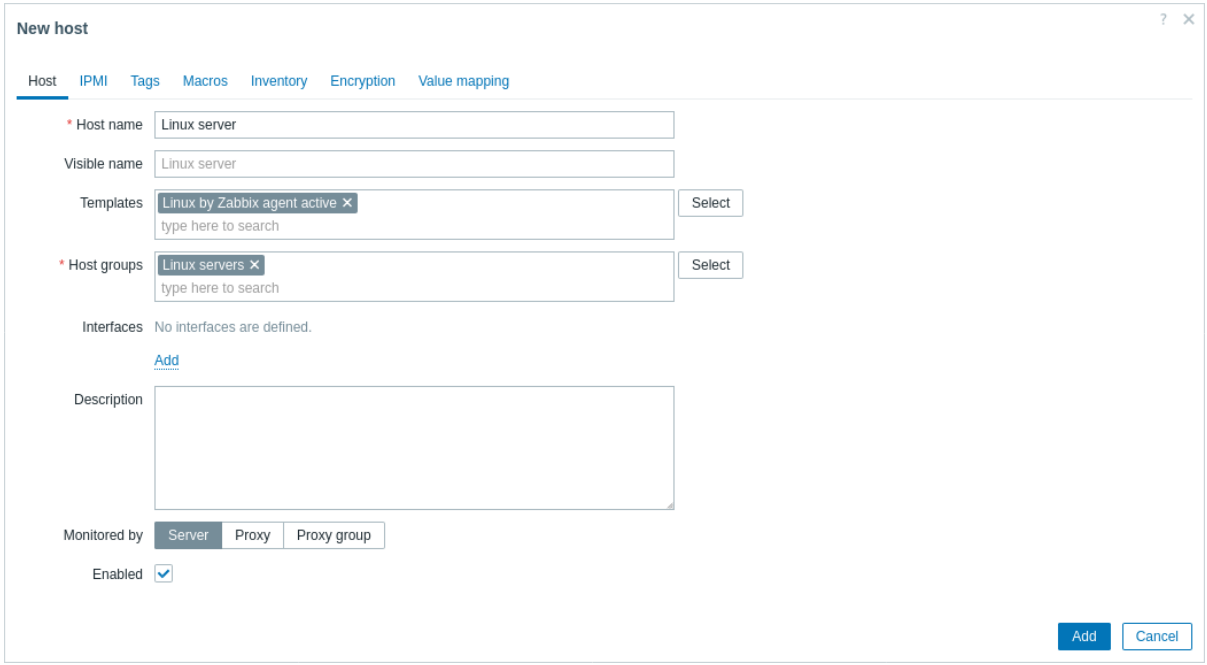
```
systemctl restart zabbix-agent
```

Веб-интерфейс Zabbix

1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix.

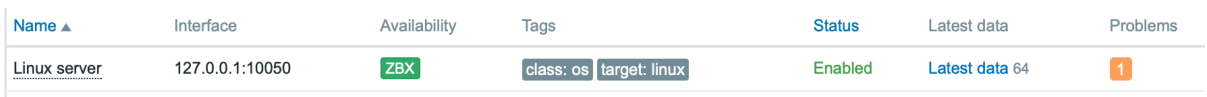
- В поле *Host name* введите имя узла сети (например, "Linux server"), которое соответствует значению параметра *Hostname*, ранее заданному в файле конфигурации агента.
- В поле *Templates* введите или выберите шаблон "Linux by Zabbix agent active", который будет связан с узлом сети.
- В поле *Host groups* введите или выберите группу узлов сети (например, "Linux servers").
- Оставьте *Interfaces* не заданным; интерфейс не нужен для активных проверок, поскольку агент инициирует соединение с сервером вместо ожидания соединения от сервера.



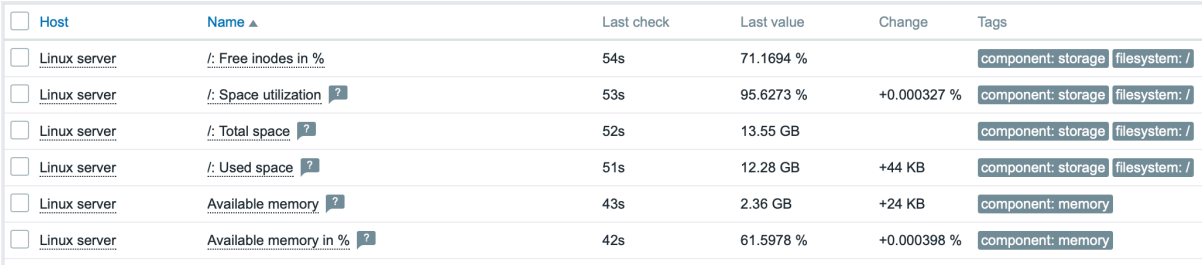

3. Нажмите *Add*, чтобы добавить узел сети. Этот узел сети будет представлять контролируемую машину Linux.

**Просмотр собранных метрик** Поздравляем! На этом этапе Zabbix уже отслеживает ваш компьютер с Linux.

Чтобы просмотреть собранные метрики, откройте раздел меню *Monitoring->Hosts* menu section и нажмите *Latest data* рядом с узлом сети.



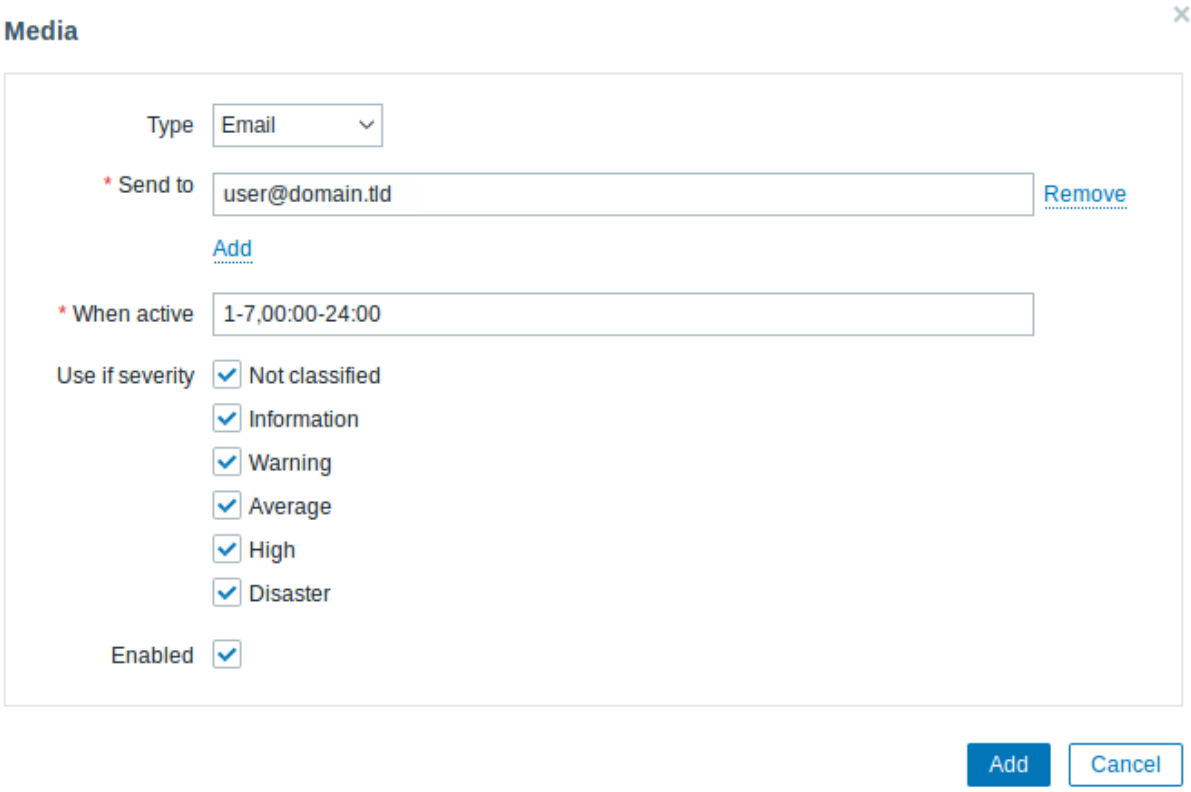

Это действие откроет список всех последних метрик, собранных с узла сети Linux server.



| Host | Name ▲ | Last check | Last value | Change | Tags |
|---|---|---|---|---|---|
| Linux server | /: Free inodes in % | 54s | 71.1694 % | | component: storage filesystem: / |
| Linux server | /: Space utilization | 53s | 95.6273 % | +0.000327 % | component: storage filesystem: / |
| Linux server | /: Total space | 52s | 13.55 GB | | component: storage filesystem: / |
| Linux server | /: Used space | 51s | 12.28 GB | +44 KB | component: storage filesystem: / |
| Linux server | Available memory | 43s | 2.36 GB | +24 KB | component: memory |
| Linux server | Available memory in % | 42s | 61.5978 % | +0.000398 % | component: memory |

**Настройка оповещений о проблемах** Zabbix может уведомлять вас о проблемах в вашей инфраструктуре с помощью различных методов.
Это руководство содержит шаги по настройке отправки оповещений по электронной почте.

1. Перейдите в *Настройки пользователя -> Профиль*, откройте вкладку *Media* и добавьте свой email.




2. Следуйте руководству по получению уведомлений о проблемах.

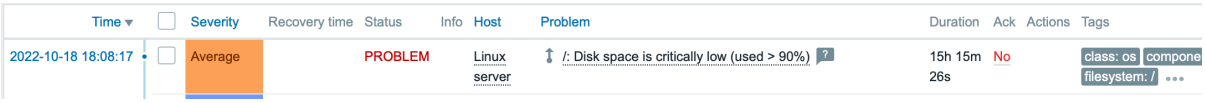
В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

**Проверьте вашу конфигурацию** В Linux вы можете имитировать высокую загрузку CPU и, как следствие, получить оповещение о проблеме, выполнив:

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

Возможно, вам потребуется запустить несколько процессов md5sum, чтобы загрузка CPU превысила пороговое значение.

Когда Zabbix обнаружит проблему, она появится в разделе Monitoring->Problems.



| Time ▼ | Severity | Recovery time | Status | Info | Host | Problem | Duration | Ack | Actions | Tags |
|---|---|---|---|---|---|---|---|---|---|---|
| 2022-10-18 18:08:17 | Average | | PROBLEM | | Linux server | /: Disk space is critically low (used > 90%) | 15h 15m 26s | No | | class: os compone filesystem: / ••• |

Если оповещения настроены, вы также получите уведомление о проблеме.

**See also:**

- Creating an item - how to start monitoring additional metrics (custom monitoring without templates).
- Zabbix agent items, Zabbix agent items for Windows - full list of metrics you can monitor using Zabbix agent on Windows.
- Problem escalations - how to create multi-step alert scenarios (e.g., first send message to the system administrator, then, if a problem is not resolved in 45 minutes, send message to the data center manager).

- Installation from packages - how to install Zabbix components using official RPM and DEB packages for various Linux distributions, ensuring access to the latest features and bug fixes.

## 2 Мониторинг Windows с помощью агента Zabbix

**Введение** На этой странице описаны шаги, необходимые для начала базового мониторинга машин Windows с помощью Zabbix.

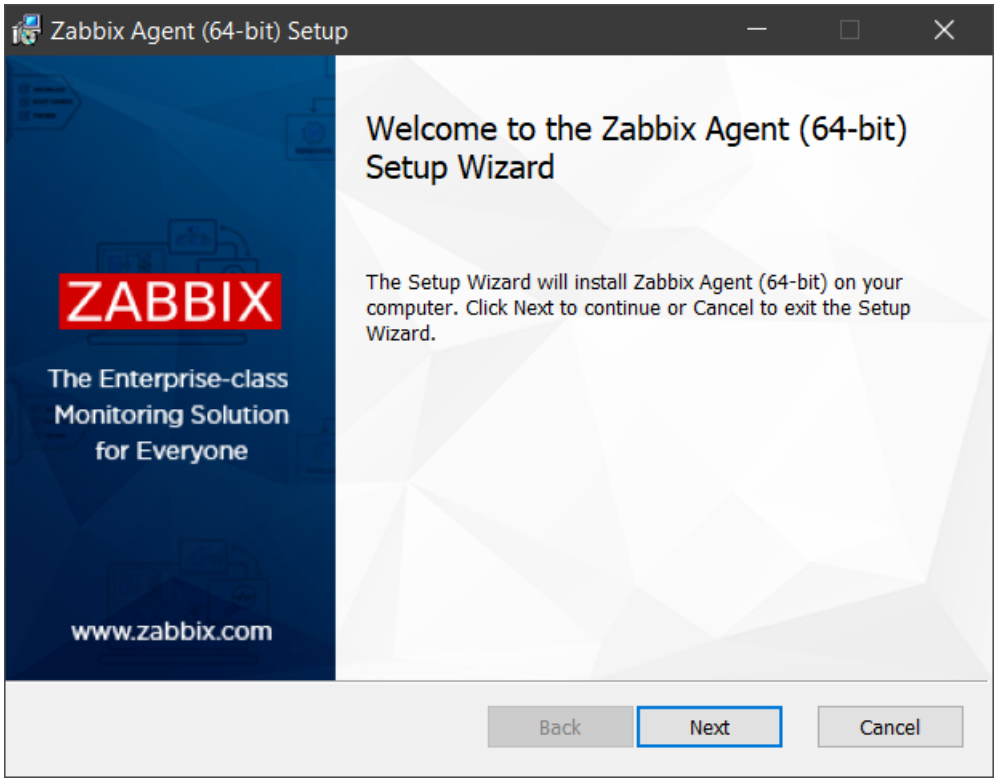
**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга вашей машины Windows. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. раздел Configuration руководства Zabbix.

**Предварительные требования**

Перед тем как продолжить это руководство по установке, вы должны скачать и установить сервер Zabbix и веб-интерфейс Zabbix в соответствии с инструкциями для вашей ОС.

**Установка Zabbix агента** Zabbix агент — это процесс, отвечающий за сбор данных. Его необходимо установить на компьютере с Windows, который вы хотите мониторить. Следуйте инструкциям по установке Zabbix агента для Windows.




**Настройка Zabbix для мониторинга** Агент Zabbix может собирать метрики в активном или пассивном режиме (одновременно).

**Note:**
Пассивная проверка — это простой запрос данных. Сервер Zabbix или прокси запрашивает некоторые данные (например, загрузку CPU), а агент Zabbix отправляет результат обратно на сервер. Активные проверки требуют более сложной обработки. Сначала агент должен получить от сервера(ов) список элементов данных для независимой обработки, а затем пакетно отправить данные обратно. Дополнительные сведения см. в разделе Пассивные и активные проверки агента.

Шаблоны мониторинга, предоставляемые Zabbix, обычно предлагают два варианта — шаблон для агента Zabbix и шаблон для агента Zabbix (active). В первом случае агент будет собирать метрики в пассивном режиме. Такие шаблоны обеспечивают одинаковые результаты мониторинга, но используют разные протоколы связи.

Дальнейшая настройка Zabbix зависит от того, выбираете ли вы шаблон для активных или пассивных проверок агента Zabbix.

**Пассивные проверки** Zabbix frontend

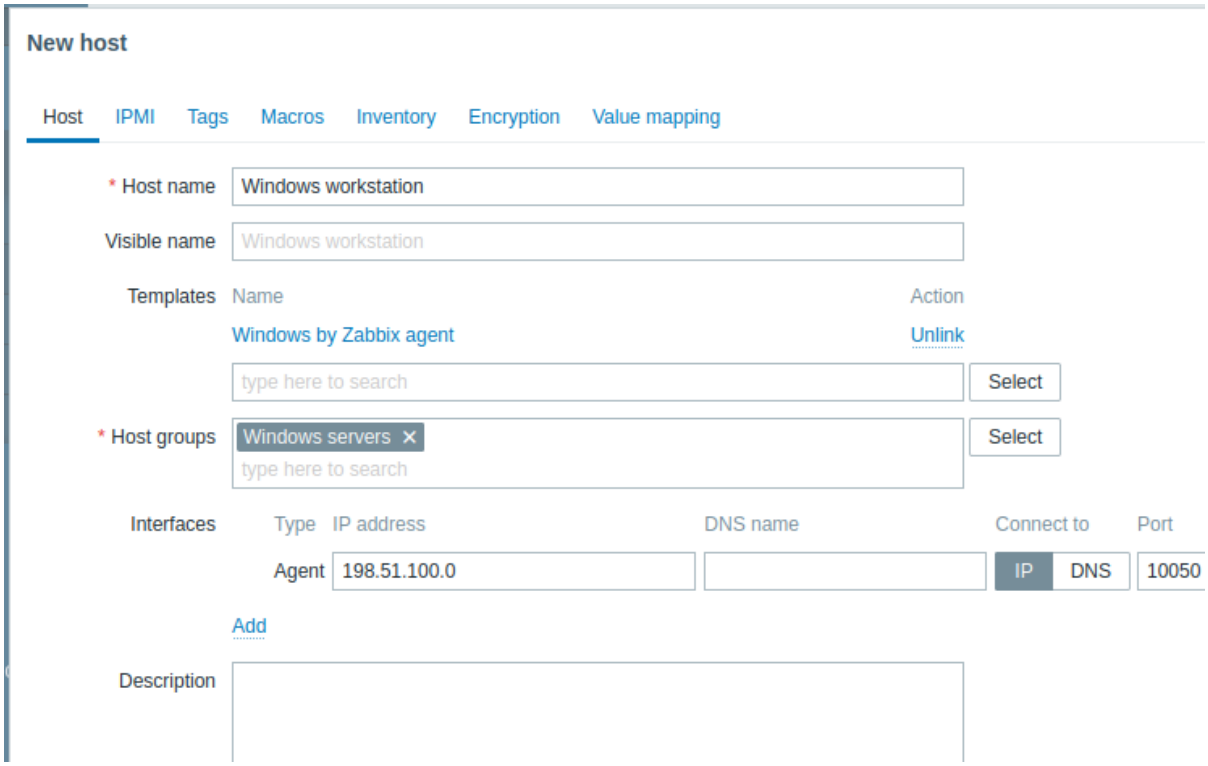
1. Log into Zabbix frontend.

2. Create a host in Zabbix web interface.

This host will represent your Windows machine.

3. In the *Interfaces* parameter, add *Agent* interface and specify the IP address or DNS name of the Windows machine where the agent is installed.

4. In the *Templates* parameter, type or select *Windows by Zabbix agent*.




Zabbix agent

For passive checks Zabbix agent needs to know the IP address or DNS name of Zabbix server. If you have provided correct information during the agent installation, the configuration file is already updated. Otherwise, you need to manually specify it. Go to the `C:\Program files\Zabbix Agent` folder, open the file *zabbix_agentd.conf* and add the IP/DNS of your Zabbix server to the *Server* parameter.

Example:

```
Server=192.0.2.22
```

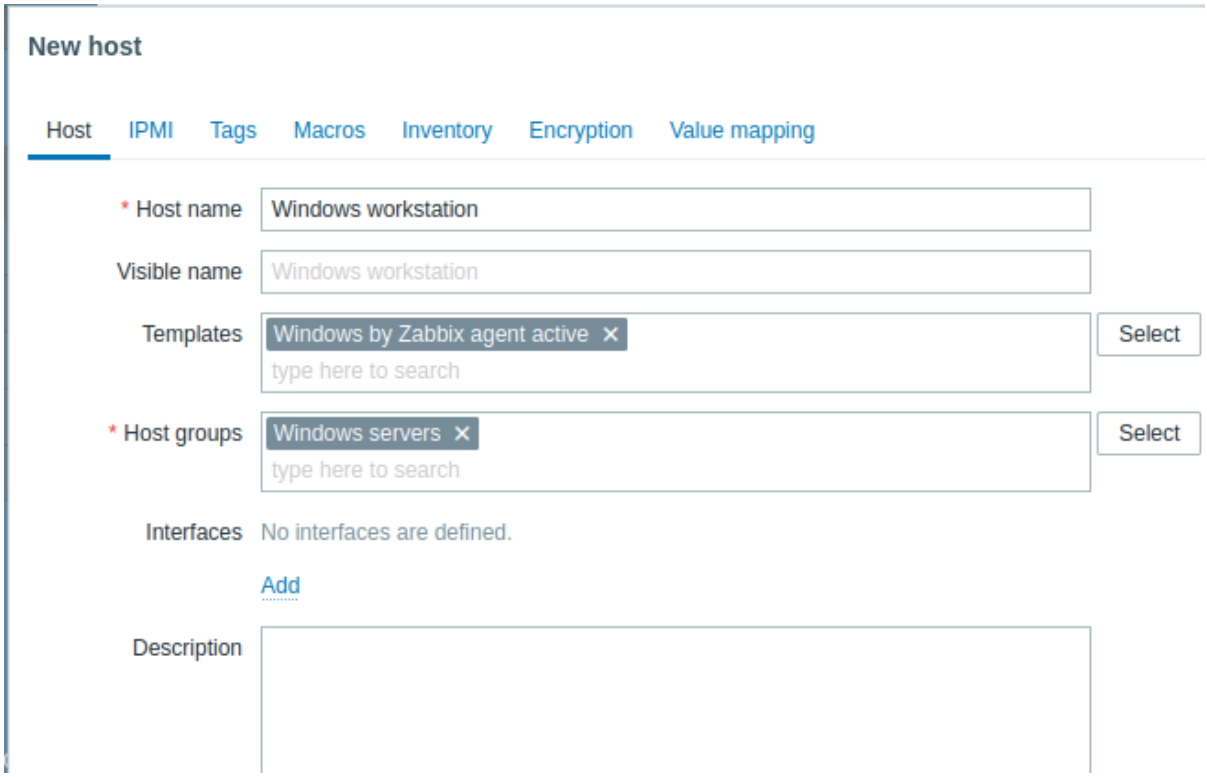
**Активные проверки** Веб-интерфейс Zabbix

1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix.

Этот узел сети будет представлять вашу машину Windows.

3. В параметре *Templates* введите или выберите *Windows by Zabbix agent active*.




Zabbix agent

In the `C:\Program files\Zabbix Agent` folder open the file *zabbix_agentd.conf* and add:

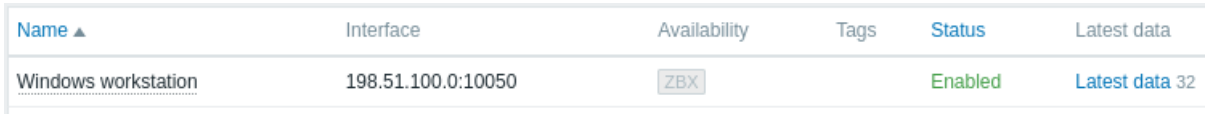
- The name of the host you created in Zabbix web interface to the *Hostname* parameter.
- The IP address or DNS name of your Zabbix server machine to the *ServerActive* parameter (might be prefilled if you have provided it during Zabbix agent setup).

Example:

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

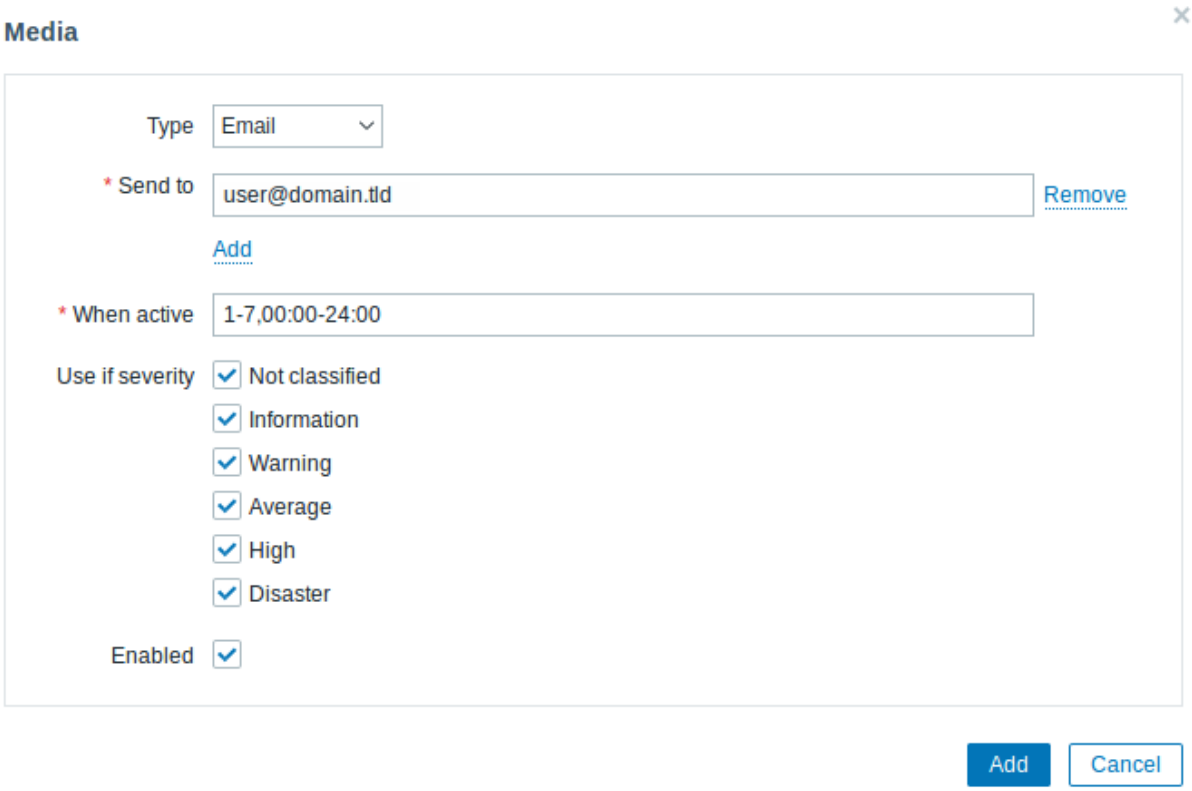
**Просмотр собранных метрик** Поздравляем! На этом этапе Zabbix уже отслеживает ваш компьютер под управлением Windows.

Чтобы просмотреть собранные метрики, откройте раздел меню *Monitoring->Hosts* menu section и нажмите *Latest data* рядом с узлом сети.




**Настройка оповещений о проблемах** Zabbix может уведомлять вас о проблемах в вашей инфраструктуре с помощью различных методов. В этом руководстве приведены шаги по настройке отправки оповещений по электронной почте.

1. Перейдите в *User settings -> Profile*, откройте вкладку *Media* и добавьте свой адрес электронной почты.




2. Следуйте руководству по получению уведомлений о проблемах.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

**Note:**
В Windows можно использовать утилиту CpuStres для имитации высокой загрузки CPU и, как следствие, получения оповещения о проблеме.

**См. также:**

- Создание элемента данных - как начать мониторинг дополнительных метрик (пользовательский мониторинг без шаблонов).
- Элементы данных Zabbix agent, Элементы данных Zabbix agent для Windows - полный список метрик, которые можно отслеживать с помощью Zabbix agent в Windows.
- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, а затем, если проблема не решена в течение 45 минут, отправить сообщение менеджеру дата-центра).

### 3 Мониторинг Apache через HTTP

**Введение** На этой странице показан быстрый и простой способ начать мониторинг веб-сервера Apache без установки какого-либо дополнительного программного обеспечения.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга вашей установки Apache. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. раздел Configuration в руководстве Zabbix.

**Предварительные требования**

Перед тем как продолжить это руководство по установке, необходимо загрузить и установить сервер Zabbix и веб-интерфейс Zabbix в соответствии с инструкциями для вашей ОС.

**Подготовка Apache** 1. Проверьте, какую версию Apache вы используете:

В системе на базе RHEL выполните:

```
httpd -v
```

В Debian/Ubuntu выполните:

```
apache2 -v
```

2. Убедитесь, что модуль Status включен в вашем экземпляре Apache.

В системе на базе RHEL выполните:

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

В Debian/Ubuntu выполните:

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

Если в списке нет status_module, включите модуль, выполнив:

В системе на базе RHEL выполните:

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

В Debian/Ubuntu выполните:

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

3. Отредактируйте файл конфигурации Apache, чтобы разрешить доступ к отчетам о состоянии с IP-адреса сервера Zabbix.

В системе на базе RHEL: `/etc/httpd/conf.modules.d/status.conf`:

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

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

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

Добавьте в файл следующие строки (**замените 198.51.100.255** на IP-адрес вашего сервера Zabbix):

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

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

4. Перезапустите Apache

В системе на базе RHEL выполните:

```
sudo systemctl restart httpd
```

В Debian/Ubuntu выполните:

```
sudo systemctl restart apache2
```

5. Чтобы проверить, что все настроено правильно, выполните (**замените 198.51.100.255** на IP-адрес вашего сервера Zabbix):

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

Ответ должен содержать статистику веб-сервера Apache.

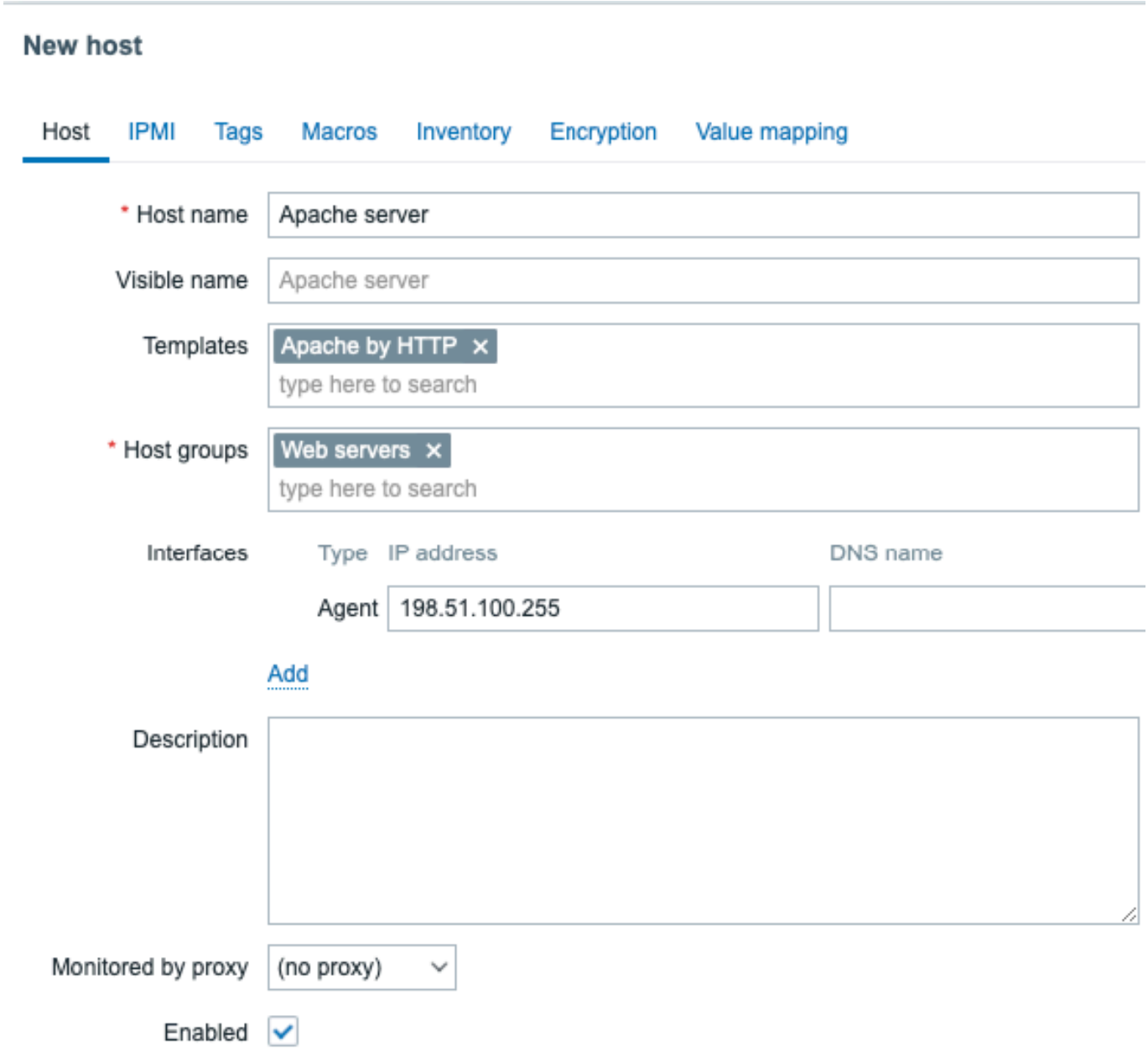
**Настройка Zabbix для мониторинга** 1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix.

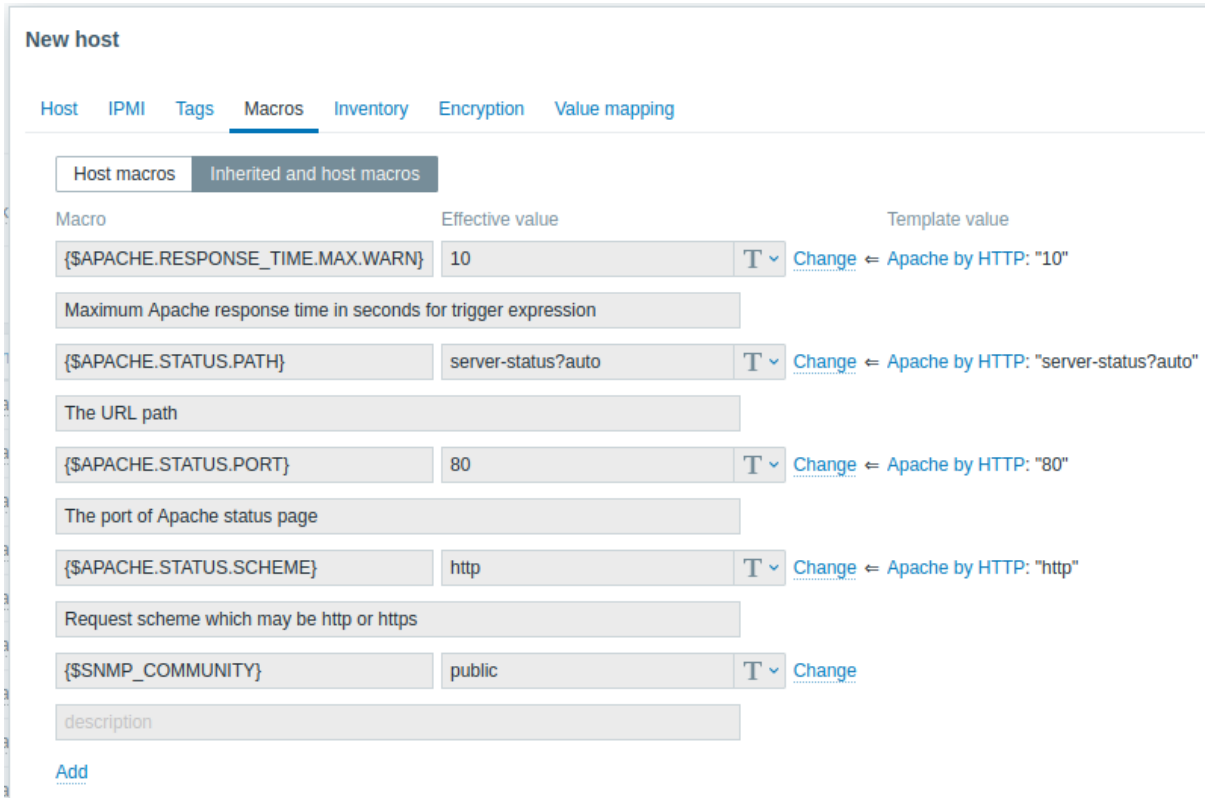
Этот узел сети будет представлять ваш сервер Apache.

3. В параметре *Interfaces* добавьте интерфейс *Agent* и укажите IP-адрес вашего экземпляра Apache. **Вам не нужно устанавливать агент Zabbix на машине**, интерфейс будет использоваться только для разрешения макроса {HOST.CONN}. Этот макрос используется в элементах данных шаблона для определения местоположения экземпляра Apache.

4. В параметре *Templates* введите или выберите *Apache by HTTP*.




5. Перейдите на вкладку **Macros** и выберите режим *Inherited and host macros*. Проверьте, что значения макросов {$APACHE.STATUS.PORT} и {$APACHE.STATUS.SCHEME} соответствуют настройкам вашей установки. По умолчанию порт — 80, а схема — http. Измените значения макросов, если вы используете другой порт и/или схему.

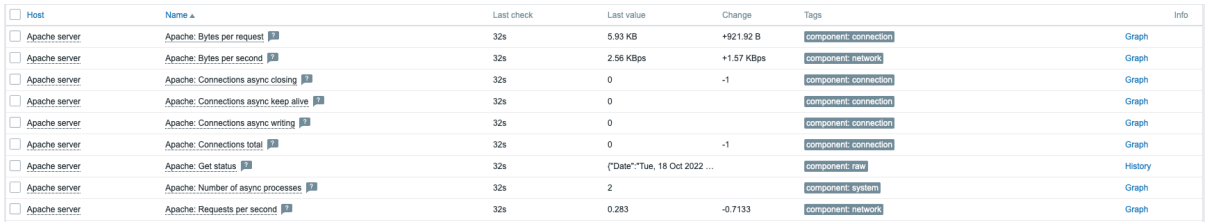



**Просмотр собранных метрик** Поздравляем! На этом этапе Zabbix уже отслеживает ваш веб-сервер Apache.

Чтобы просмотреть собранные метрики, откройте раздел меню *Monitoring->Hosts* menu section и нажмите *Dashboards* рядом с узлом сети.

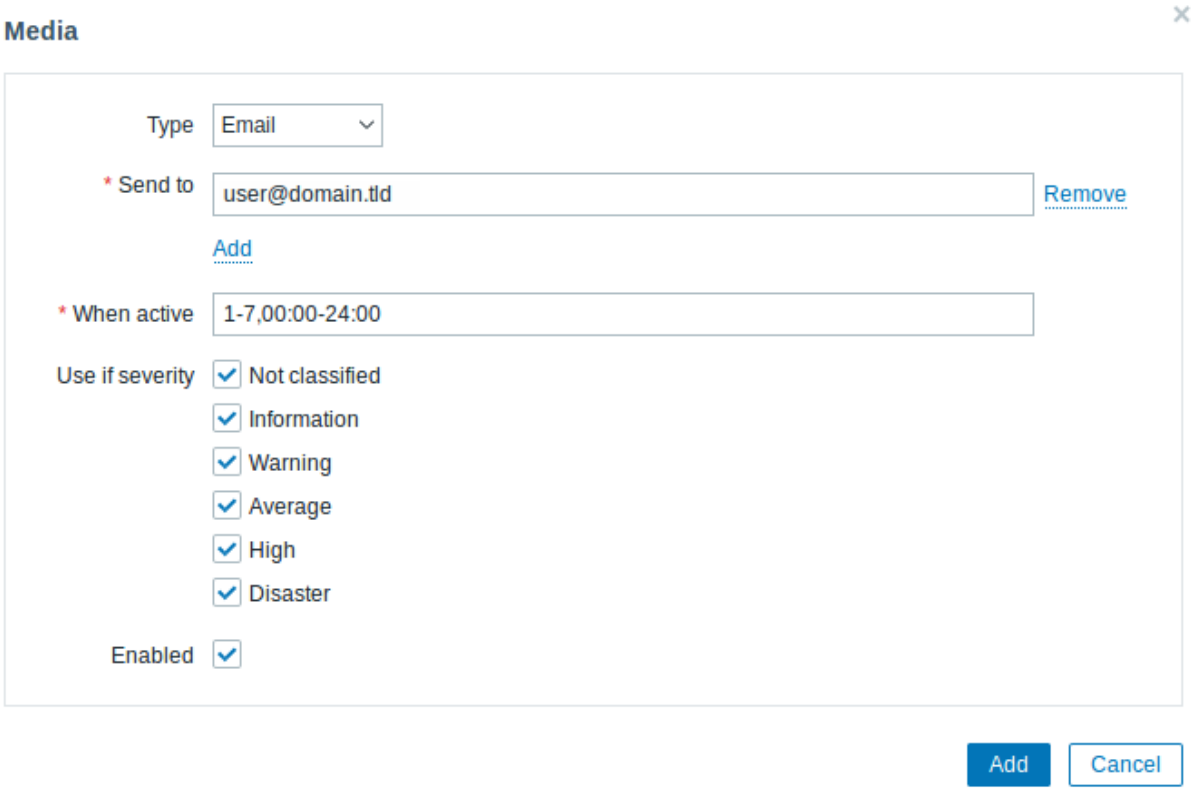
Это действие откроет панель узла сети с наиболее важными метриками, собранными со страницы Apache /server-status.

Кроме того, в *Monitoring->Hosts* можно нажать *Latest data*, чтобы просмотреть все последние собранные метрики в виде списка.



**Настройка оповещений о проблемах** Zabbix может уведомлять вас о проблемах в вашей инфраструктуре с помощью различных методов.
Это руководство содержит шаги по настройке отправки оповещений по электронной почте.

1. Перейдите в *User settings -> Profile*, откройте вкладку *Media* и добавьте свой email.




2. Следуйте руководству по Receiving problem notification.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

**Проверьте вашу конфигурацию** Чтобы смоделировать реальную проблему и получить тестовое оповещение о проблеме:

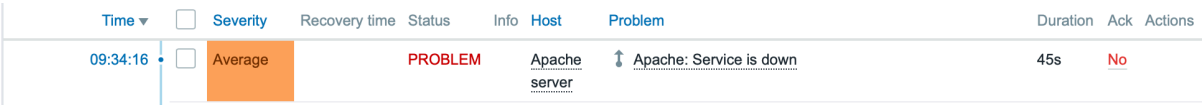
1. Откройте конфигурацию узла сети *Apache server* в Zabbix.

2. Перейдите на вкладку Macros и выберите *Inherited and host macros*.

3. Нажмите *Change* рядом с макросом {$APACHE.STATUS.PORT} и задайте другой порт.

4. Нажмите *Update*, чтобы сохранить конфигурацию узла сети.

5. Через несколько минут Zabbix обнаружит проблему *Apache service is down*, потому что теперь он не может подключиться к экземпляру. Она появится в разделе Monitoring->Problems.




Если оповещения настроены, вы также получите уведомление о проблеме.

6. Измените значение макроса обратно, чтобы устранить проблему и продолжить мониторинг Apache.

**См. также:**

- Усиление защиты веб-сервера - рекомендуемые настройки для повышения безопасности веб-сервера.
- Создание элемента данных - как начать мониторинг дополнительных метрик.
- Элементы данных HTTP - как отслеживать пользовательские метрики с помощью HTTP-агента.
- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, а затем, если проблема не решена в течение 45 минут, отправить сообщение менеджеру центра обработки данных).

#### 4 Мониторинг MySQL с помощью агента Zabbix 2

Введение

На этой странице вы пройдете через шаги, необходимые для начала базового мониторинга сервера MySQL.

Для мониторинга сервера MySQL существует несколько подходов: Zabbix agent, Zabbix agent 2 или стандарт Open Database Connectivity (ODBC). Основное внимание в этом руководстве уделено мониторингу сервера MySQL с помощью Zabbix agent 2, который является **рекомендуемым** подходом благодаря простой настройке в различных конфигурациях. Однако на этой странице также приведены инструкции для других подходов, поэтому вы можете выбрать тот, который лучше всего соответствует вашим требованиям.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга сервера MySQL. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. раздел Configuration руководства Zabbix.

**Prerequisites**

Before proceeding with this guide, you need to download and install Zabbix server, Zabbix frontend and Zabbix agent 2 according to the instructions for your OS.

Depending on your environment, some steps in this guide may slightly differ. This guide is based on an environment running Ubuntu.

Создание пользователя MySQL

Для мониторинга сервера MySQL Zabbix требуется доступ к нему и его процессам. В вашей установке MySQL уже есть пользователь с необходимым уровнем доступа (пользователь "zabbix", который был создан при установке Zabbix), однако у этого пользователя больше привилегий, чем необходимо для простого мониторинга (привилегии на DROP баз данных, DELETE записей из таблиц и т. д.). Поэтому для *только* мониторинга сервера MySQL необходимо создать отдельного пользователя MySQL.

1. Подключитесь к клиенту MySQL, создайте пользователя "zbx_monitor" (замените *<password>* для пользователя "zbx_monitor" на пароль по вашему выбору) и GRANT необходимые привилегии этому пользователю:

```
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;
```

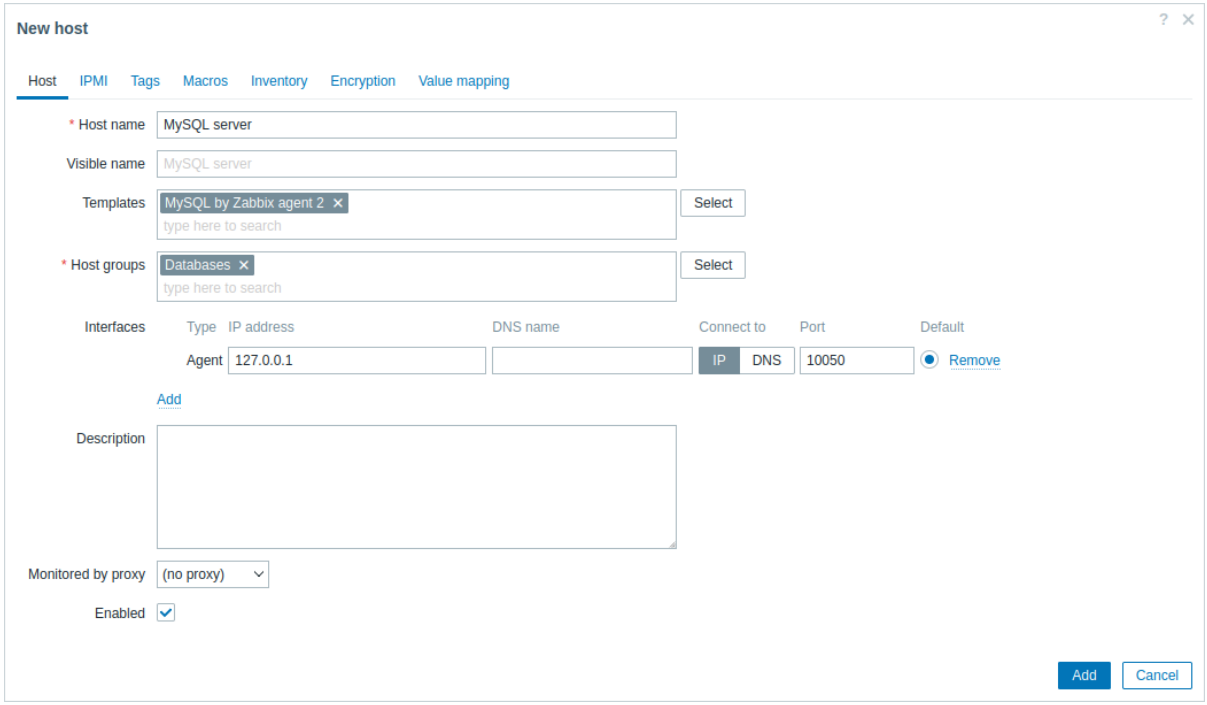
После создания пользователя можно переходить к следующему шагу.

Настройка веб-интерфейса Zabbix

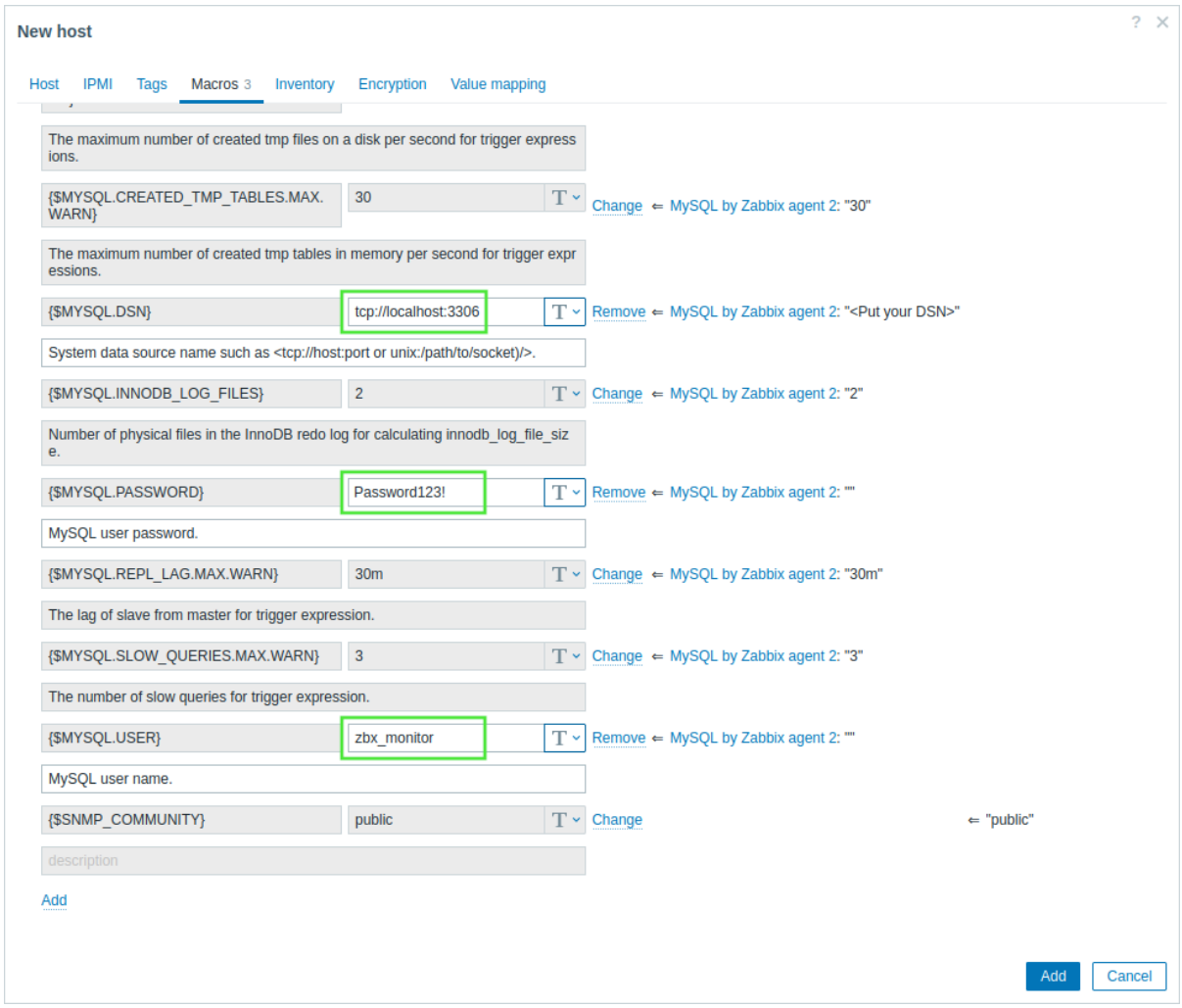
1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix:

- В поле *Host name* введите имя узла сети (например, "MySQL server").
- В поле *Templates* введите или выберите шаблон "MySQL by Zabbix agent 2", который будет связан с узлом сети.
- В поле *Host groups* введите или выберите группу узлов сети (например, "Databases").
- В поле *Interfaces* добавьте интерфейс типа "Agent" и укажите IP-адрес вашего сервера MySQL. В этом руководстве используется "127.0.0.1" (localhost) для мониторинга сервера MySQL, который установлен на той же машине, что и сервер Zabbix и Zabbix agent 2.




- На вкладке *Macros* переключитесь на *Inherited and host macros*, найдите следующие макросы и нажмите *Change* рядом со значением макроса, чтобы обновить его:
  - {$MYSQL.DSN} - задайте источник данных сервера MySQL ( строку подключения именованной сессии из файла конфигурации плагина MySQL Zabbix agent 2). В этом руководстве используется источник данных по умолчанию "tcp://localhost:3306" для мониторинга сервера MySQL, который установлен на той же машине, что и сервер Zabbix и Zabbix agent 2.
  - {$MYSQL.PASSWORD} - задайте пароль ранее созданного пользователя MySQL "zbx_monitor".
  - {$MYSQL.USER} - задайте имя ранее созданного пользователя MySQL "zbx_monitor".

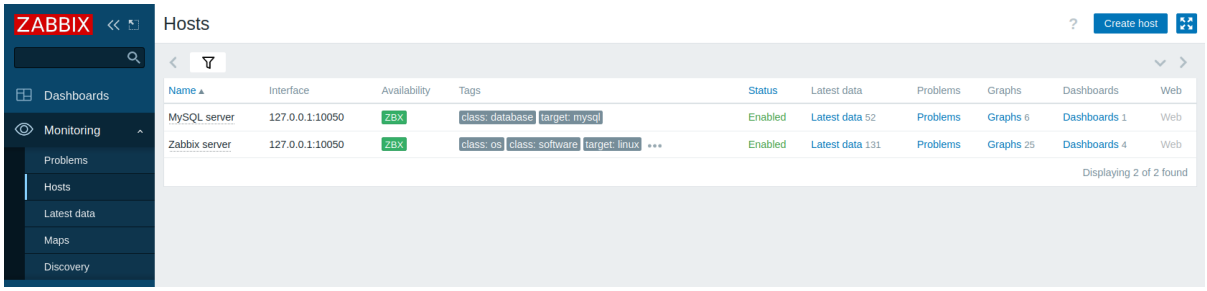


3. Нажмите *Add*, чтобы добавить узел сети. Этот узел сети будет представлять ваш сервер MySQL.

Просмотр собранных метрик

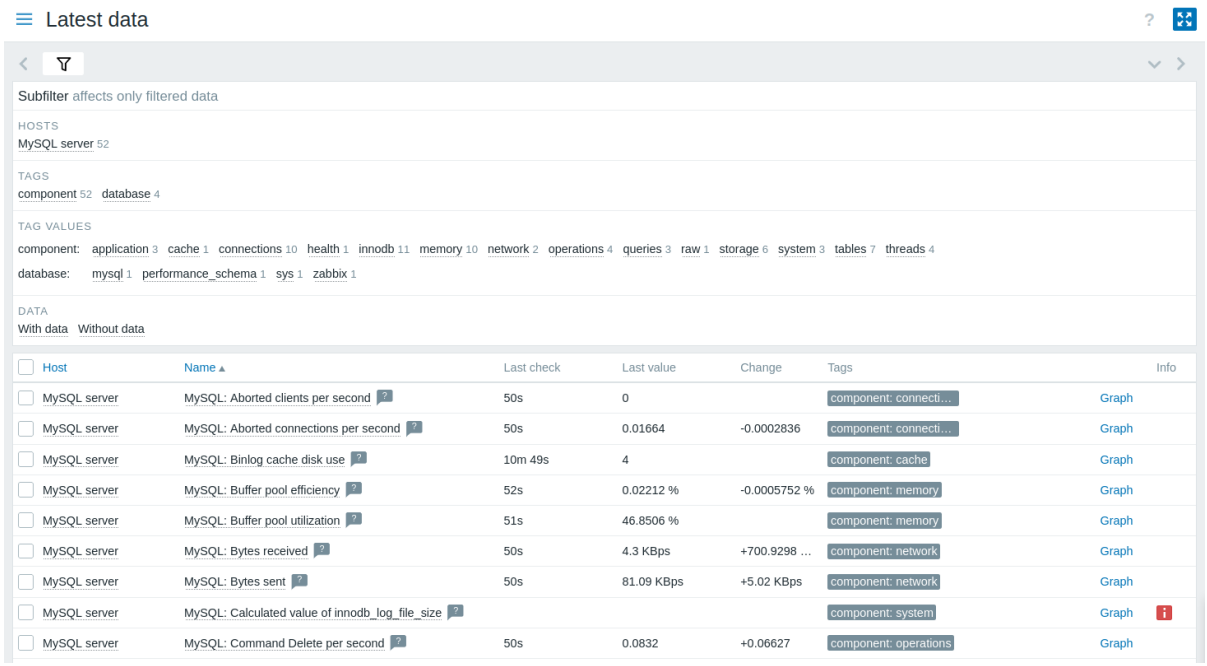
Поздравляем! На этом этапе Zabbix уже выполняет мониторинг вашего сервера MySQL.

Чтобы просмотреть собранные метрики, перейдите в раздел меню *Мониторинг → Узлы сети* и нажмите *Дашборды* рядом с узлом сети.




Это действие откроет дашборд узла сети (настроенный на уровне шаблона) с наиболее важными метриками, собранными с сервера MySQL.

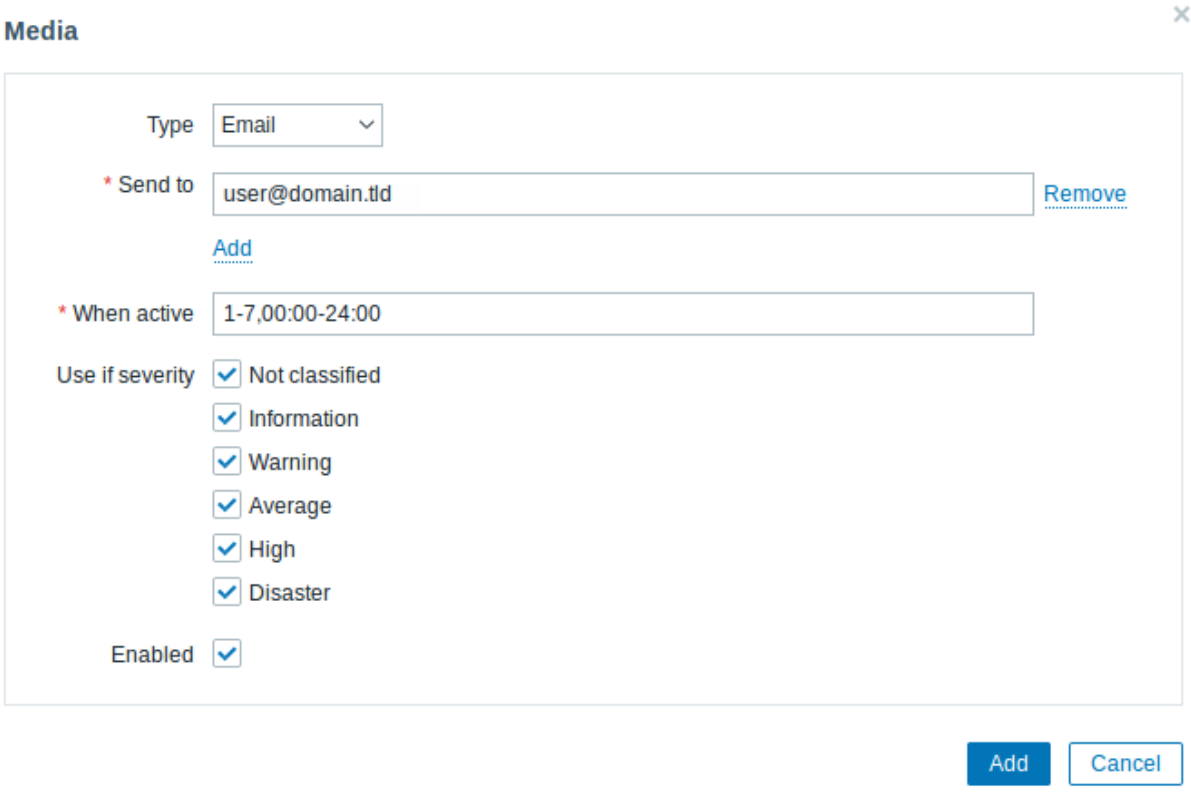
Кроме того, в разделе меню *Мониторинг → Узлы сети* можно нажать *Последние данные*, чтобы просмотреть все последние собранные метрики в виде списка. Обратите внимание, что для элемента данных *MySQL: Calculated value of innodb_log_file_size* ожидается отсутствие данных, поскольку значение будет вычисляться на основе данных за последний час.




Настройка оповещений о проблемах

Zabbix может уведомлять вас о проблемах в вашей инфраструктуре с помощью различных методов. Это руководство содержит базовые шаги настройки для отправки оповещений по электронной почте.

1. Перейдите в *Настройки пользователя → Профиль*, откройте вкладку *Media* и добавьте свой email.




2. Следуйте руководству по получению уведомления о проблеме.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

Проверьте вашу конфигурацию

Чтобы проверить вашу конфигурацию, мы можем смоделировать реальную проблему, изменив конфигурацию узла сети в веб-интерфейсе Zabbix.

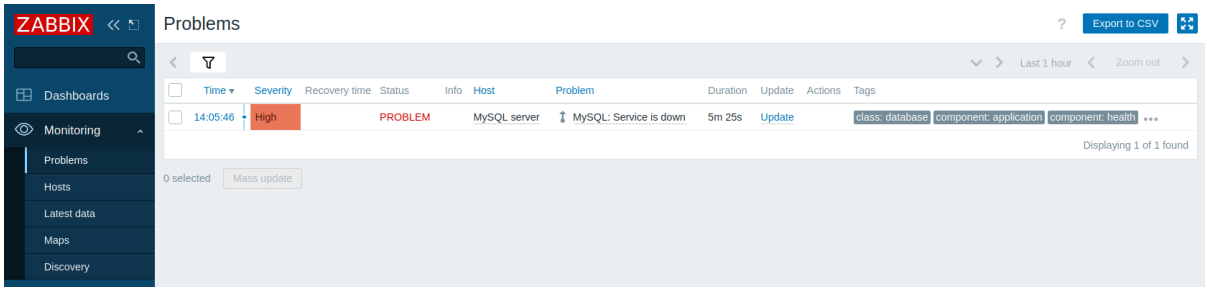
1. Откройте конфигурацию узла сети MySQL server в Zabbix.

2. Перейдите на вкладку *Macros* и выберите *Inherited and host macros*.

3. Нажмите *Change* рядом, например, со значением макроса {$MYSQL.USER}, настроенным ранее, и задайте другое имя пользователя MySQL.

4. Нажмите *Update*, чтобы обновить конфигурацию узла сети.

5. Через несколько мгновений Zabbix обнаружит проблему "MySQL: Service is down", поскольку не сможет подключиться к MySQL server. Проблема появится в *Monitoring → Problems*.




Если оповещения настроены, вы также получите уведомление о проблеме.

6. Измените значение макроса обратно на предыдущее, чтобы устранить проблему и продолжить мониторинг MySQL server.

Другие способы мониторинга MySQL

Вместо мониторинга сервера MySQL с помощью агента Zabbix 2 вы также можете использовать агент Zabbix или стандарт Open Database Connectivity (ODBC). Хотя рекомендуется использовать агент Zabbix 2, могут существовать некоторые варианты настройки, которые не поддерживают агент Zabbix 2 или требуют нестандартного подхода.

Ключевое различие между агентом Zabbix и ODBC заключается в методе сбора данных: агент Zabbix устанавливается непосредственно на сервер MySQL и собирает данные с помощью встроенных функций, тогда как ODBC использует драйвер ODBC для установления соединения с сервером MySQL и получения данных с помощью SQL-запросов.

Хотя многие шаги настройки похожи на мониторинг сервера MySQL с помощью агента Zabbix 2, есть и некоторые существенные различия — для мониторинга сервера MySQL необходимо настроить агент Zabbix или ODBC. Следующие инструкции проведут вас через эти **различия**.

Мониторинг MySQL с помощью агента Zabbix

Чтобы отслеживать сервер MySQL с помощью агента Zabbix, необходимо скачать и установить сервер Zabbix, веб-интерфейс Zabbix и агент Zabbix в соответствии с инструкциями для вашей ОС.

После успешной установки необходимых компонентов Zabbix нужно создать пользователя MySQL, как описано в разделе *Создание пользователя MySQL*.

После создания пользователя MySQL необходимо настроить агент Zabbix так, чтобы он мог устанавливать соединение с сервером MySQL и отслеживать его. Это включает настройку нескольких пользовательских параметров для выполнения пользовательских проверок агента, а также предоставление агенту Zabbix необходимых учетных данных для подключения к серверу MySQL от имени ранее созданного пользователя "zbx_monitor".

**Настройка агента Zabbix**

1. Перейдите в каталог дополнительных конфигураций агента Zabbix.

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

> **Attention:**
> Каталог дополнительных конфигураций агента Zabbix должен находиться в том же каталоге, что и файл конфигурации вашего агента Zabbix (*zabbix_agentd.conf*). В зависимости от вашей ОС и установки Zabbix этот каталог может располагаться в другом месте, чем указано в этом руководстве. Для значений по умолчанию проверьте параметр `Include` в файле конфигурации агента Zabbix.

Вместо того чтобы определять все необходимые пользовательские параметры для мониторинга сервера MySQL в файле конфигурации агента Zabbix, эти параметры будут определены в отдельном файле в каталоге дополнительных конфигураций.

2. Создайте файл *template_db_mysql.conf* в каталоге дополнительных конфигураций агента Zabbix.

```
vi template_db_mysql.conf
```

3. Скопируйте содержимое файла *template_db_mysql.conf* (расположенного в репозитории Zabbix) в созданный вами файл *template_db_mysql.conf* и сохраните его.

4. Перезапустите агент Zabbix, чтобы обновить его конфигурацию.

```
systemctl restart zabbix-agent
```

После того как вы настроите пользовательские параметры агента Zabbix, можно перейти к настройке учетных данных, которые позволят агенту Zabbix получить доступ к серверу MySQL.

5. Перейдите в домашний каталог агента Zabbix (если он не существует в вашей системе, его нужно создать; по умолчанию: */var/lib/zabbix*).

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

6. Создайте файл *.my.cnf* в домашнем каталоге агента Zabbix.

```
vi .my.cnf
```

7. Скопируйте следующий содержимое в файл *.my.cnf* (замените <*password*> на пароль пользователя ”zbx_monitor”).

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

**Настройте веб-интерфейс Zabbix и проверьте вашу конфигурацию**

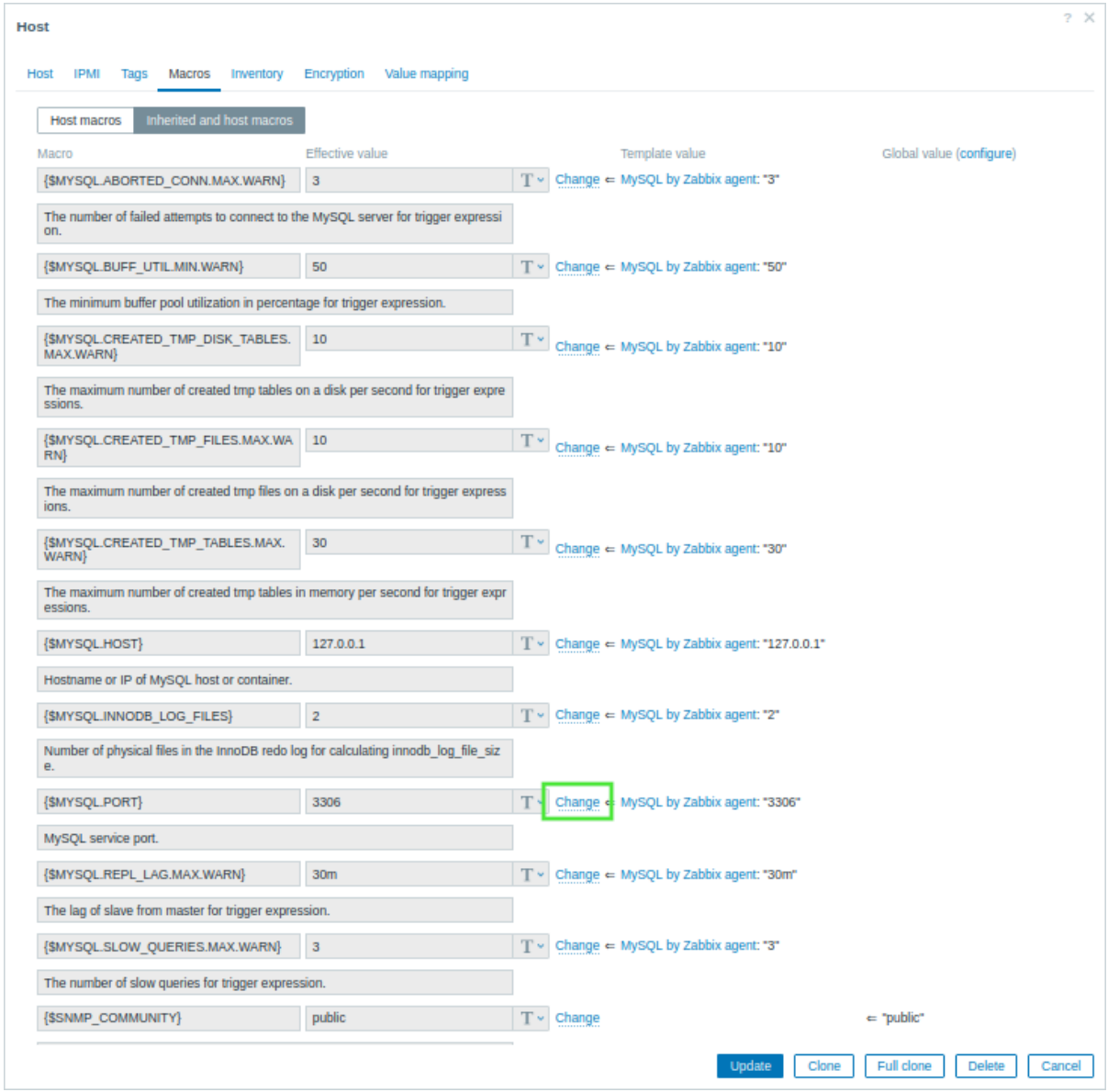
Чтобы настроить веб-интерфейс Zabbix, следуйте инструкциям в разделе *Настройка веб-интерфейса Zabbix* со следующими изменениями:

- В поле *Шаблоны* введите или выберите шаблон ”MySQL by Zabbix agent”, который будет привязан к узлу сети.
- Настройка *Макросов* не требуется.

После настройки веб-интерфейса Zabbix вы можете просмотреть собранные метрики и настроить оповещения о проблемах.

Чтобы проверить вашу конфигурацию, следуйте инструкциям в разделе *Проверка вашей конфигурации* со следующими изменениями:

- В разделе *Унаследованные макросы и макросы узла сети* в конфигурации узла сети MySQL server нажмите *Изменить* рядом со значением макроса {$MYSQL.PORT} и задайте другой порт (например, ”6033”).




Мониторинг MySQL с помощью ODBC

Чтобы отслеживать сервер MySQL с помощью ODBC, необходимо скачать и установить сервер Zabbix и веб-интерфейс Zabbix.

После успешной установки необходимых компонентов Zabbix нужно создать пользователя MySQL, как описано в разделе *Создание пользователя MySQL*.

После создания пользователя MySQL необходимо настроить ODBC. Это включает установку одной из наиболее часто используемых реализаций API ODBC с открытым исходным кодом - unixODBC - и драйвера unixODBC, а также редактирование файла конфигурации драйвера ODBC.

**Настройка ODBC**

1. Установите unixODBC. Рекомендуемый способ установки unixODBC — использовать стандартные репозитории пакетов операционной системы Linux.

```
apt install unixodbc
```

2. Установите драйвер базы данных MariaDB unixODBC. Хотя у вас база данных MySQL, для обеспечения совместимости используется драйвер MariaDB unixODBC.

```
apt install odbc-mariadb
```

3. Проверьте расположение файлов конфигурации ODBC *odbcinst.ini* и *odbc.ini*.

```
odbcinst -j
```

Результат выполнения этой команды должен быть похож на следующий.

```
unixODBC 2.3.9
```

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

4. Чтобы настроить драйвер ODBC для мониторинга базы данных MySQL, вам потребуется имя драйвера, которое находится в файле *odbcinst.ini*. В следующем примере файла *odbcinst.ini* имя драйвера — "MariaDB Unicode".

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

5. Скопируйте следующее содержимое в файл *odbc.ini* (замените <*password*> на пароль пользователя "zbx_monitor"). В этом руководстве в качестве адреса сервера MySQL используется "127.0.0.1" (localhost) для мониторинга сервера MySQL, установленного на той же машине, что и драйвер ODBC. Обратите внимание на имя источника данных (DSN) "test", которое потребуется при настройке веб-интерфейса Zabbix.

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

**Настройте веб-интерфейс Zabbix и проверьте вашу конфигурацию**

Чтобы настроить веб-интерфейс Zabbix, следуйте инструкциям в разделе *Настройка веб-интерфейса Zabbix* с учетом следующих изменений:

- В поле *Шаблоны* введите или выберите шаблон "MySQL by ODBC", который будет связан с узлом сети.
- Настраивать *Интерфейсы* не требуется.
- Значение макроса {$MYSQL.DSN} в разделе *Унаследованные и макросы узла сети* в конфигурации узла сети MySQL server должно быть установлено в имя DSN из файла *odbc.ini*.

После настройки веб-интерфейса Zabbix вы можете просмотреть собранные метрики, настроить оповещения о проблемах и проверить вашу конфигурацию.

См. также

- Создание элемента данных - как начать мониторинг дополнительных метрик.
- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, а затем, если проблема не будет решена в течение 45 минут, отправить сообщение менеджеру центра обработки данных).
- Мониторинг ODBC - как настроить ODBC в других дистрибутивах Linux и как начать мониторинг дополнительных метрик, связанных с базой данных, с помощью ODBC.
- Шаблон *MySQL by Zabbix agent* - дополнительная информация о шаблоне *MySQL by Zabbix agent*.
- Шаблон *MySQL by Zabbix agent 2* - дополнительная информация о шаблоне *MySQL by Zabbix agent 2*.
- Шаблон *MySQL by ODBC* - дополнительная информация о шаблоне *MySQL by ODBC*.

## 5 Мониторинг VMware с помощью Zabbix

Введение

На этой странице описаны шаги, необходимые для начала базового мониторинга VMware.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга VMware. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. раздел Мониторинг виртуальных машин или раздел Конфигурация руководства Zabbix.

**Предварительные требования**

Перед тем как продолжить работу с этим руководством, необходимо загрузить и установить сервер Zabbix и веб-интерфейс Zabbix в соответствии с инструкциями для вашей ОС.

В зависимости от вашей среды некоторые шаги в этом руководстве могут незначительно отличаться. Это руководство основано на среде, работающей под Ubuntu.

Предполагается, что VMware уже настроен. Это руководство не охватывает настройку VMware.

Настройка сервера Zabbix

Чтобы выполнять мониторинг VMware, необходимо включить процессы Zabbix *vmware collector*. Дополнительные сведения о том, как выполняется мониторинг VMware, см. в разделе Мониторинг виртуальных машин.

1. Откройте файл конфигурации сервера Zabbix.

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

2. Найдите и задайте параметр `StartVMwareCollectors` в файле конфигурации сервера Zabbix равным 2 или больше (значение по умолчанию — 0).

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

StartVMwareCollectors=2
```

3. Перезапустите сервер Zabbix.

```
systemctl restart zabbix-server
```

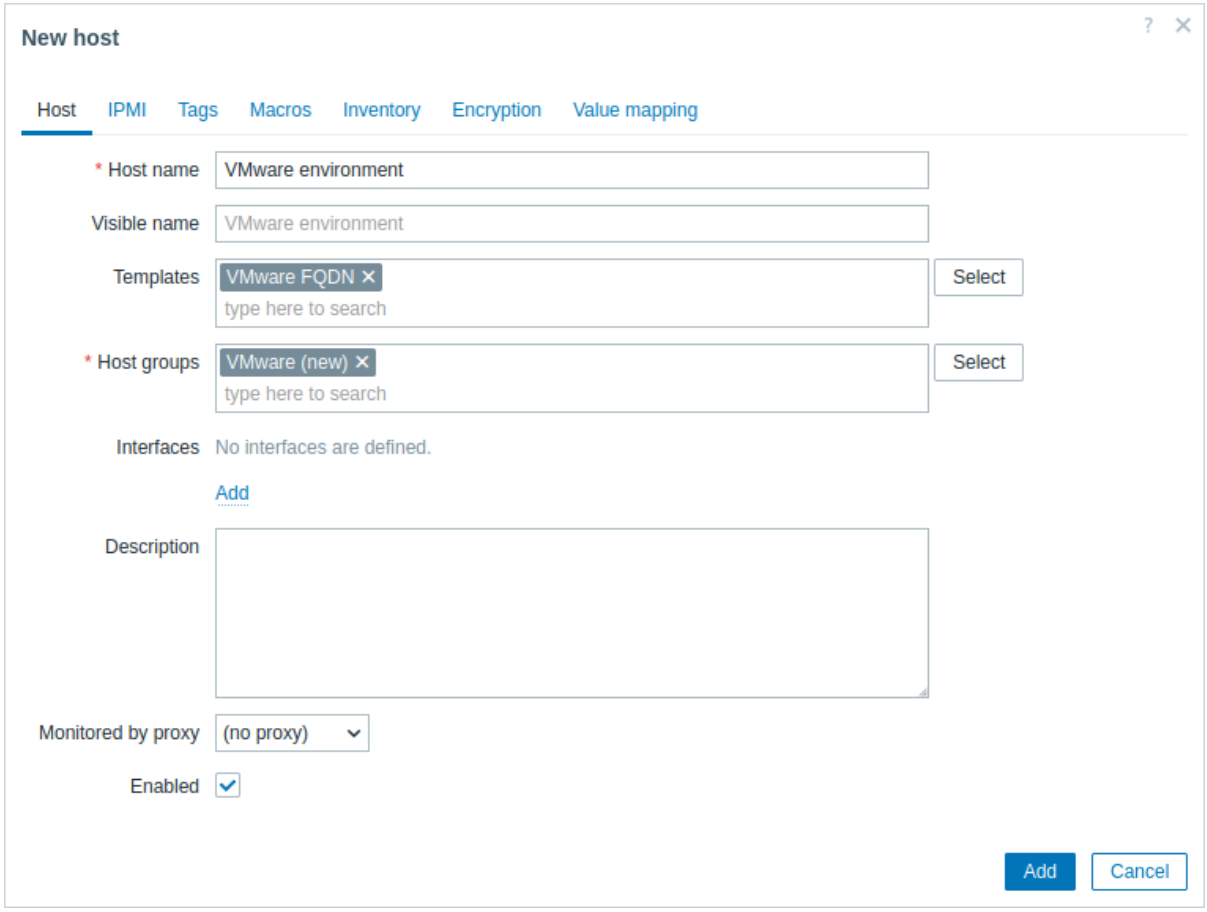
После запуска процессов *vmware collector* переходите к следующему шагу.

Настройка веб-интерфейса Zabbix

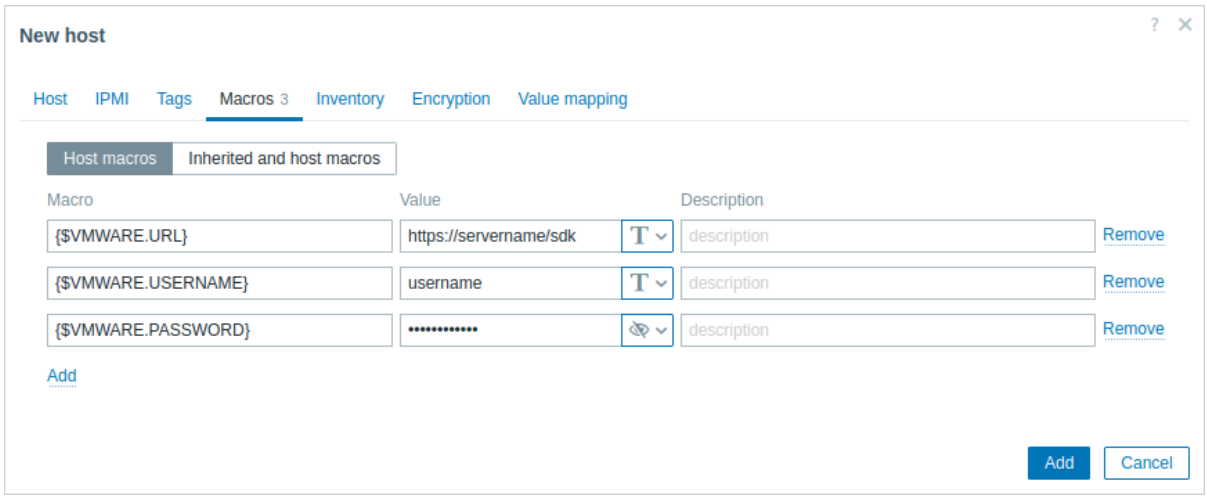
1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix:

- В поле *Host name* введите имя узла сети (например, ”VMware environment”).
- В поле *Templates* введите или выберите шаблон ”VMware FQDN” (или ”VMware”). Дополнительные сведения об этих шаблонах см. в разделе Мониторинг виртуальных машин.
- В поле *Host groups* введите или выберите группу узлов сети (например, новую группу узлов сети ”VMware”).




- На вкладке *Macros* задайте следующие макросы узла сети:
  - {$VMWARE.URL} - URL SDK службы VMware (vCenter или гипервизор ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} - имя пользователя службы VMware
  - {$VMWARE.PASSWORD} - пароль пользователя службы VMware {$VMWARE.USERNAME}



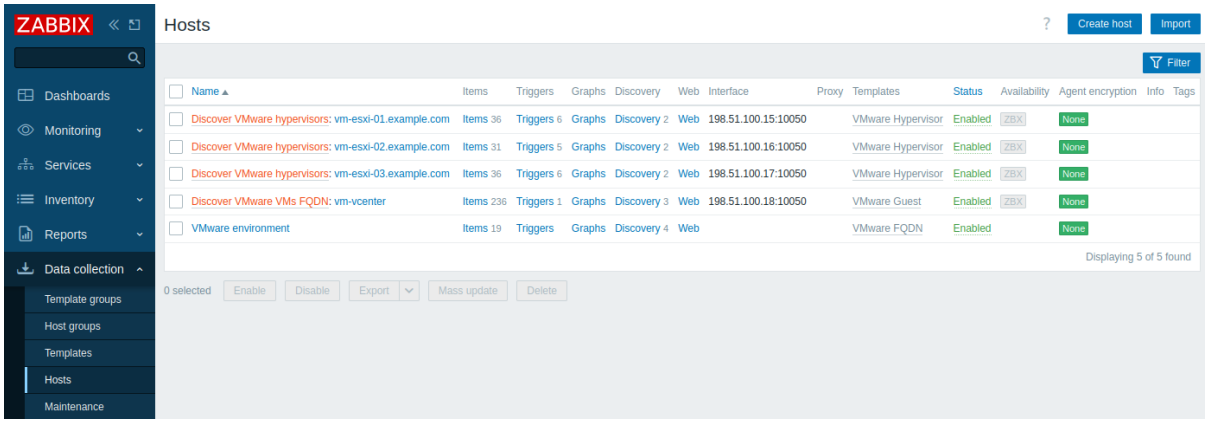

3. Нажмите кнопку *Add*, чтобы создать узел сети. Этот узел сети будет представлять вашу среду VMware.

Просмотр собранных метрик

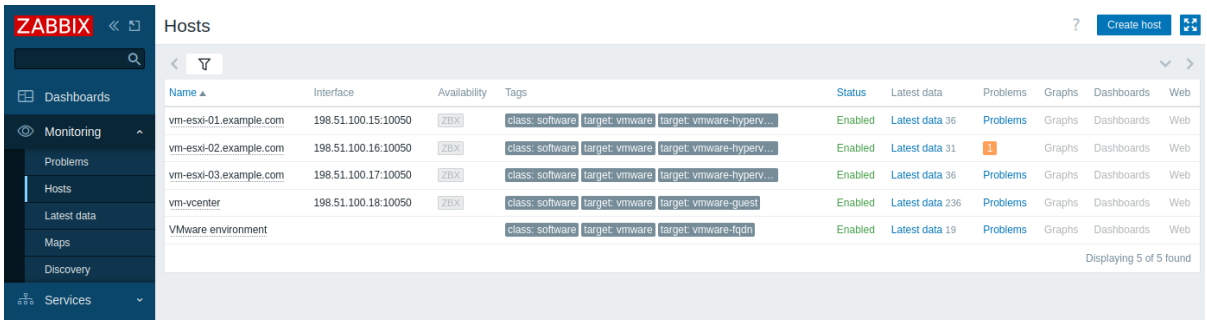
Поздравляем! На этом этапе Zabbix уже выполняет мониторинг вашей среды VMware.

В зависимости от конфигурации вашей среды VMware Zabbix может обнаружить и затем создать узлы сети для обнаруженных сущностей. Обратите внимание, что обнаружение и создание узлов сети при необходимости также можно выполнить вручную.

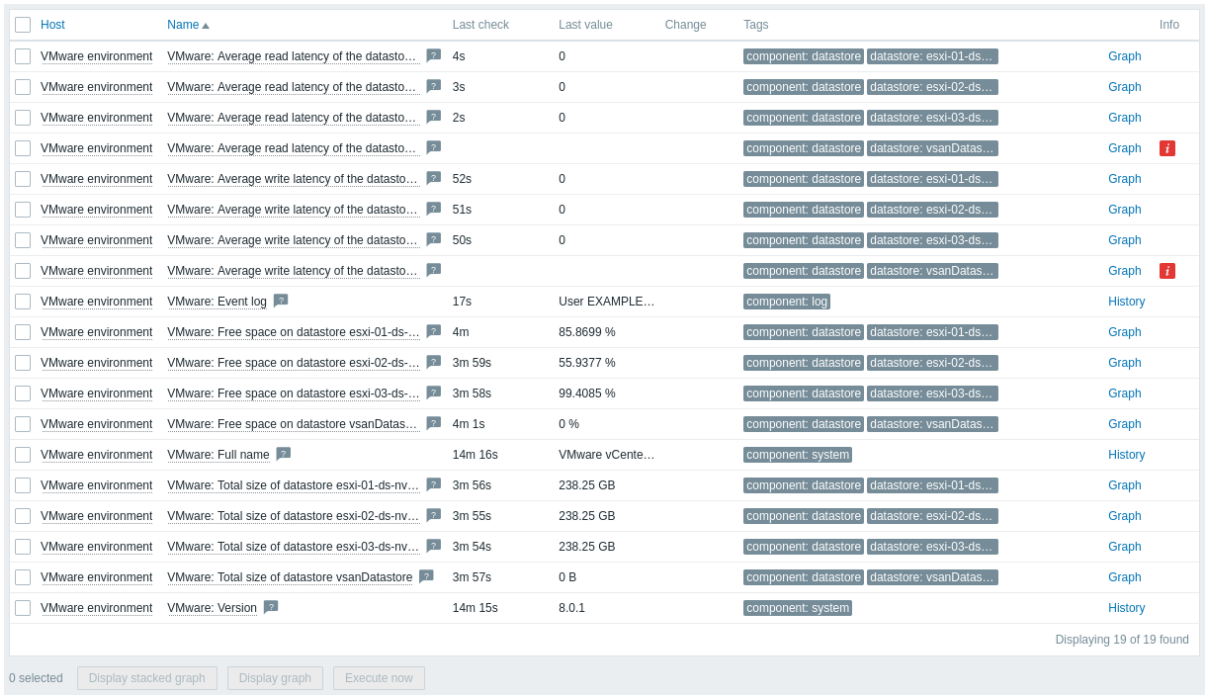
Чтобы просмотреть созданные узлы сети, перейдите в раздел меню *Сбор данных → Узлы сети*.




Чтобы просмотреть собранные метрики, перейдите в раздел меню *Мониторинг → Узлы сети* и нажмите *Последние данные* рядом с созданным узлом сети "VMware environment" или одним из узлов сети, созданных для обнаруженных сущностей.




Это действие откроет список всех последних метрик, собранных с выбранного узла сети.

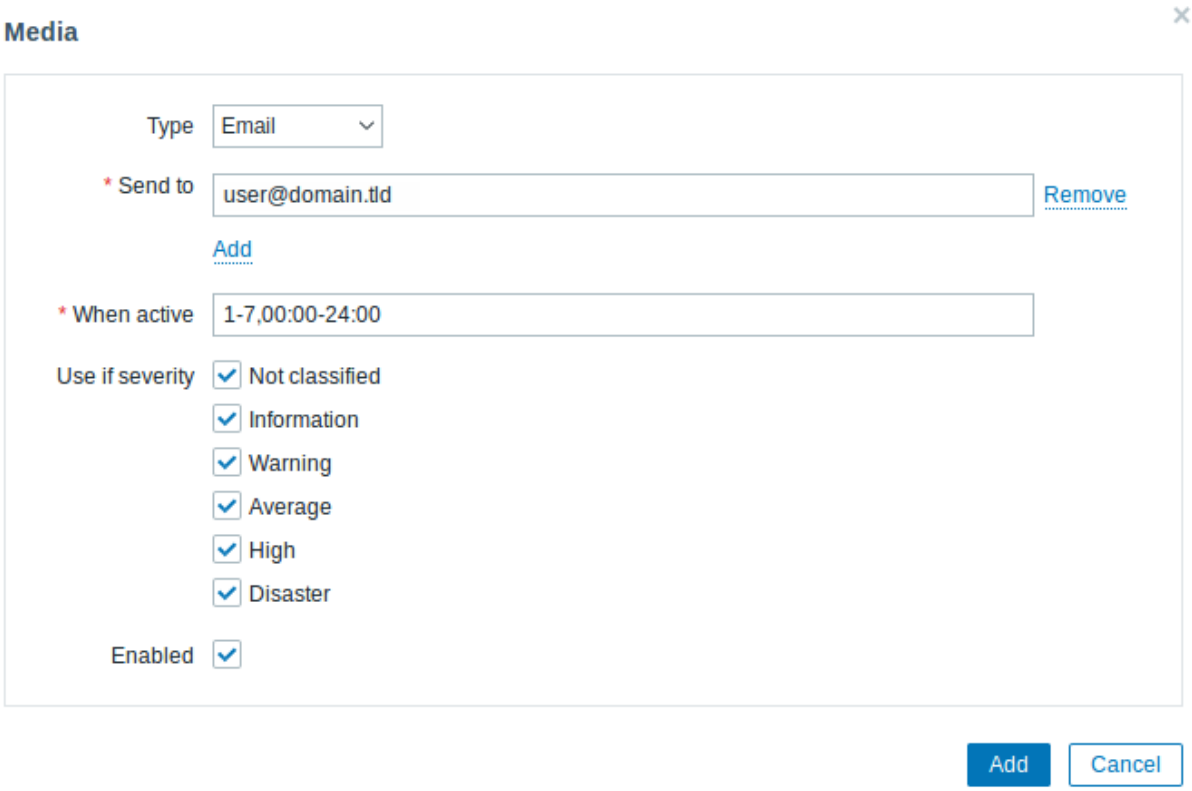



Обратите внимание, что для некоторых элементов данных отсутствуют данные и они находятся в состоянии *Не поддерживается*. Это связано с тем, что Zabbix не может найти допустимые счетчики производительности для конкретного хранилища данных, поскольку оно не включено в отслеживаемой среде VMware.

Настройка оповещений о проблемах

Zabbix может уведомлять вас о проблемах в вашей инфраструктуре с помощью различных методов. В этом руководстве приведены основные шаги настройки для отправки оповещений по электронной почте.

1. Перейдите в *Настройки пользователя → Профиль*, откройте вкладку *Media* и добавьте свой адрес электронной почты.




2. Следуйте руководству по получению уведомления о проблеме.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

См. также

- Создание элемента данных - как начать мониторинг дополнительных метрик.
- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, затем, если проблема не будет решена в течение 45 минут, отправить сообщение менеджеру центра обработки данных).
- Мониторинг виртуальных машин - дополнительная информация о мониторинге VMware (процесс сбора данных, параметры конфигурации сервера, рекомендации по устранению неполадок и т. д.).
- Ключи элементов данных для мониторинга VMware - полный список метрик VMware, которые можно отслеживать с помощью Zabbix.
- Шаблон *VMware* - дополнительная информация о шаблоне *VMware*.
- Шаблон *VMware FQDN* - дополнительная информация о шаблоне *VMware FQDN*.

## 6 Мониторинг сетевого трафика с помощью Zabbix

**Введение** На этой странице описаны шаги, необходимые для начала базового мониторинга сетевого трафика с помощью Zabbix.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга сетевого трафика. Если вы ищете возможности глубокой настройки или вам требуется более продвинутая конфигурация, см. раздел Configuration руководства Zabbix.

**Prerequisites**

Before proceeding with this guide, you need to download and install Zabbix server, Zabbix frontend and Zabbix agent according to the instructions for your OS. Note that you should install Zabbix agent on the machine that requires traffic monitoring. This can be either the same host where Zabbix server is installed or a different host.

This guide will provide the instructions on configuring the network traffic monitoring of *eth0* interface on a separate machine named *Remote host*.

**Настройка Zabbix для мониторинга** Агент Zabbix может собирать метрики в активном или пассивном режиме (одновременно). Дополнительную информацию см. в разделе Пассивные и активные проверки агента. В этом руководстве будет описан мониторинг с помощью пассивных проверок.

Настройка агента Zabbix

1. Откройте файл конфигурации агента на машине, где установлен агент (по умолчанию путь: */usr/local/etc/zabbix_agentd.conf*):

```
sudo vi /usr/local/etc/zabbix_agentd.conf
```

2. Добавьте IP-адрес или DNS-имя вашего сервера Zabbix в параметр *Server*. Например:

```
Server=192.0.2.22
```

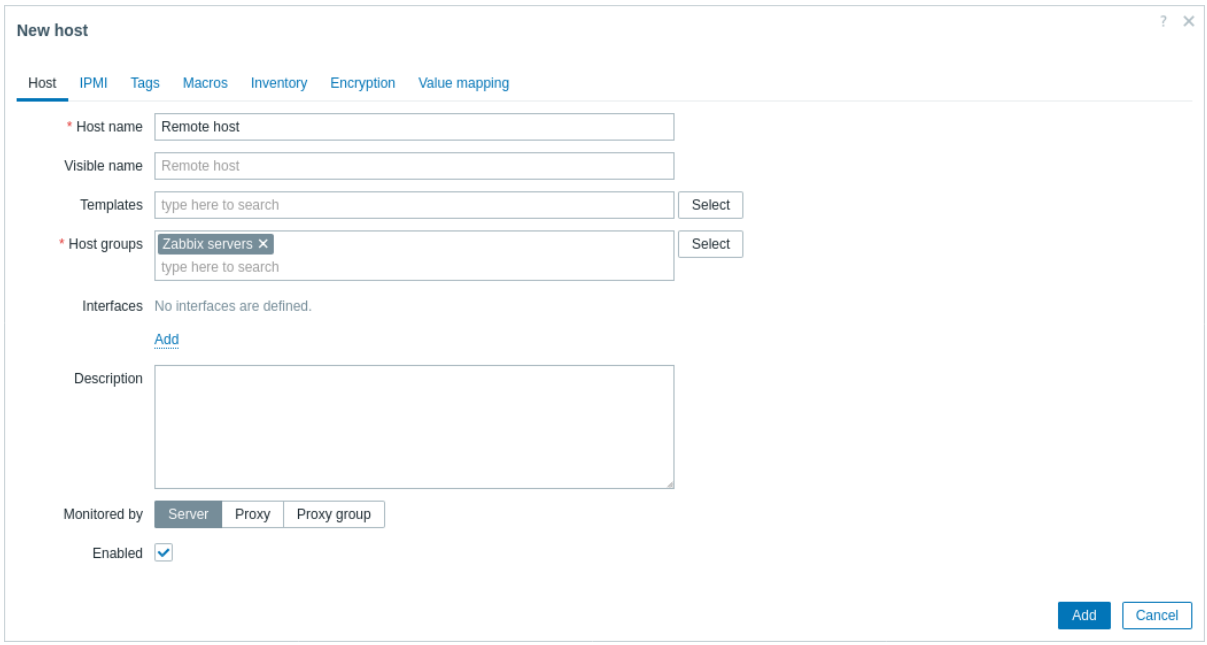
3. Перезапустите агент Zabbix:

```
systemctl restart zabbix-agent
```

Веб-интерфейс Zabbix

1. Войдите в веб-интерфейс Zabbix.

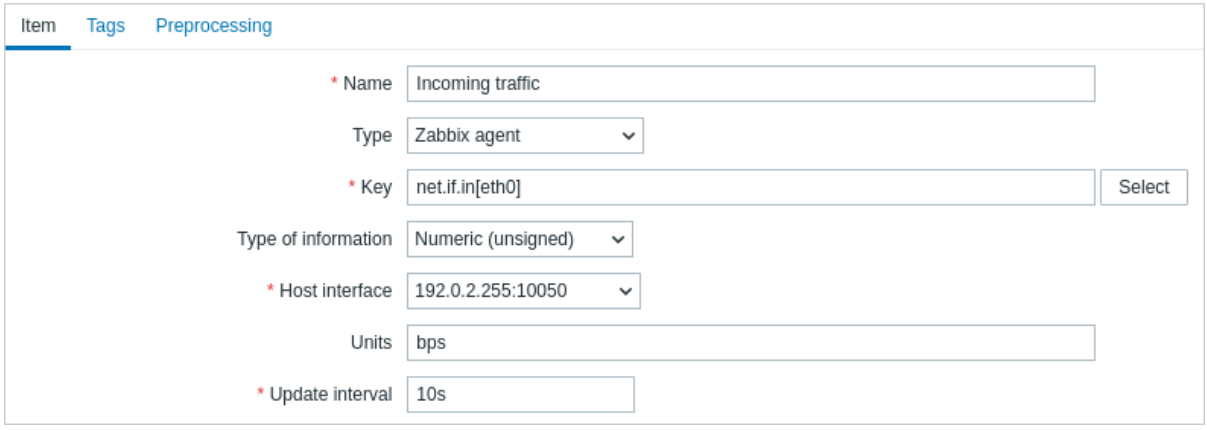
2. Создайте узел сети в веб-интерфейсе Zabbix, указав IP-адрес или DNS-имя машины, на которой установлен агент.



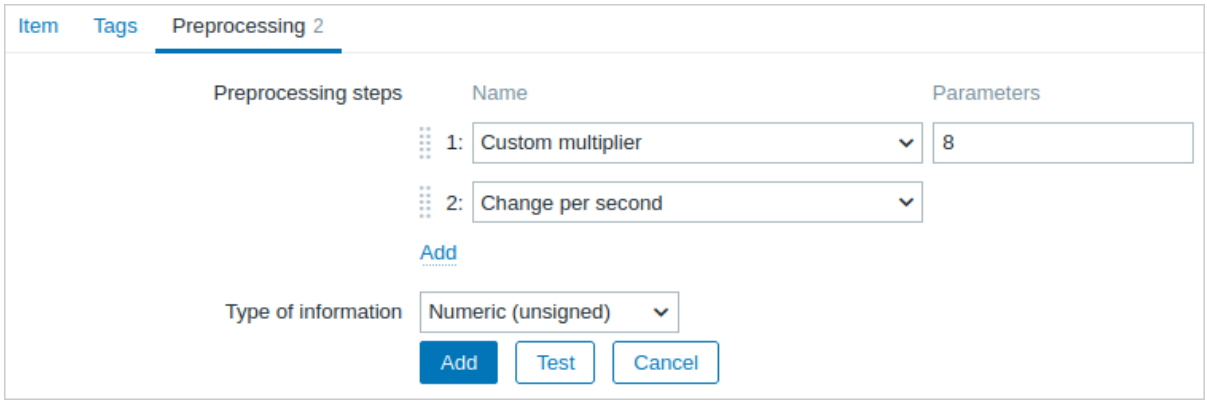

**Создание элементов данных** Следуйте инструкциям в разделе создание элемента данных, чтобы добавить элементы данных для мониторинга трафика, а именно:

- Входящий трафик
- Исходящий трафик
- Общий трафик

Простая настройка элемента данных для мониторинга входящего трафика будет выглядеть следующим образом:

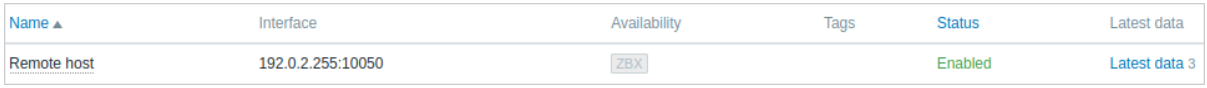


Чтобы сделать собранные данные пригодными для практического использования, при создании элементов данных можно задать некоторые шаги предобработки. В данном случае это может быть умножение на 8 (для преобразования байтов в биты) и представление в виде изменения за секунду.

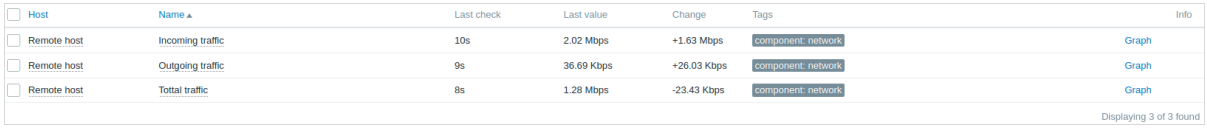



**Просмотр собранных данных** Поздравляем! На этом этапе Zabbix уже отслеживает сетевой трафик в вашей сети.

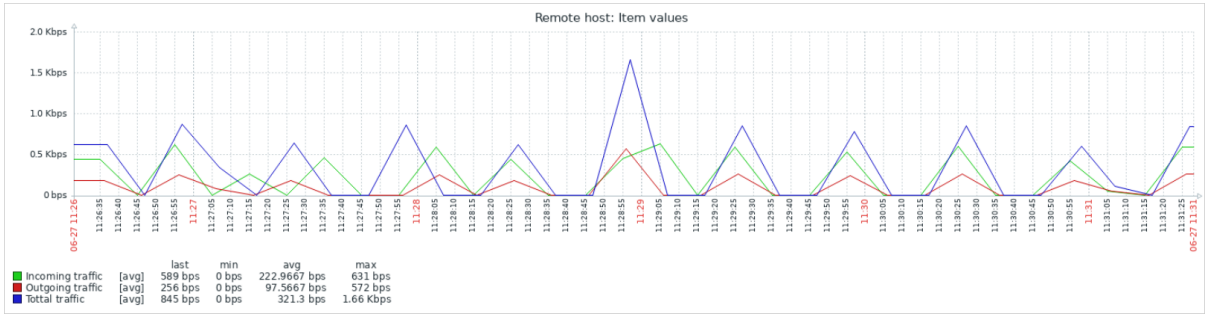
Чтобы просмотреть собранные метрики, откройте раздел меню *Monitoring → Hosts* и нажмите *Latest data* в строке узла сети.



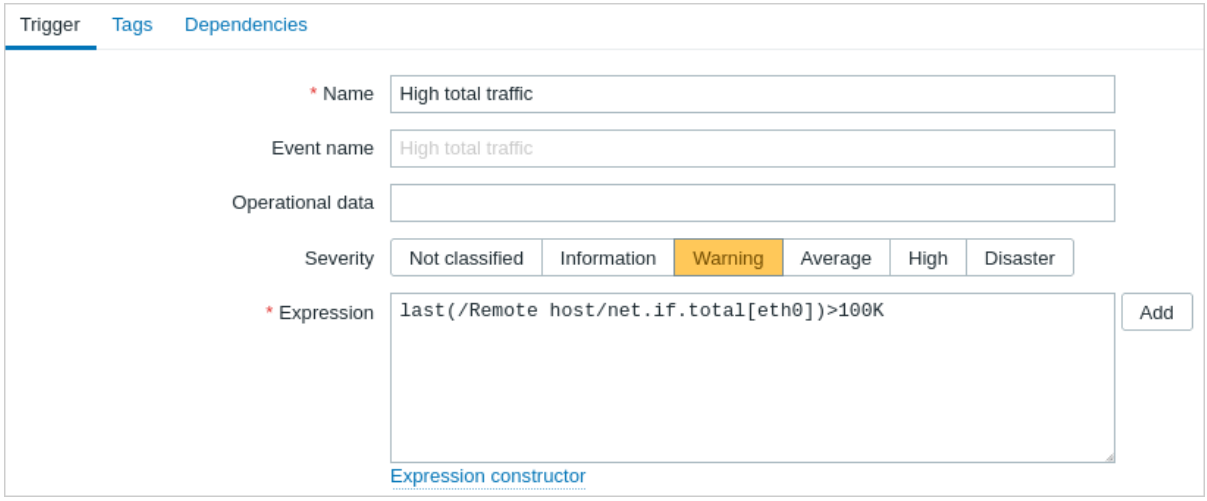

Вы увидите данные о трафике.



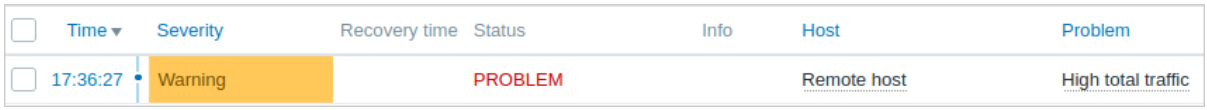

**Просмотр графиков** Собранные данные можно отображать в виде графиков. Чтобы просмотреть их, в разделе *Последние данные* нажмите *График* в строке элемента данных или выберите нужные элементы данных и нажмите *Показать график* ниже.




**Настройка триггеров** Вы можете задать триггеры, чтобы обнаруживать аномальный сетевой трафик. См. инструкции по настройке триггера и добавьте триггер, сигнализирующий о том, что общий трафик слишком высок, например:




Теперь сделайте так, чтобы трафик превысил порог, заданный в выражении триггера, и перейдите в *Monitoring → Problems*, чтобы проверить, что проблема там отображается.




**Настройка оповещений о проблемах** Существует несколько способов получать уведомления о проблеме. Электронная почта — самый популярный из них; следуйте инструкциям по настройке уведомления о проблеме по электронной почте. Вы также можете выбрать другие типы медиа, которые будут использоваться для доставки уведомлений.

**См. также:**

- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, а затем, если проблема не будет решена в течение 45 минут, отправить сообщение менеджеру центра обработки данных).
- Подтверждение проблемы - как указать, что проблема известна, добавлять комментарии по ее устранению, подавлять или закрывать проблему.
- Мониторинг Linux с помощью агента Zabbix - как начать базовый мониторинг наиболее важных элементов данных, связав предварительно настроенный шаблон.

### 7 Мониторинг сетевого трафика с использованием активных проверок

**Введение** На этой странице вы пройдете через шаги, необходимые для начала базового мониторинга сетевого трафика с помощью Zabbix с использованием активных проверок.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга сетевого трафика с использованием активных проверок. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. раздел Configuration руководства Zabbix.

**Prerequisites**

Before proceeding with this guide, you need to download and install Zabbix server, Zabbix frontend and Zabbix agent according to the instructions for your OS. Note that you should install Zabbix agent on the machine that requires traffic monitoring. This can be either the same host where Zabbix server is installed or a different host.

This guide will provide the instructions on configuring the network traffic monitoring of *eth0* interface on a separate machine named *Remote host*.

**Настройка Zabbix для мониторинга** Агент Zabbix может собирать метрики в активном или пассивном режиме (одновременно). Дополнительную информацию см. в разделе Пассивные и активные проверки агента. В этом руководстве будет описан мониторинг с помощью **активных проверок**.

Настройка Zabbix agent

1. Откройте файл конфигурации агента на машине, где установлен агент.

Если вы используете Zabbix agent:

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

Если вы используете Zabbix agent 2:

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

2. Добавьте IP-адрес или DNS-имя (и необязательный порт) вашего сервера Zabbix в параметр ServerActive. Например:

```
ServerActive=192.0.2.1:10051
```

Агент Zabbix будет использовать этот адрес для подключения к порту trapper сервера Zabbix (по умолчанию: 10051) и запроса данных конфигурации для активных проверок.

3. Укажите параметр Hostname, который должен совпадать с именем узла сети, которое будет задано в веб-интерфейсе Zabbix. В нашем примере это:

```
Hostname=Remote host
```

Значение Hostname должно совпадать, потому что для активных проверок агент Zabbix использует его для получения правильной конфигурации узла сети с сервера. В частности, агент инициирует подключение к серверу и идентифицирует себя с помощью значения Hostname. Затем сервер предоставляет конфигурацию мониторинга для этого узла сети. Если эти значения различаются, агент не получит соответствующую конфигурацию, что приведет к отсутствию метрик или проблемам с мониторингом.

4. Перезапустите агент Zabbix.

Если вы используете Zabbix agent:

```
systemctl restart zabbix-agent
```

Если вы используете Zabbix agent 2:

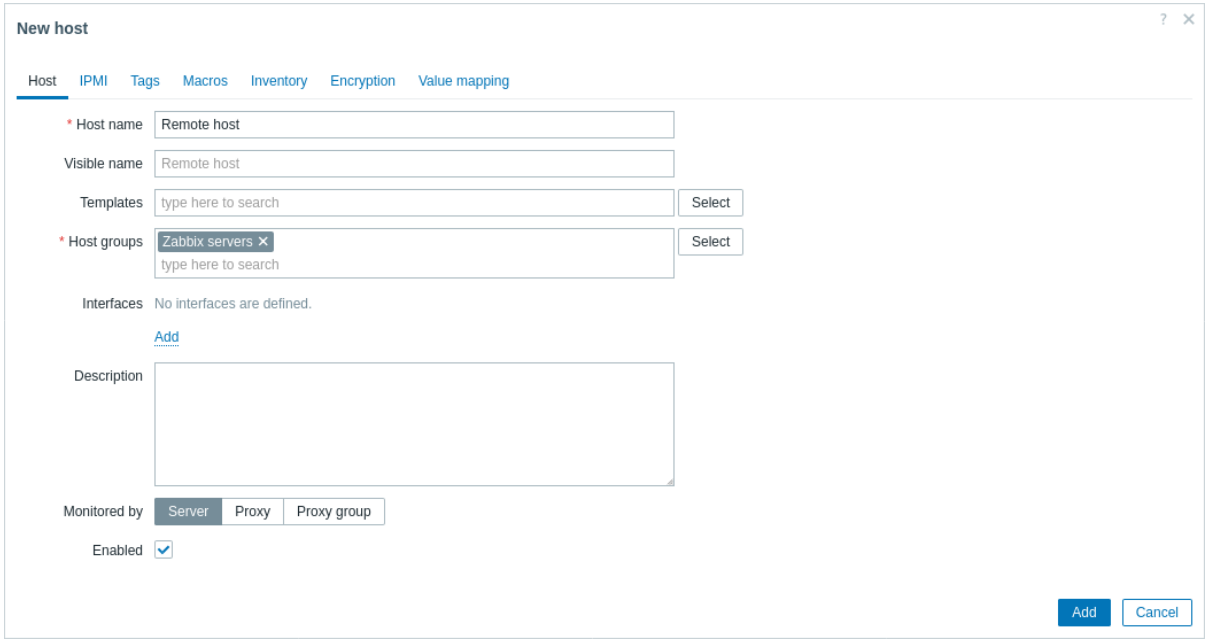
```
systemctl restart zabbix-agent2
```

Веб-интерфейс Zabbix

1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix.

- В поле *Host name* введите имя узла сети (например, ”Remote host”), которое соответствует значению параметра Hostname, ранее заданному в файле конфигурации агента.
- В поле *Host groups* введите или выберите группу узлов сети (например, ”Zabbix servers”).
- Оставьте *Interfaces* не заданным; интерфейс не требуется для активных проверок, поскольку агент инициирует соединение с сервером вместо ожидания подключения от сервера.

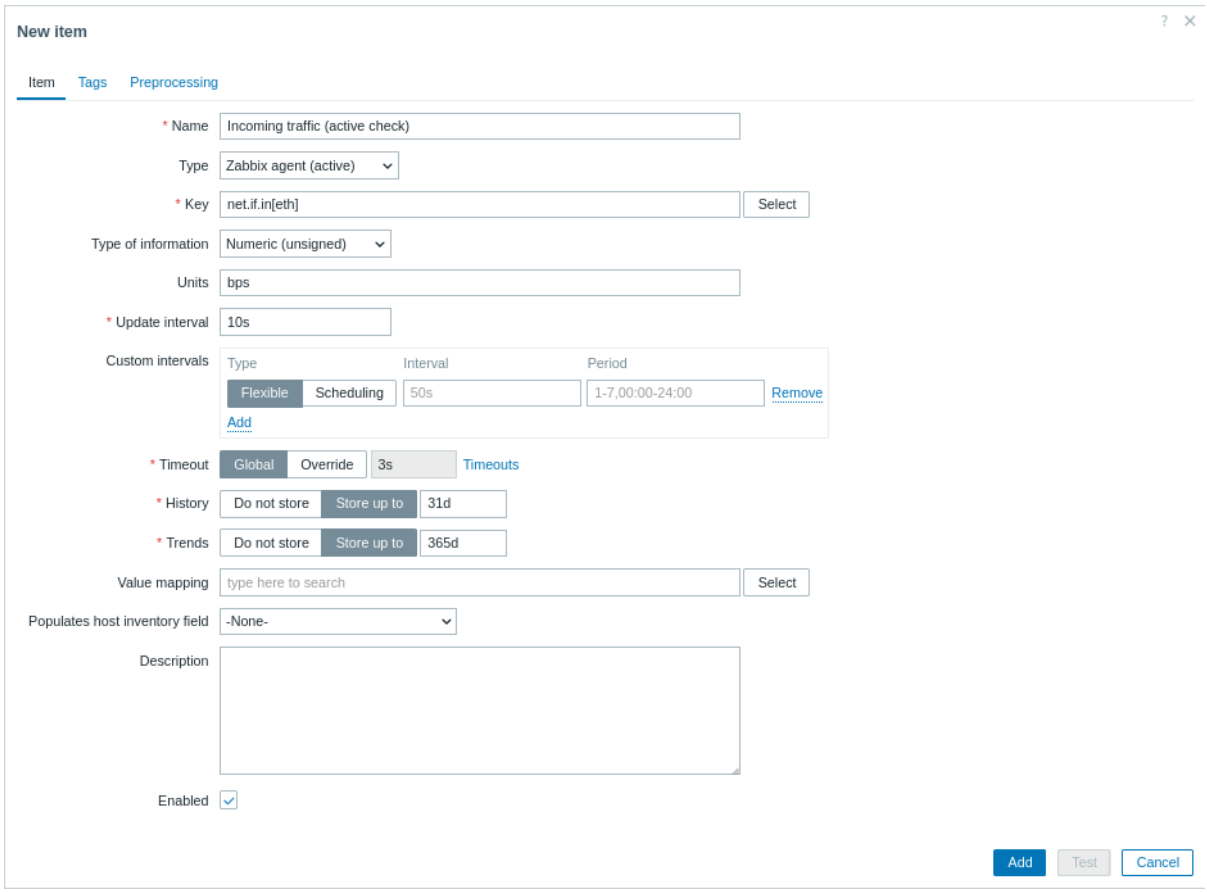



3. Нажмите *Add*, чтобы добавить узел сети. Этот узел сети будет представлять контролируемую машину Linux.

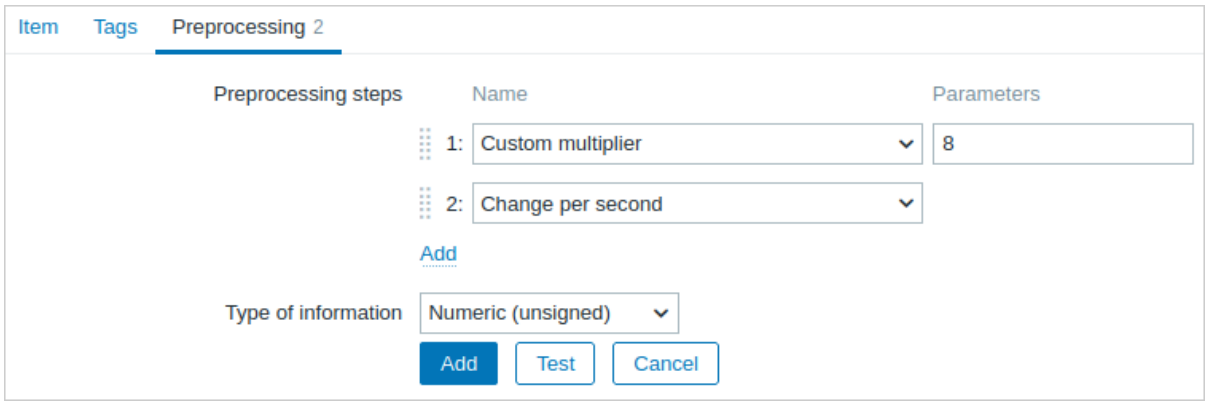
**Создание элементов данных** Следуйте инструкциям по созданию элемента данных, чтобы добавить элементы данных для мониторинга трафика, а именно:

- Входящий трафик
- Исходящий трафик
- Общий трафик

Простая настройка элемента данных для мониторинга входящего трафика с активной проверкой будет выглядеть следующим образом:

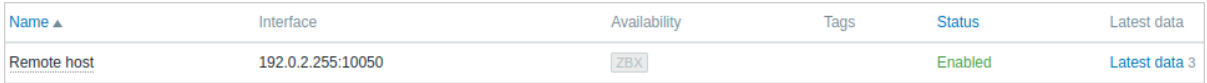



Чтобы собранные данные были пригодны для практического использования, при создании элементов данных можно задать некоторые шаги предобработки. В данном случае это может быть умножение на 8 (для преобразования байтов в биты) и представление в виде изменения за секунду.

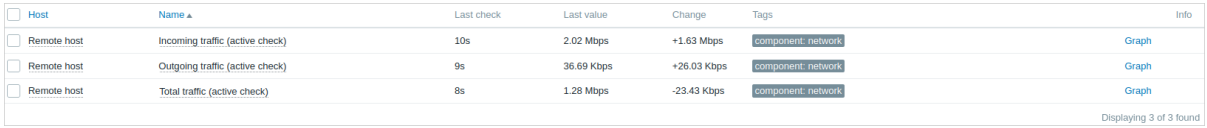



**Просмотр собранных данных** Поздравляем! На этом этапе Zabbix уже отслеживает сетевой трафик в вашей сети.

Чтобы просмотреть собранные метрики, откройте раздел меню *Monitoring → Hosts* и нажмите *Latest data* в строке узла сети.




Вы увидите данные о трафике.



| Host | Name ▲ | Last check | Last value | Change | Tags | Info |
|---|---|---|---|---|---|---|
| Remote host | Incoming traffic (active check) | 10s | 2.02 Mbps | +1.63 Mbps | component: network | Graph |
| Remote host | Outgoing traffic (active check) | 9s | 36.69 Kbps | +26.03 Kbps | component: network | Graph |
| Remote host | Total traffic (active check) | 8s | 1.28 Mbps | -23.43 Kbps | component: network | Graph |

Displaying 3 of 3 found

**См. также:**

- Просмотр графиков - как отображать собранные данные в виде графиков.
- Настройка триггеров - как настроить триггеры для обнаружения аномального сетевого трафика.
- Настройка оповещений о проблемах - как настроить уведомления о проблемных ситуациях.
- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, затем, если проблема не решена в течение 45 минут, отправить сообщение менеджеру дата-центра).
- Подтверждение проблемы - как указать, что проблема известна, добавлять комментарии по ее устранению, подавлять или закрывать проблему.
- Мониторинг Linux с помощью агента Zabbix - как начать базовый мониторинг наиболее важных элементов данных, связав предварительно настроенный шаблон.
- Установка из пакетов - как устанавливать компоненты Zabbix с использованием официальных пакетов RPM и DEB для различных дистрибутивов Linux, обеспечивая доступ к последним функциям и исправлениям ошибок.

## 8 Мониторинг веб-сайтов с помощью элементов данных Browser

Введение

На этой странице описаны шаги, необходимые для начала базового мониторинга веб-сайтов с помощью элементов данных Browser.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга веб-сайтов с помощью элементов данных Browser. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. страницу Browser items или раздел Configuration руководства Zabbix.

**Prerequisites**

Before proceeding with this guide, you need to download and install Zabbix server and Zabbix frontend according to the instructions for your OS.

Depending on your environment, some steps in this guide may slightly differ. This guide is based on an environment running Ubuntu.

Configure WebDriver

Browser items require an automation framework (either Selenium Server or a plain WebDriver, for example, ChromeDriver) as a web testing endpoint that controls and interacts with a browser, executing test commands such as clicking buttons or entering text. As an example, this guide will use Selenium Server with Chrome in a Docker container.

It is assumed that Docker is already configured. This guide does not cover the configuration of Docker. For installation instructions, see Install Docker Engine on Ubuntu.

1. Launch Selenium Server with Chrome in a Docker container with the following options:

- **docker run --name browser** - runs a new Docker container named "browser";
- **-p 4444:4444** - maps port 4444 on your host machine to port 4444 on the container (this is the port used by Selenium Server to accept commands);
- **-p 7900:7900** - maps port 7900 on your host machine to port 7900 on the container (this is the port used by the Virtual Network Computing (VNC) server, allowing you to view the browser GUI remotely; requires a VNC client);
- **--shm-size="2g"** - allocates 2GB of shared memory to the container (this is important for Chrome to run properly, as it can require a significant amount of shared memory to avoid crashes);
- **-d** - runs the container in detached mode, meaning it will run in the background;
- **selenium/standalone-chrome:latest** - specifies the Docker image to use; in this case, the latest version of Selenium Server with Chrome.

```
docker run --name browser \
-p 4444:4444 \
-p 7900:7900 \
```

```
--shm-size="2g" \
-d selenium/standalone-chrome:latest
```

2. Ensure that the `browser` Docker container is running and accessible.

- Retrieve the IP address of the container (in this example, 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
###     ...
```

- Test the connection to the container with Ncat:

```
nc -zv 192.0.2.1 4444

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

- Retrieve the webpage content from Selenium Server with 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:**
For troubleshooting, refer to Docker documentation.

Настройка сервера Zabbix

Элементы данных браузера выполняются и обрабатываются процессами Zabbix *browser poller*, которые необходимо включить, изменив параметр конфигурации сервера StartBrowserPollers. Кроме того, параметр WebDriverURL должен указывать на ранее настроенную конечную точку веб-тестирования.

По умолчанию параметр StartBrowserPollers установлен в 1, поэтому вам нужно указать только конечную точку веб-тестирования.

1. Откройте файл конфигурации сервера Zabbix.

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

2. Найдите и задайте параметр WebDriverURL в файле конфигурации сервера Zabbix:

```
##### Option: WebDriverURL
###WebDriver interface HTTP[S] URL. For example http://localhost:4444 used with Selenium WebDriver standal
###
### Mandatory: no
### Default:
### WebDriverURL=

WebDriverURL=192.0.2.1:4444
```

3. Перезапустите сервер Zabbix.

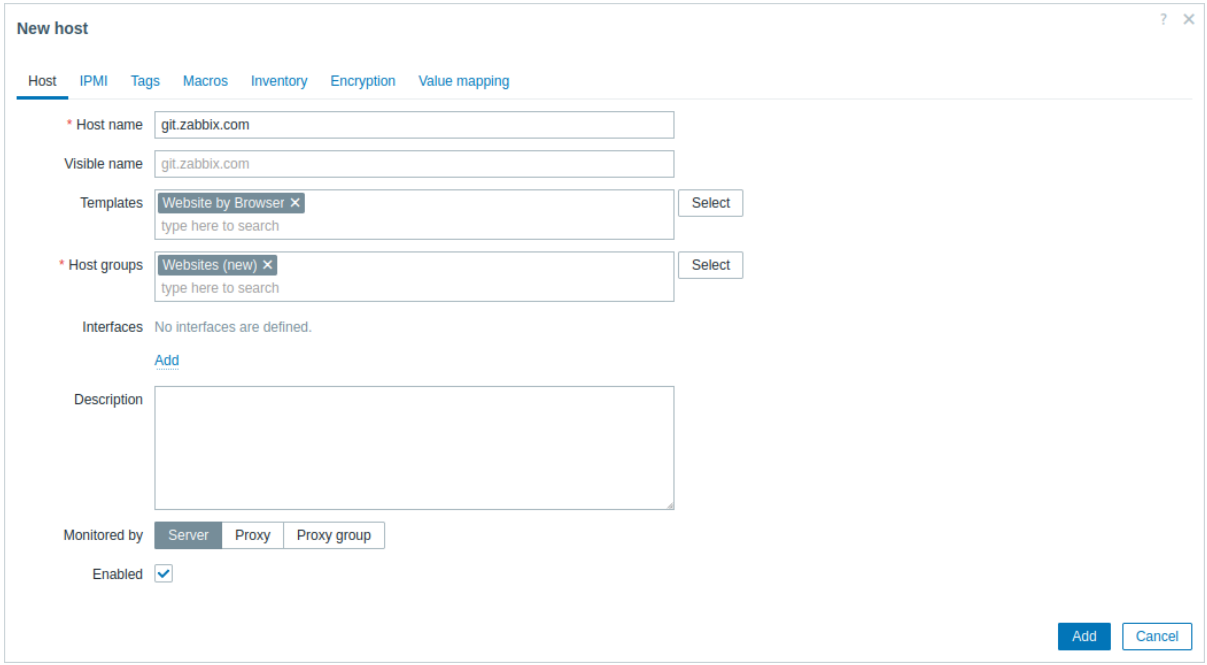
```
systemctl restart zabbix-server
```

Настройка веб-интерфейса Zabbix

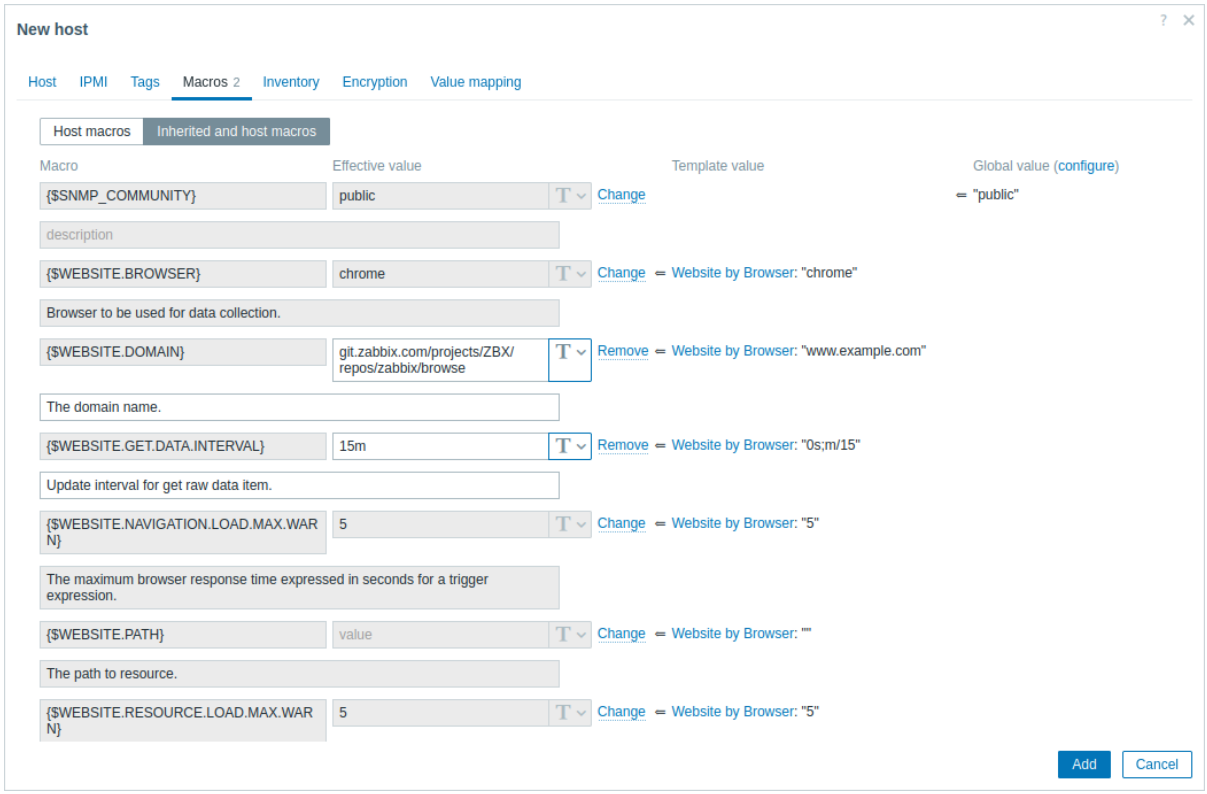
1. Войдите в веб-интерфейс Zabbix.

2. Создайте узел сети в веб-интерфейсе Zabbix:

- В поле *Host name* введите имя узла сети (например, ”git.zabbix.com”).
- В поле *Templates* введите или выберите шаблон ”Website by Browser”. Дополнительные сведения об этом шаблоне см. в разделе Website by Browser.
- В поле *Host groups* введите или выберите группу узлов сети (например, новую группу узлов сети ”Websites”).




- На вкладке *Macros* переключитесь на *Inherited and host macros*, найдите следующие макросы и нажмите *Change* рядом со значением макроса, чтобы обновить его:
  - {$WEBSITE.DOMAIN} - Domain name (например, git.zabbix.com/projects/ZBX/repos/zabbix/browse)
  - {$WEBSITE.GET.DATA.INTERVAL} - Item data update interval (например, 15m)

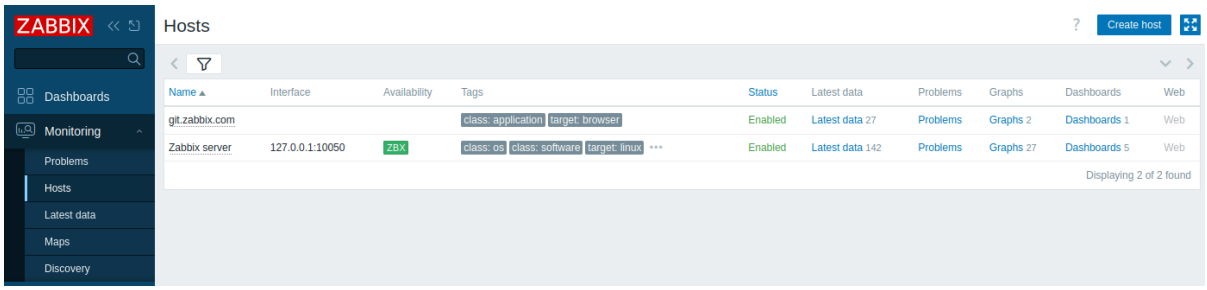


3. Нажмите кнопку *Add*, чтобы создать узел сети. Этот узел сети будет представлять веб-сайт, который вы хотите мониторить.

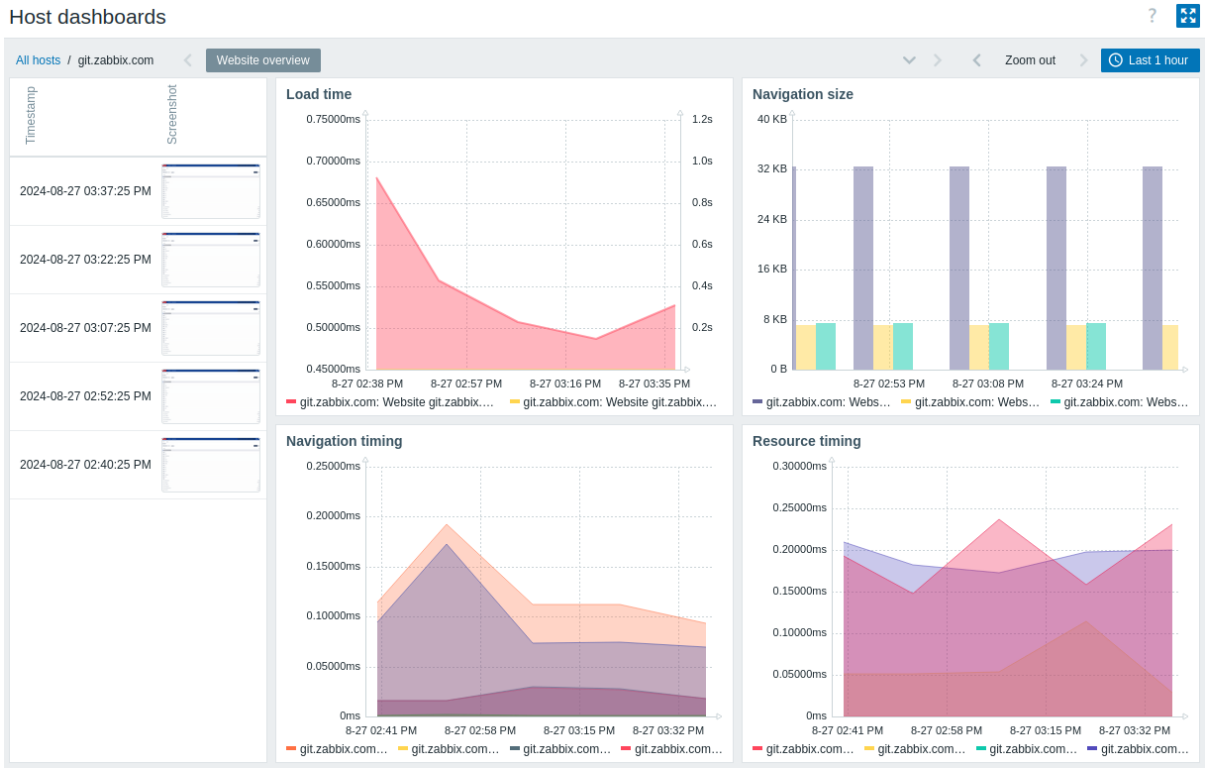
Просмотр собранных метрик

Поздравляем! К этому моменту Zabbix уже отслеживает указанный вами веб-сайт.

Чтобы просмотреть собранные метрики, перейдите в раздел меню *Monitoring → Hosts* и нажмите *Dashboards* рядом с узлом сети.



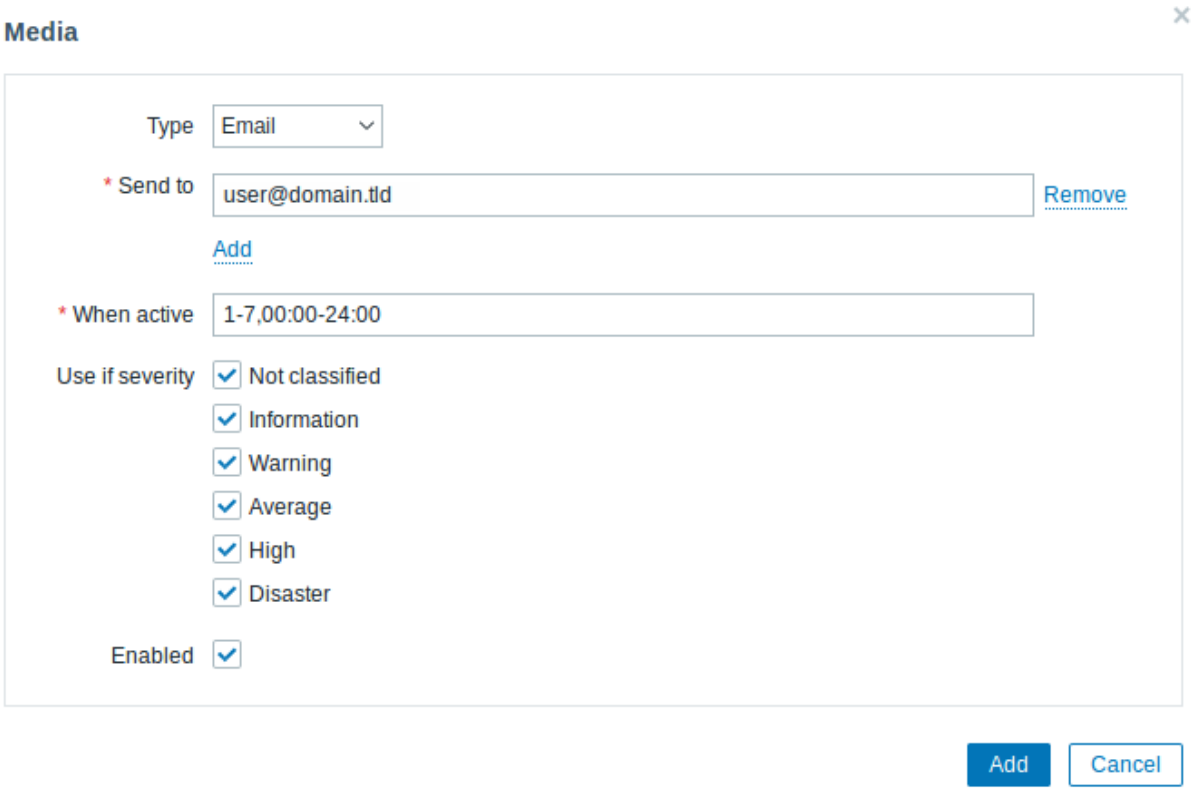

Это действие откроет панель узла сети (настроенную на уровне шаблона) с наиболее важными метриками, собранными с веб-сайта.




Настройка оповещений о проблемах

Zabbix может уведомлять вас о проблемах в вашей инфраструктуре с помощью различных методов. Это руководство содержит базовые шаги настройки для отправки оповещений по электронной почте.

1. Перейдите в *Настройки пользователя → Профиль*, откройте вкладку *Media* и добавьте свой адрес электронной почты.




2. Следуйте руководству по получению уведомления о проблеме.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

Проверьте вашу конфигурацию

Чтобы проверить вашу конфигурацию, мы можем смоделировать реальную проблему, изменив конфигурацию узла сети в веб-интерфейсе Zabbix.

1. Откройте конфигурацию узла сети вашего веб-сайта в Zabbix.

2. Перейдите на вкладку *Macros* и выберите *Inherited and host macros*.

3. Нажмите *Change* рядом, например, со значением макроса {$WEBSITE.DOMAIN}, настроенного ранее, и задайте неверное имя домена (например, /git.zabbix.com/projects/ZBX/repos/zabbix/browse).

4. Нажмите *Update*, чтобы обновить конфигурацию узла сети.

5. Через несколько мгновений Zabbix обнаружит проблему "Failed to get JSON of the requested website", потому что не сможет подключиться к указанному веб-сайту. Проблема появится в *Monitoring → Problems*.

Если настроены оповещения, вы также получите уведомление о проблеме.

6. Измените значение макроса обратно на прежнее, чтобы устранить проблему и продолжить мониторинг веб-сайта.

См. также

- Создание элемента данных - как начать мониторинг дополнительных метрик.
- Эскалации проблем - как создавать многошаговые сценарии оповещений (например, сначала отправить сообщение системному администратору, затем, если проблема не будет решена в течение 45 минут, отправить сообщение менеджеру дата-центра).
- Элементы данных Browser - как настроить элементы данных Browser.
- Шаблон Website by Browser - дополнительная информация о шаблоне *Website by Browser*.

## 9 Мониторинг сертификатов веб-сайтов с помощью Zabbix agent 2 (пассивный)

Введение

Это руководство предоставляет исчерпывающий обзор того, как настроить и отслеживать SSL/TLS-сертификаты с использованием ключа `web.certificate.get` в агенте Zabbix 2. Оно предназначено для упрощения мониторинга сертификатов для одного или нескольких веб-сайтов, позволяя администраторам быстро выявлять потенциальные проблемы, такие как просроченные или недействительные сертификаты.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и содержит минимальный набор шагов, необходимых для включения базового мониторинга сертификатов веб-сайтов. Если вам нужны возможности глубокой настройки или более расширенная конфигурация, см. раздел Configuration руководства Zabbix.

**Предварительные требования**

Перед тем как продолжить работу с этим руководством, вам необходимо загрузить и установить сервер Zabbix, веб-интерфейс Zabbix и агент Zabbix 2 в соответствии с инструкциями для вашей ОС. В этом руководстве предполагается, что сервер Zabbix и агент установлены на одном и том же компьютере; поэтому в конфигурации используется `127.0.0.1`.

Configure Zabbix agent 2

1. Open the Zabbix agent configuration file (default path: `/etc/zabbix/zabbix_agent2.conf`):

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

2. Set the Server parameter to 127.0.0.1, since the agent and server are running on the same machine:

```
Server=127.0.0.1
```

3. Save the file and restart the Zabbix agent 2 service:

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

4. After setup and configuring Zabbix agent 2, test its availability with:

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

Zabbix agent 2 includes the WebCertificate plugin by default, so no separate installation or configuration is required.

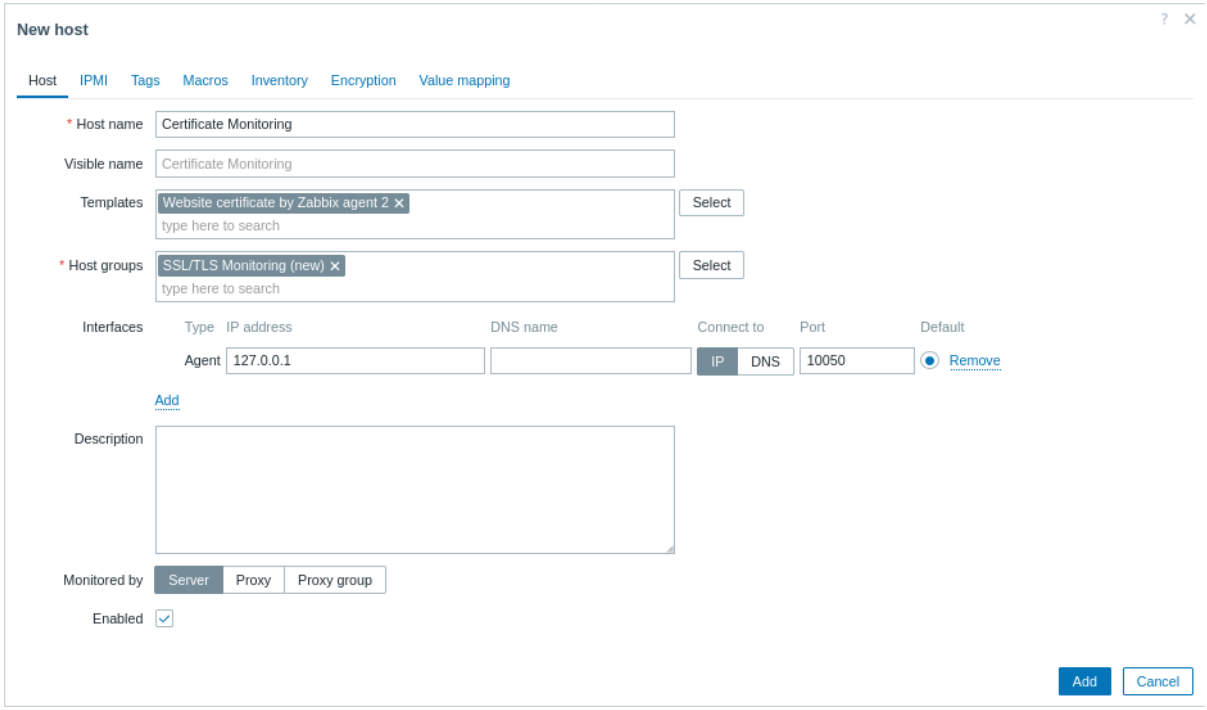
Настройка веб-интерфейса Zabbix

1. Войдите в веб-интерфейс Zabbix.

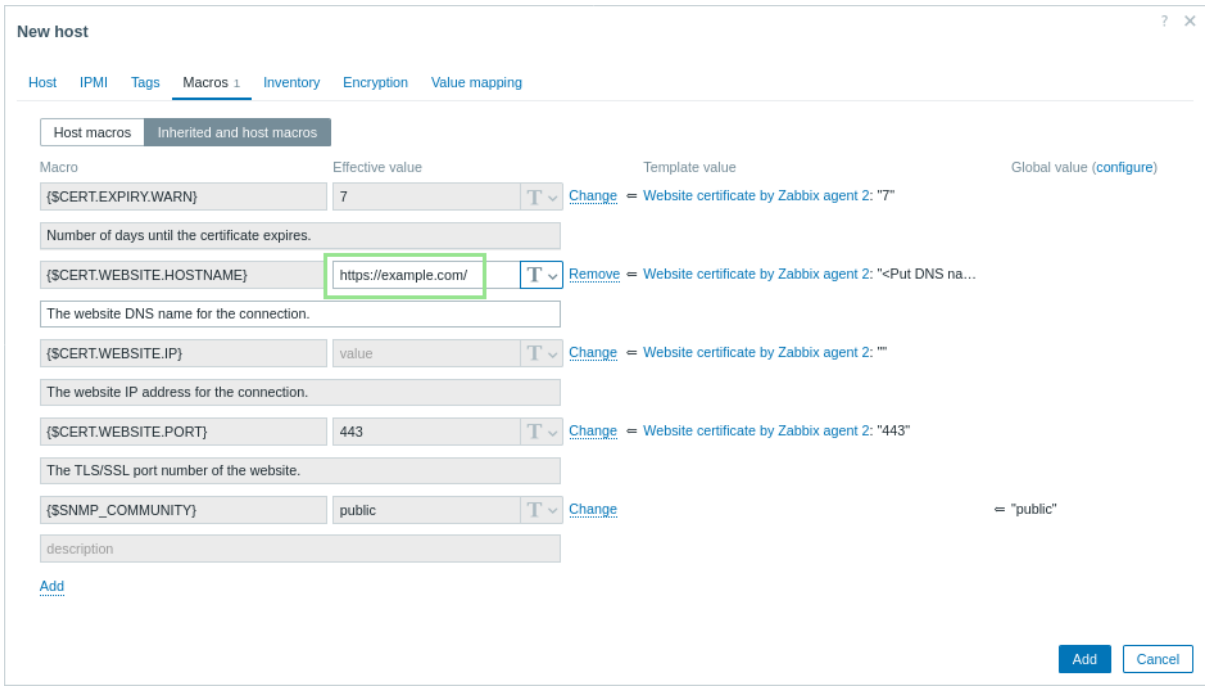
2. Перейдите в *Monitoring* > *Hosts*.

3. Нажмите на существующий узел сети, для которого вы хотите отслеживать сертификаты веб-сайтов, или создайте узел сети, если это необходимо:

- В поле *Host name* введите имя узла сети (например, ”Certificate Monitoring”).
- В поле *Templates* введите или выберите шаблон ”Website certificate by Zabbix agent 2”, который будет связан с узлом сети.
- В поле *Host groups* введите или выберите группу узлов сети (например, ”SSL/TLS Monitoring”).
- В поле *Interfaces* добавьте интерфейс типа ”Agent” и укажите IP-адрес. В этом примере используется ”127.0.0.1”.




- На вкладке *Macros* переключитесь на *Inherited and host macros*, найдите следующие макросы и нажмите *Change* рядом со значением макроса, чтобы обновить его:
  - `{$CERT.WEBSITE.HOSTNAME}` - укажите в качестве значения нужное DNS-имя веб-сайта.




4. Нажмите *Add*, чтобы добавить узел сети.

Чтобы отслеживать несколько веб-сайтов, вы можете указать в макросе `{$CERT.WEBSITE.HOSTNAME}` список их имен хостов, разделенных запятыми. При необходимости вы также можете указать в макросе `{$CERT.WEBSITE.PORT}` список портов, разделенных запятыми, где каждый порт соответствует соответствующему имени хоста в том же порядке. Например, чтобы отслеживать example.com и example.org на порту 8443:

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

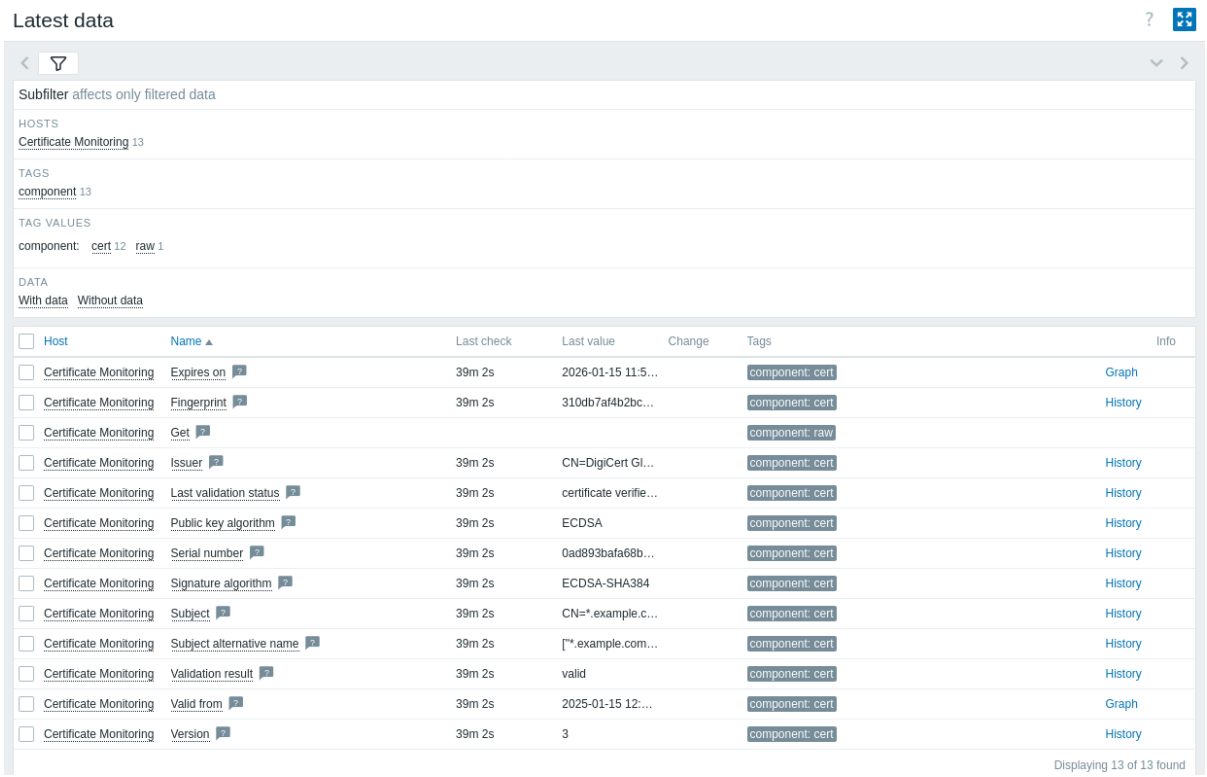
Для каждого веб-сайта, указанного в макросе `{$CERT.WEBSITE.HOSTNAME}`, Zabbix создаст соответствующий набор элементов данных и триггеров. Это позволяет выполнять отдельный мониторинг и получать отдельные оповещения для SSL-сертификата каждого веб-сайта.

| 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:… |

Просмотр собранных метрик

Поздравляем! На этом этапе Zabbix уже отслеживает нужный веб-сертификат.

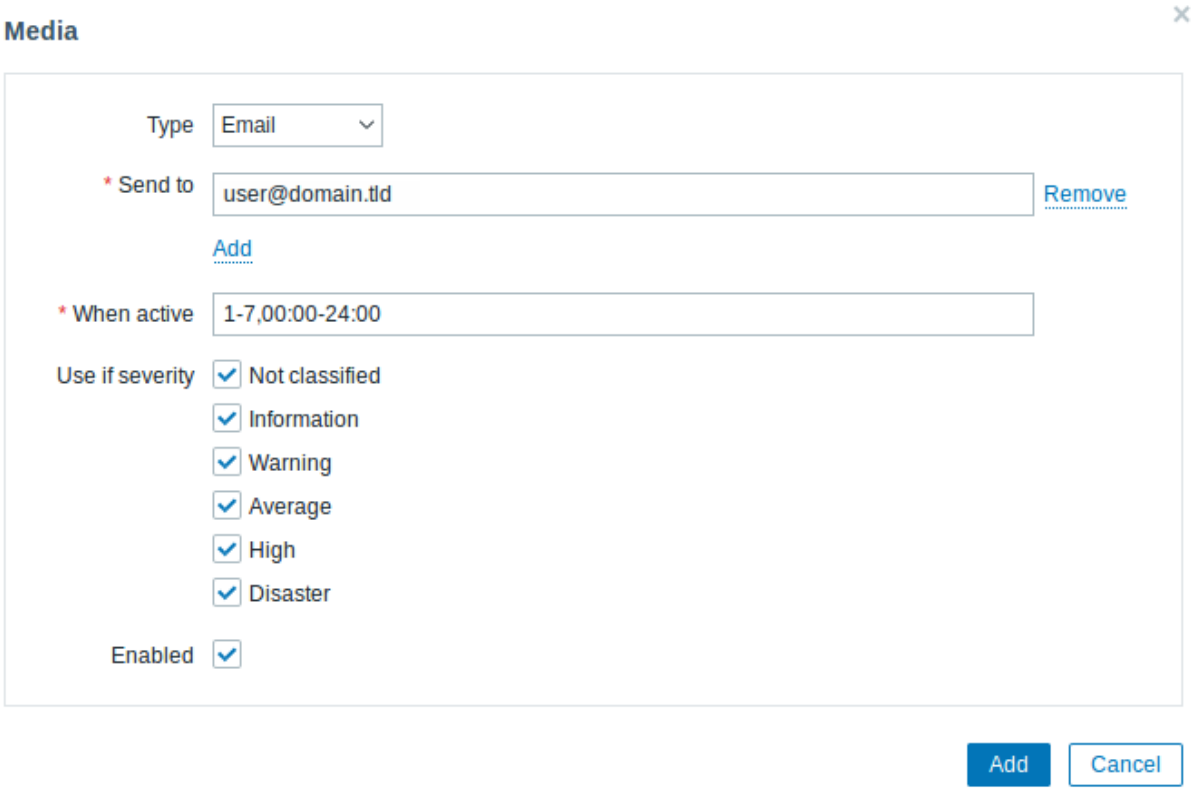
Чтобы просмотреть собранные метрики, перейдите в раздел меню *Monitoring > Hosts* и нажмите *Latest data* рядом с узлом сети, чтобы просмотреть все последние собранные метрики в виде списка, например дату истечения срока действия, издателя и субъект.




Настройка оповещений о проблемах

Zabbix может уведомлять вас о проблемах в инфраструктуре. Это руководство содержит основные шаги настройки для отправки оповещений по электронной почте.

1. Перейдите в *Настройки пользователя > Профиль*, откройте вкладку *Media* и добавьте свой адрес электронной почты.




2. Следуйте руководству по получению уведомления о проблеме.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

Проверьте вашу конфигурацию

Чтобы проверить вашу конфигурацию, мы можем смоделировать реальную проблему, обновив конфигурацию узла сети в веб-интерфейсе Zabbix.

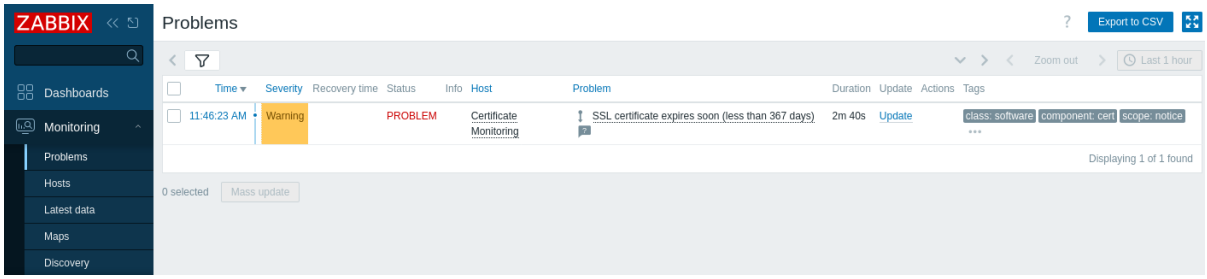
1. Откройте конфигурацию узла сети "Certificate Monitoring" в Zabbix.

2. Перейдите на вкладку *Macros* и выберите *Inherited and host macros*.

3. Нажмите *Change* рядом с ранее настроенным значением макроса `{$CERT.EXPIRY.WARN}` и задайте очень большое число дней (достаточно значения более 365 дней), чтобы получать предупреждение до истечения срока действия сертификата.

4. Нажмите *Update*, чтобы обновить конфигурацию узла сети.

5. Через несколько мгновений Zabbix обнаружит проблему "SSL certificate expires soon" с указанием количества дней до истечения срока действия. Проблема появится в *Monitoring > Problems*.




Если настроены оповещения, вы также получите уведомление о проблеме.

6. Измените значение макроса обратно на прежнее, чтобы устранить проблему и продолжить мониторинг значений сертификата.

См. также

- Zabbix agent 2 - перечисляет ключи элементов данных.
- Шаблон *Website certificate by Zabbix agent 2* - дополнительная информация о шаблоне *Website certificate by Zabbix agent 2*.
- Шаблон *Website certificate by Zabbix agent 2 active* - дополнительная информация о шаблоне *Website certificate by Zabbix agent 2 active*.
- Мониторинг веб-сайтов с помощью элементов данных Browser - как начать базовый мониторинг веб-сайтов с помощью элементов данных Browser.

## 10 Мониторинг сетевого коммутатора или маршрутизатора с помощью Zabbix

**Введение** Это руководство проведет вас через шаги, необходимые для начала базового мониторинга вашего сетевого коммутатора или маршрутизатора с помощью Zabbix. В качестве примера используется маршрутизатор Cisco, но процедура применима к любому сетевому устройству с поддержкой SNMP.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и сетевых администраторов, которые хотят быстро включить базовый мониторинг сетевых устройств. Если вам требуется глубокая настройка или расширенные параметры конфигурации, обратитесь к странице SNMP agent или разделу Configuration руководства Zabbix.

**Предварительные требования**

Перед тем как продолжить, убедитесь, что у вас есть:

- Установлены сервер Zabbix и веб-интерфейс Zabbix: установите их в соответствии с инструкциями для вашей операционной системы (см. Installation from packages и Web interface installation).
- Установлен агент Zabbix, если выполняется мониторинг локальных сетевых метрик.
- Устройство с поддержкой SNMP: сетевой коммутатор или маршрутизатор (например, маршрутизатор Cisco) с включенным SNMP.
- Установлены MIB files: установка MIB-файлов позволяет Zabbix преобразовывать числовые OID в понятные человеку имена и описания. Без надлежащей поддержки MIB вы можете видеть только числовые значения, что затрудняет настройку элементов данных и устранение неполадок.

Чтобы установить MIB-файлы в Ubuntu:

1. Установите пакет загрузчика MIB:

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

Если вам нужно добавить MIB-файлы, специфичные для производителя (например, от Cisco, Juniper), поместите их в соответствующий каталог MIB:

- Для систем на базе Linux обычные расположения включают /usr/share/snmp/mibs/ или /usr/local/share/snmp/mibs/.
- Для установок Zabbix MIB-файлы можно хранить в /var/lib/zabbix/mibs/.

Убедитесь, что переменная окружения MIBDIRS или файл snmp.conf содержит правильный путь.

Чтобы проверить, что система распознает новые MIB, используйте:

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

Подробные инструкции см. в документации вашей SNMP-библиотеки:

- Cisco MIBs
- Juniper MIBs

2. Отредактируйте `/etc/snmp/snmp.conf` и закомментируйте строку, начинающуюся с `mibs :`, чтобы разрешить системе загружать все доступные MIB.

3. Проверьте работу, выполнив `snmpwalk` (например, `snmpwalk -v 2c -c <your_community_string> <device_IP>`), и убедитесь, что OID отображаются с описательными именами.

В зависимости от вашей среды некоторые шаги в этом руководстве могут немного отличаться. Это руководство основано на среде с Ubuntu и сетевым устройством Cisco Catalyst 3750V2-24FS, находящимся под мониторингом.

Предполагается, что ваше сетевое устройство уже физически установлено и подключено.

**Configure the network device (Cisco router example)** For monitoring via SNMP, you must configure your network device to allow SNMP queries. The example below is for SNMPv2 and does not take into account existing settings. Caution: applying these commands may override current SNMP configurations.

For a Cisco router, the configuration typically involves steps written below.

**SNMPv2 Example** 1. Enable SNMP and set community string.

Log in to your Cisco router's console and enter configuration mode:

```
configure terminal
```

Then, enable SNMP by specifying a read-only community string. For example:

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

Replace `<your_community_string>` with your secure community string. Note: the RO (Read-Only) option allows SNMP to retrieve data from the device but prevents any configuration changes.

It is recommended to restrict SNMP access to only the necessary devices for security reasons. For further guidance on configuring access control lists (ACLs), refer to Cisco's official documentation.

2. Save the configuration.

Save your changes to ensure SNMP settings persist after a reboot:

```
write memory
```

**SNMPv3 Example** SNMPv3 provides enhanced security with authentication and encryption. Its configuration is more secure than SNMPv2 and should be verified against your device-specific documentation.

1. Create an SNMP group.

Configure an SNMPv3 group with privacy (encryption) enabled:

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

2. Create an SNMP user.

Add an SNMPv3 user with authentication and privacy. Replace the placeholders with your desired values:

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

3. Save the configuration:

```
write memory
```

For further details or model-specific instructions, you may refer to external Cisco SNMP configuration tutorials. This guide, however, provides the basic steps for enabling SNMP monitoring.

**Настройка веб-интерфейса Zabbix**

**Создание узла сети в веб-интерфейсе Zabbix** 1. Войдите в веб-интерфейс Zabbix.

2. Добавьте новый узел сети.

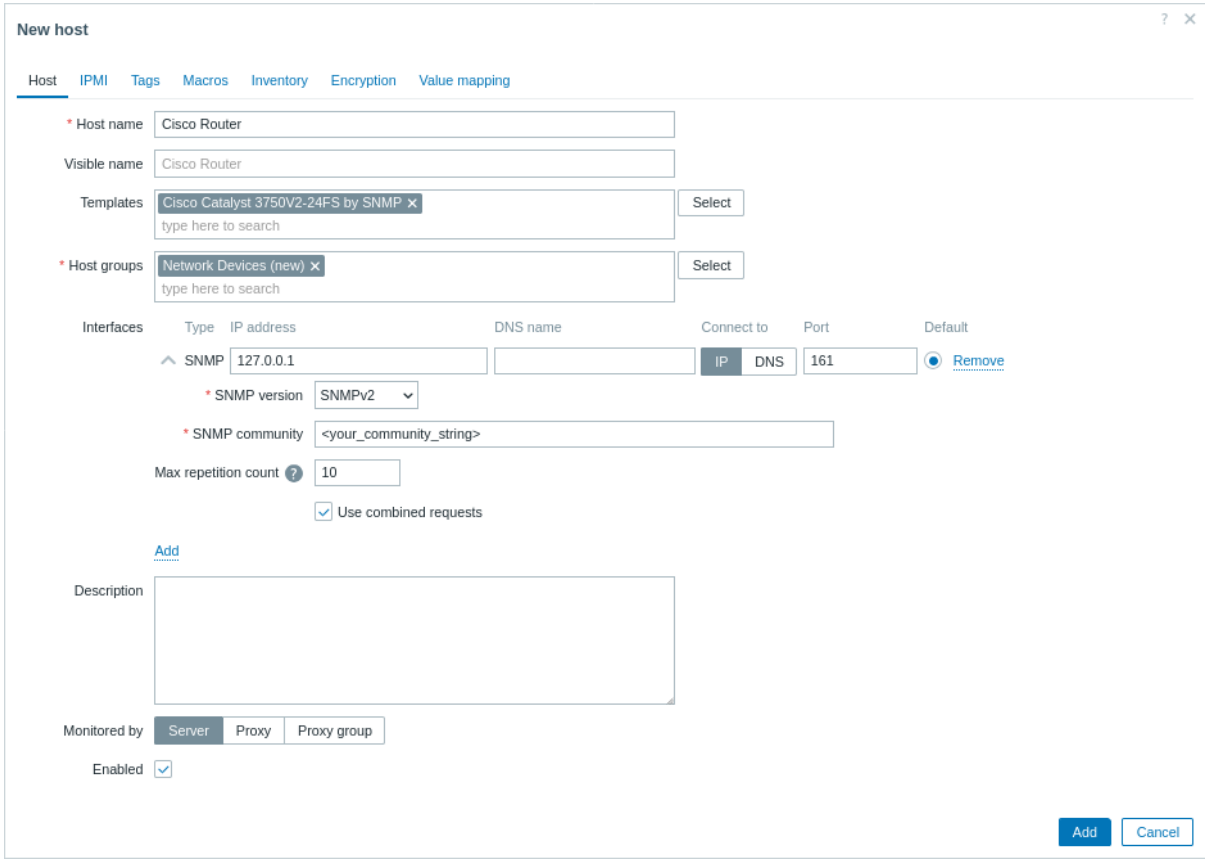
Перейдите в *Сбор данных* > *Узлы сети* и нажмите *Создать узел сети*.

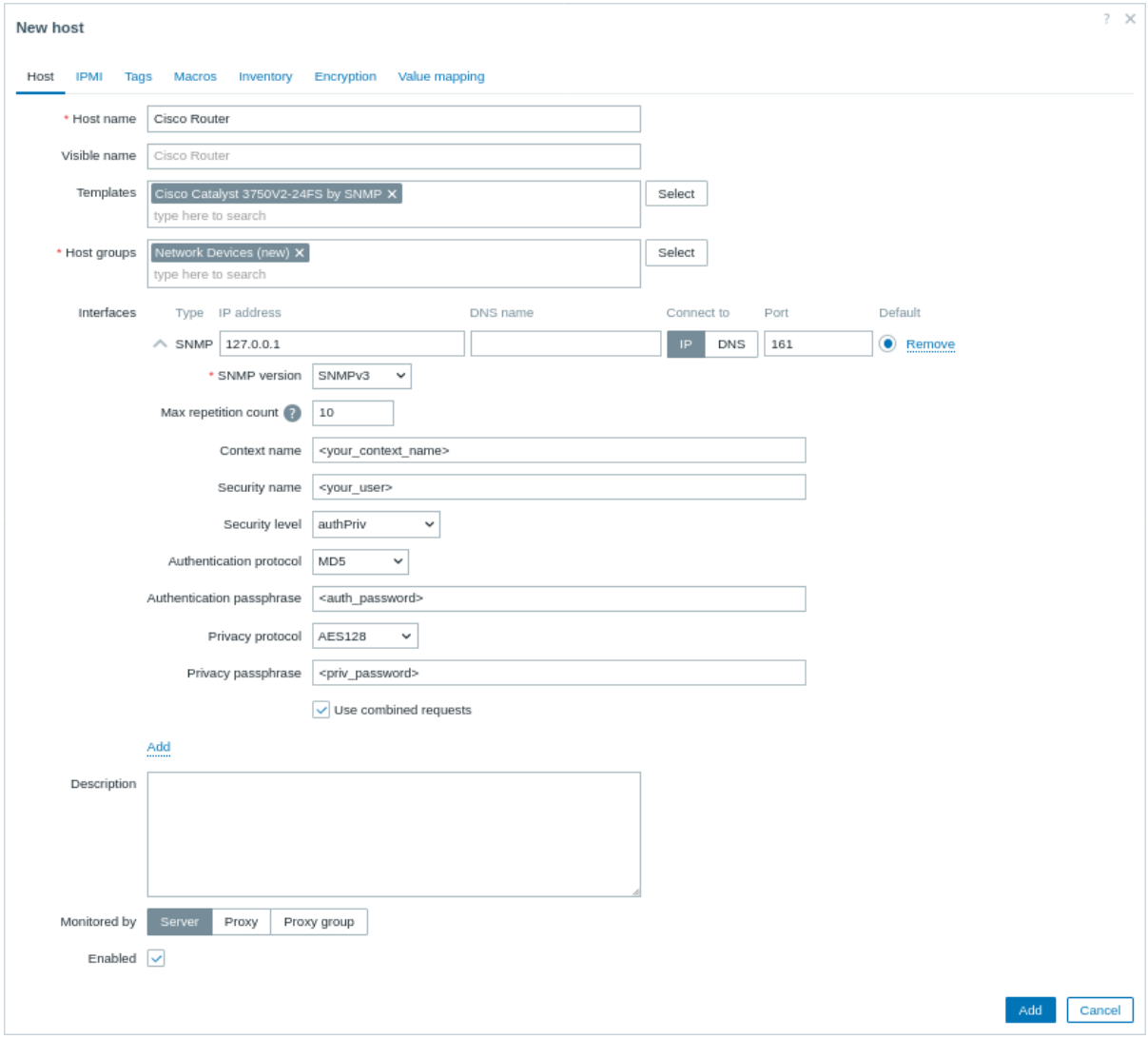
- *Имя узла сети*: введите имя вашего устройства (например, ”Cisco Router”).
- Группы узлов сети: выберите существующую группу или создайте новую, например ”Network Devices”.
- Интерфейсы:
  - Нажмите *Добавить* в разделе Interfaces.
  - Выберите *SNMP* в качестве типа интерфейса.
  - Введите IP-адрес или DNS-имя вашего маршрутизатора Cisco.
  - Установите порт SNMP по умолчанию (обычно 161).
  - Используйте раскрывающееся меню, чтобы выбрать подходящую версию SNMP (например, SNMPv2).
  - Для SNMPv1/v2 введите строку community в поле *SNMP community*. Для SNMPv3 будут запрошены дополнительные учетные данные (*Context name*, *Security name* и *Security level* и т. д.).

3. Свяжите шаблоны

В поле *Templates* выберите шаблон SNMP, который лучше всего соответствует вашему устройству. Zabbix предоставляет набор готовых SNMP templates для многих семейств устройств. Например, если вы отслеживаете устройство Cisco, выберите шаблон, соответствующий ОС или модели вашего устройства (например, Cisco IOS SNMP или Cisco Catalyst 3750<device model> SNMP).

4. Нажмите *Add*, чтобы сохранить узел сети.

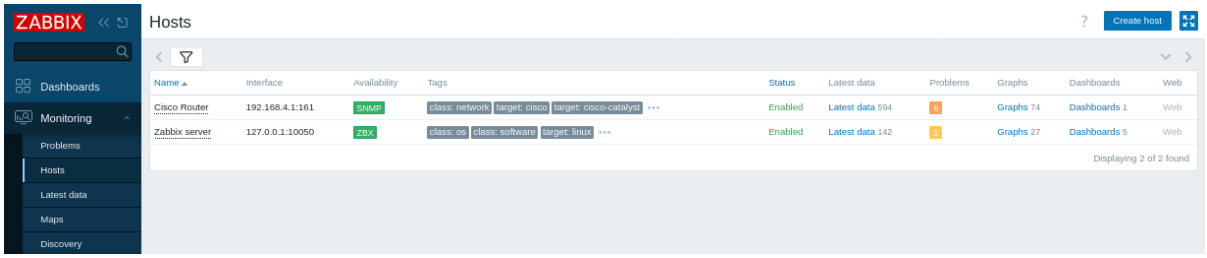




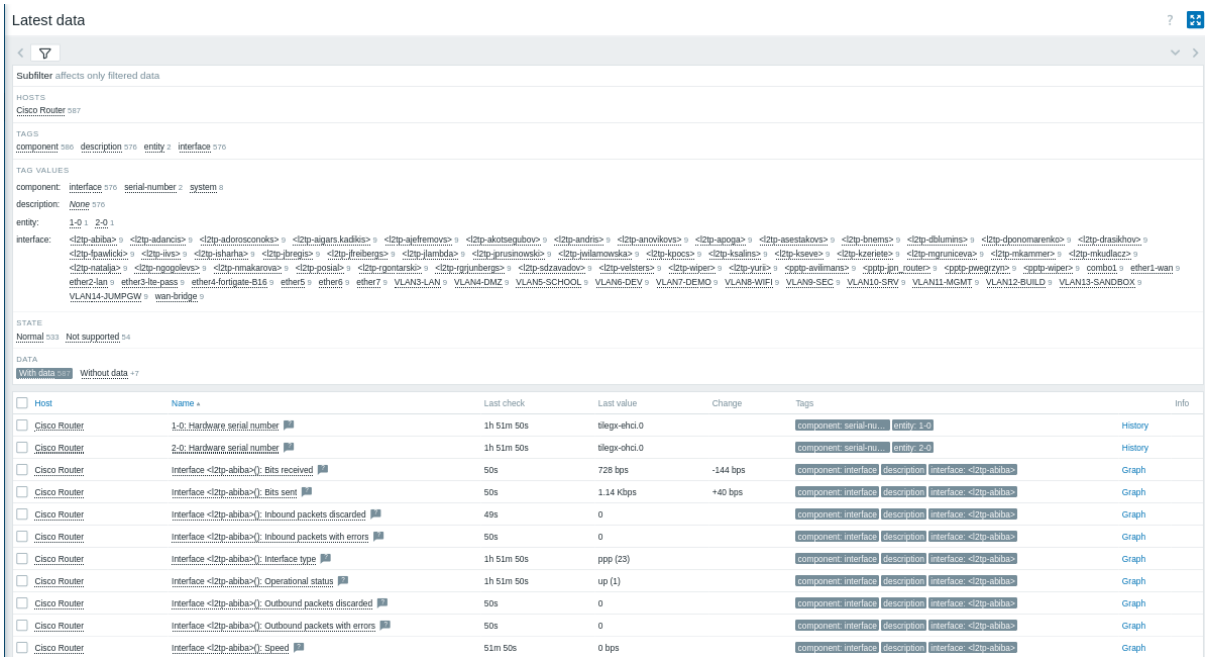

**Просмотр собранных метрик** Поздравляем! Zabbix теперь настроен для мониторинга вашего сетевого устройства.

Последние данные:

- Перейдите в Monitoring > Latest data в веб-интерфейсе Zabbix.




- Выберите ваш узел сети ”Cisco Router” (или обнаруженные узлы сети), чтобы просмотреть такие метрики, как время безотказной работы оборудования и сети, потери ICMP, ping и время отклика и т. д.




- Графики и экраны:

Чтобы визуализировать данные о производительности, нажмите *Graphs* рядом с элементами данных SNMP, чтобы увидеть подробные метрики.

В качестве следующего шага вы можете:

- Добавить пользовательские элементы данных SNMP, чтобы отслеживать дополнительные метрики.
- Настроить оповещения о проблемах, чтобы получать уведомления о возможных неполадках.

**Создание элементов данных SNMP** После настройки узла сети вы можете создать элементы данных для мониторинга конкретных метрик. Примечание: этот шаг необязателен, если вы используете шаблон, так как шаблоны уже содержат наборы элементов данных по умолчанию.

1. Определите SNMP OID:

Используйте команду `snmpwalk`, чтобы вывести список доступных OID на вашем устройстве. Например:

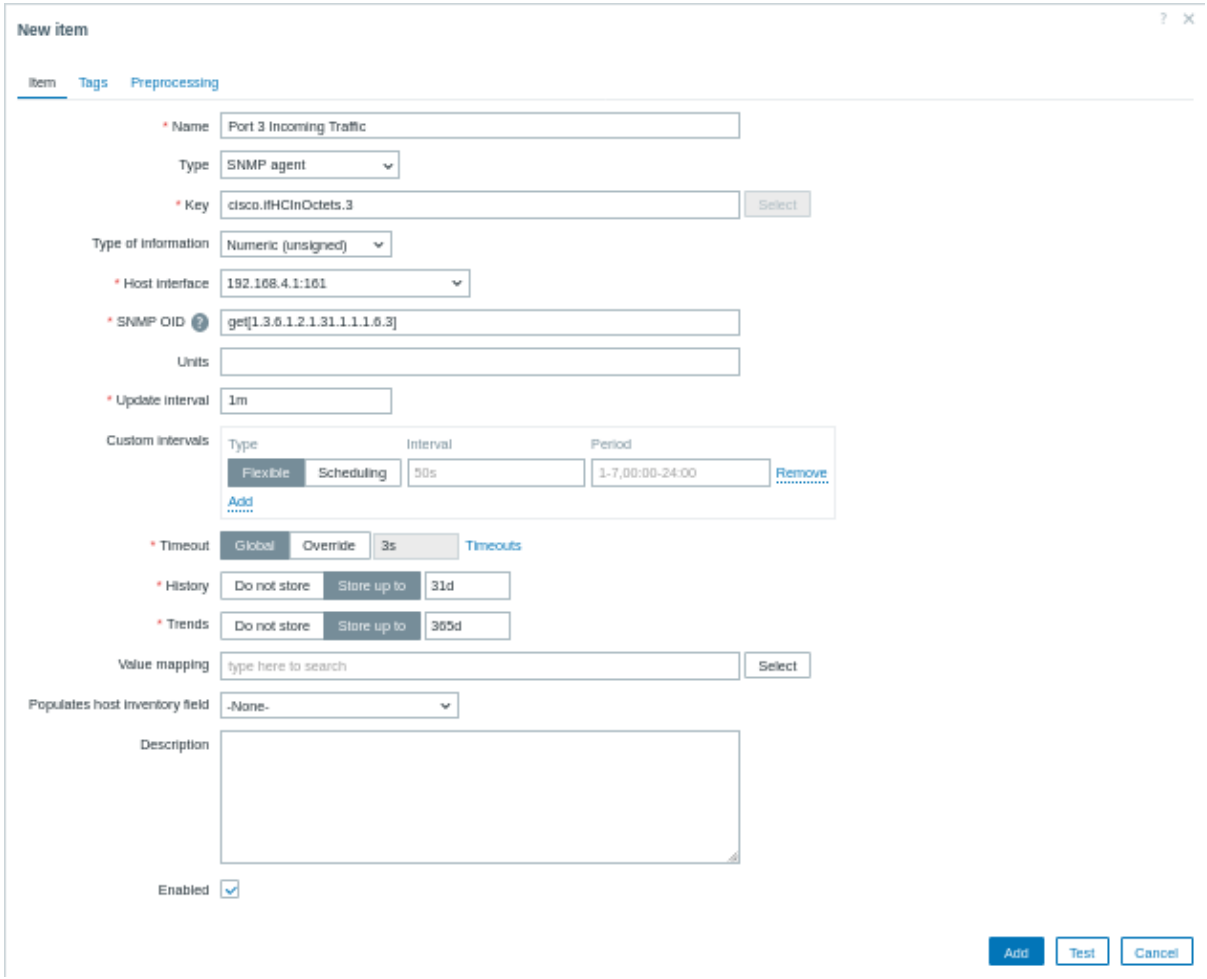
```
snmpwalk -v 2c -c <your_community_string> <device_IP> .
```

Найдите OID для метрики, которую вы хотите мониторить (например, IF-MIB::ifHCInOctets.3 для входящего трафика на порту 3). Чтобы получить числовой OID, можно использовать:

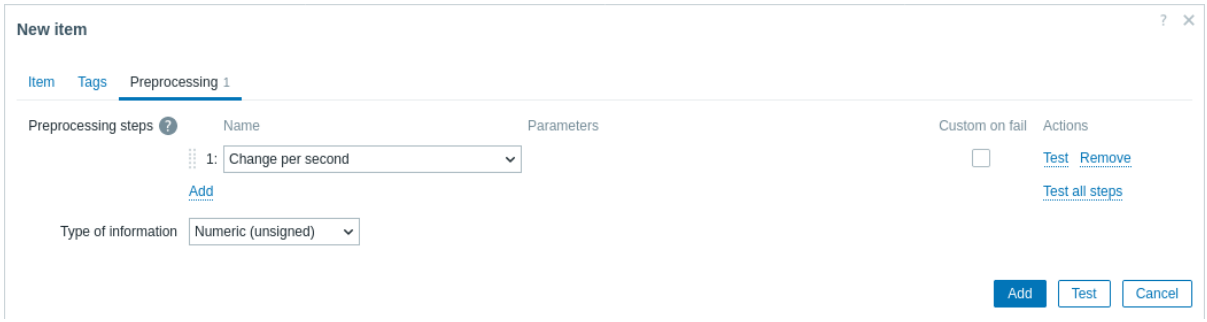
```
snmpget -v 2c -c <your_community_string> -On <device_IP> IF-MIB::ifHCInOctets.3
```

2. Создайте элемент данных SNMP:

- Перейдите в *Сбор данных* > *Узлы сети* и откройте вкладку *Элементы данных* для вашего SNMP-узла сети, затем нажмите *Создать элемент данных*.
- *Имя*: введите понятное имя (например, ”Входящий трафик на порту 3”).
- *Тип*: выберите *SNMP agent*.
- *Ключ*: укажите осмысленный ключ (например, `cisco.ifHCInOctets.3`).
- *Интерфейс узла сети*: убедитесь, что выбран SNMP-интерфейс.
- *SNMP OID*: введите OID, используя один из поддерживаемых форматов, например:
  - `get[1.3.6.1.2.1.31.1.1.1.6.3]` для одного значения;
  - `walk[1.3.6.1.2.1.31.1.1.1.6.3]` для асинхронного получения поддерева значений.




- *Предобработка* (при необходимости): если элемент данных возвращает накопительный счетчик (например, трафик интерфейса), перейдите на вкладку *Предобработка* и добавьте шаг предобработки, например "Изменение за секунду", чтобы вычислить скорость.




Чтобы получить несколько значений за одну SNMP-транзакцию, можно указать несколько OID, используя синтаксис `walk[OID1,OID2,...]`.

**Translating OIDs between Numeric and MIB Names** When working with SNMP, you might need to convert between numeric OIDs and their corresponding MIB names. This translation helps in identifying and troubleshooting metrics more easily.

- Translating a MIB name to a numeric OID: use the `snmptranslate` command with the `-On` option. For example, to translate the MIB name `IF-MIB::ifHCInOctets.3` to its numeric OID, run:

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

This command might output:

```
.1.3.6.1.2.1.31.1.1.1.6.3
```

- Translating a numeric OID to its MIB name: use the `snmptranslate` command with the `-IR` (or `-m ALL`) option to reverse the translation. For example, to translate the numeric OID `.1.3.6.1.2.1.31.1.1.1.6.3` back to its MIB name, run:

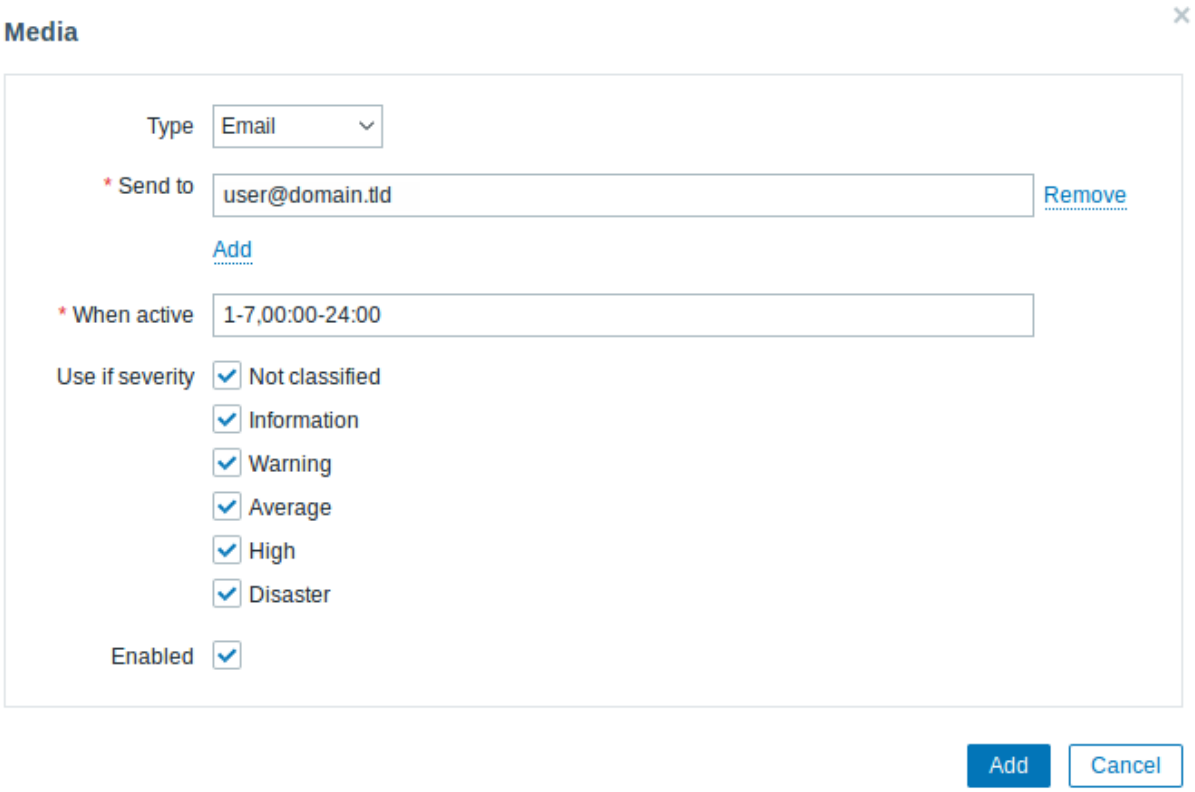
```
snmptranslate -IR -On .1.3.6.1.2.1.31.1.1.1.6.3
```

This command might output:

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

**Настройка оповещений о проблемах** Это руководство содержит основные шаги настройки отправки оповещений по электронной почте.

1. Перейдите в *Настройки пользователя > Профиль*, откройте вкладку *Media* и добавьте свой email.




2. Следуйте инструкциям из руководства Получение уведомления о проблеме.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

**Проверьте вашу конфигурацию** Чтобы убедиться, что Zabbix корректно обнаруживает проблемы с производительностью сети, смоделируйте реальную проблему, увеличив порог времени отклика ICMP ping.

1. Откройте конфигурацию узла сети "Cisco Router" в Zabbix.

2. Перейдите на вкладку *Macros* и выберите *Inherited and host macros*.

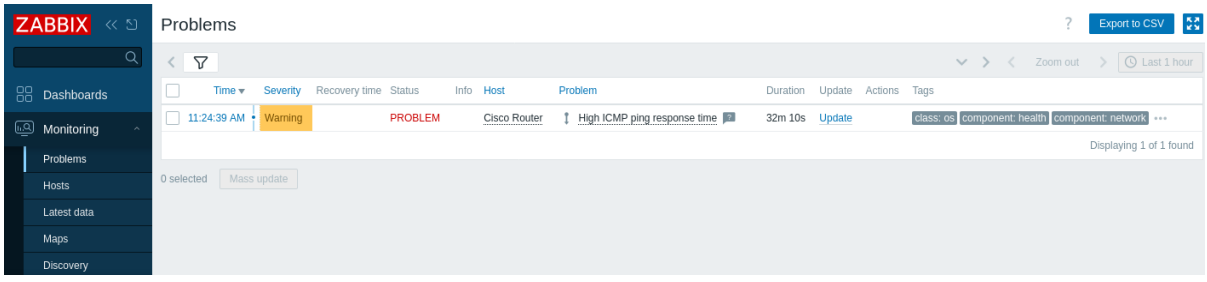
3. Найдите макрос `{$ICMP_RESPONSE_TIME_WARN}` (или аналогичный макрос порога времени отклика).

4. Установите очень низкое значение (например, 0.001), чтобы сработало оповещение, когда время отклика ping превысит это значение.

5. Нажмите *Update*, чтобы применить изменения.

6. Подождите несколько минут, пока Zabbix обнаружит смоделированную проблему.

7. Перейдите в *Monitoring* > *Problems*, чтобы убедиться, что появилось оповещение (например, "High ICMP ping response time").




Если оповещения настроены, вы также должны получить оповещение о проблеме.

8. Верните значение макроса к исходному и нажмите *Update*, чтобы сохранить изменения.

9. Убедитесь, что проблема устранена и исчезла из раздела *Problems*.

**Troubleshooting SNMP Monitoring** If you notice that the SNMP icon in the Zabbix frontend appears RED or no data is collected, try the following steps:

1. Check SNMP connectivity.

For SNMPv2 run the following command from your Zabbix server:

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

This command verifies that the device responds to SNMP queries.

For SNMPv3, include the appropriate SNMPv3 credentials:

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

This verifies that SNMPv3 credentials are correct and the device is responding securely.

2. Ensure that MIB files are installed and enabled as described in the prerequisites. To ensure that, following command must not give error when you query a network device:

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

This should return translated OIDs without errors.

3. Confirm that the SNMP version and credentials configured in Zabbix match those set on your device. For instance, review the SNMP settings in the Zabbix host configuration and verify them against your device's configuration. On a Cisco device, you might check the SNMP settings by running:

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

This ensures that the community string (for SNMPv2) or SNMPv3 user details are correct.

4. Verify that SNMP is correctly enabled on your network device. On a Cisco router, log in to the console and run:

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

This command displays the active SNMP configuration and helps confirm that SNMP is properly configured.

5. Ensure that no firewalls or network issues are blocking SNMP traffic (typically on port 161) between the Zabbix server and the device. You can test connectivity using:

nc -zv <device_IP> 161

`nc -zv` checks if port 161 is open and listening on the device.

Additionally, if you are using UFW on Ubuntu, check the firewall status:

```
sudo ufw status
```

Or, for iptables:

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

6. Review the Zabbix server log files for any SNMP-related errors to help pinpoint the issue:

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

`tail -f` allows you to monitor log updates in real time.

**См. также:**

- Создание элемента данных - узнайте, как добавлять дополнительные метрики.
- Агент SNMP - дополнительная информация о мониторинге SNMP с помощью Zabbix.
- Стандартизированные шаблоны для сетевых устройств - информация о доступных шаблонах SNMP.
- Обнаружение OID SNMP - дополнительная информация об обнаружении SNMP на коммутаторе.
- Настройка правила сетевого обнаружения - дополнительная информация о том, как настроить правило сетевого обнаружения, используемое Zabbix для обнаружения узлов сети и служб.

### 11 Мониторинг журнала событий Windows с использованием активных проверок

Введение

В этом руководстве объясняется, как отслеживать журналы событий Windows с помощью Zabbix, используя активные проверки. С помощью ключей элементов данных Zabbix, специфичных для Windows, вы можете собирать и анализировать критические события (например, неудачные попытки входа в систему, системные ошибки и т. д.) в реальном времени.

**Для кого предназначено это руководство**

Это руководство предназначено для новых пользователей Zabbix и сетевых администраторов, которые хотят отслеживать журналы событий Windows. Для получения сведений о дополнительных параметрах настройки см. документацию по ключам элементов данных, специфичным для Windows.

**Предварительные требования**

Перед тем как продолжить работу с этим руководством, вам необходимо скачать и установить сервер Zabbix и веб-интерфейс Zabbix в соответствии с инструкциями для вашей ОС. Вам также необходимо, чтобы агент Zabbix был скачан и установлен на компьютере Windows, который вы хотите отслеживать.

Настройка агента Zabbix для мониторинга журнала событий Windows

1. Откройте `zabbix_agentd.conf` (путь по умолчанию `C:\Program Files\Zabbix Agent\zabbix_agentd.conf`) на вашем узле сети Windows и убедитесь, что параметр ServerActive установлен в IP-адрес вашего сервера Zabbix, а параметр Hostname соответствует имени узла сети, которое будет задано в веб-интерфейсе Zabbix. Это позволяет агенту запрашивать активные проверки для своего узла сети и у указанного сервера Zabbix. Например:

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

2. Перезапустите службу агента Zabbix, чтобы применить изменения:

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

3. Проверьте, что узел сети Windows работает:

- Убедитесь, что служба агента Zabbix запущена на узле сети Windows.
- Проверьте, что узел сети Windows может подключаться к серверу Zabbix через порт 10051. Чтобы проверить подключение с узла сети Windows, откройте PowerShell и выполните следующую команду:

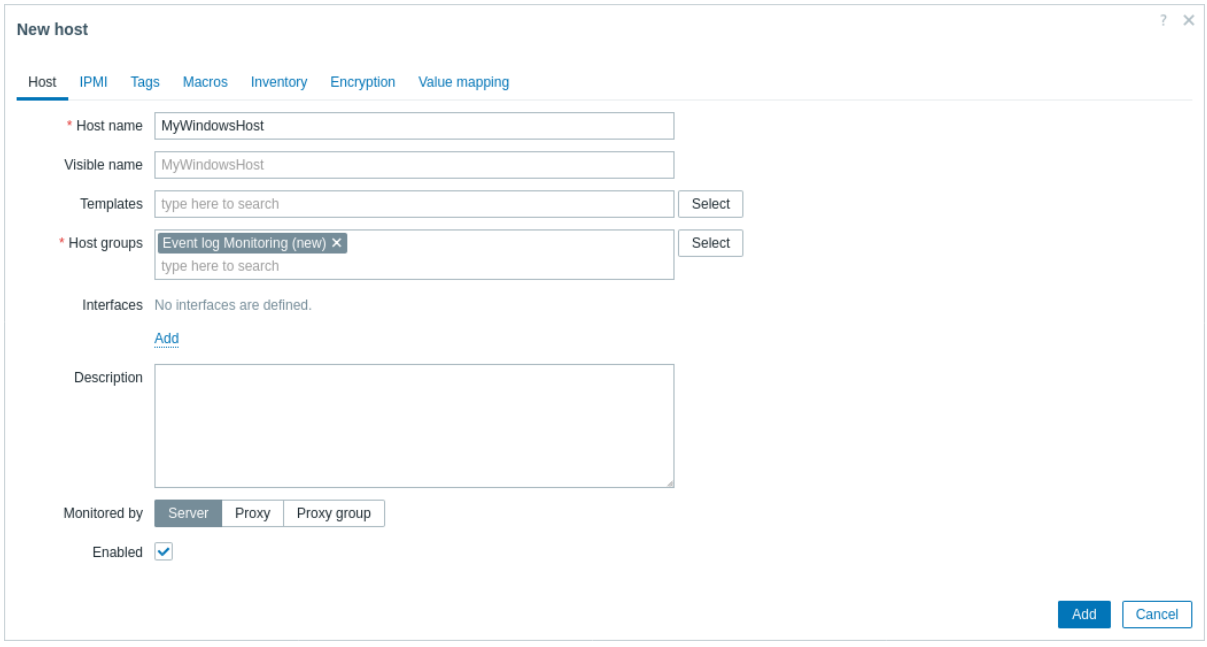
```
Test-NetConnection -ComputerName <Zabbix-server-IP> -Port 10051
```

Настройка веб-интерфейса Zabbix

1. Перейдите в *Сбор данных* > *Узлы сети* и создайте узел сети:

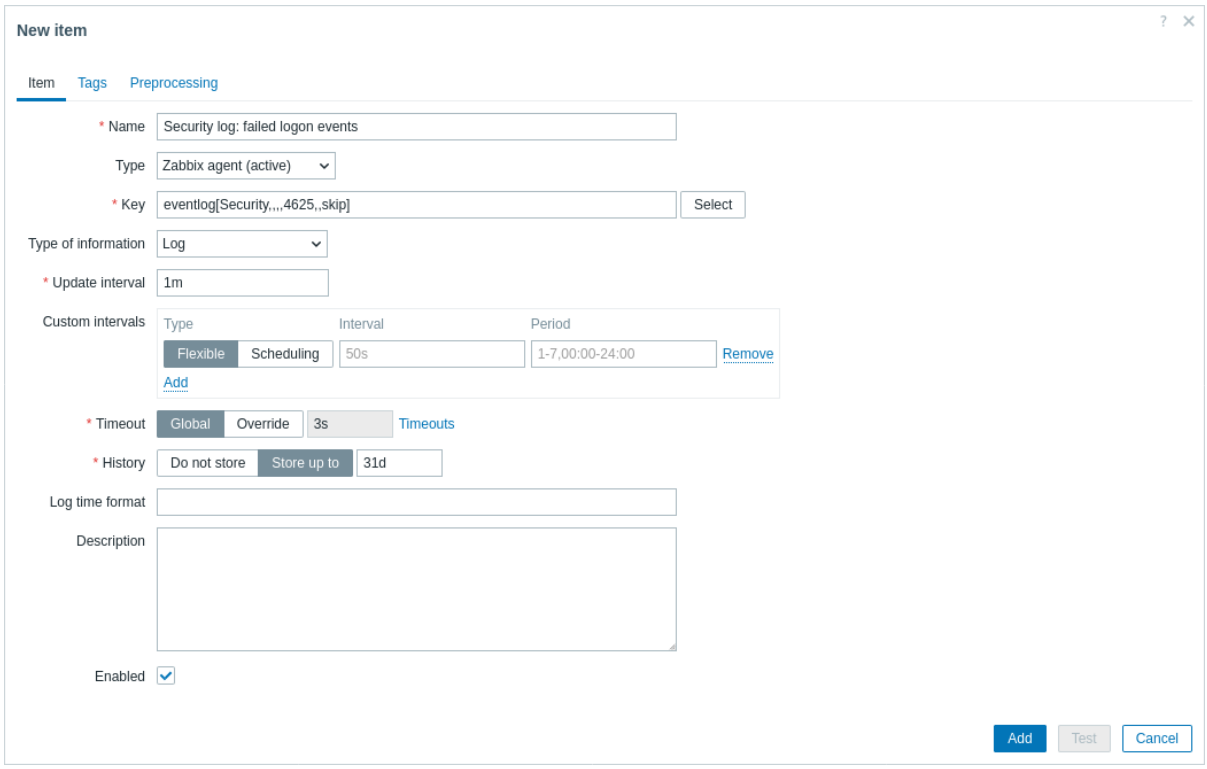
- В поле *Host name* введите имя узла сети (например, ”MyWindowsHost”).
- В поле *Host groups* введите или выберите группу узлов сети (например, ”Event log Monitoring”).
- Нажмите *Add*, чтобы сохранить настроенный узел сети.

**Note:**
В поле *Templates* можно добавить шаблон ”Windows by Zabbix agent active”, чтобы помочь в поиске неисправностей, наблюдая, обновляются ли другие активные элементы данных на том же узле сети.



2. Создайте новый элемент данных со следующими параметрами:

- В поле *Name* введите описательное имя элемента данных (например, ”Security log: failed logon events”).
- В раскрывающемся списке *Type* выберите ”Zabbix agent (active)” (требуется для мониторинга журнала событий).
- В поле *Key* используйте ключ элемента данных eventlog. Например, чтобы отслеживать неудачные попытки входа в систему (Event ID: 4625) в журнале Security и игнорировать записи, которые старше последней проверки элемента данных (с использованием параметра `skip`), введите следующий ключ элемента данных: `eventlog[Security,,,,4625,,skip]`
- В раскрывающемся списке *Type of information* выберите ”Log”.



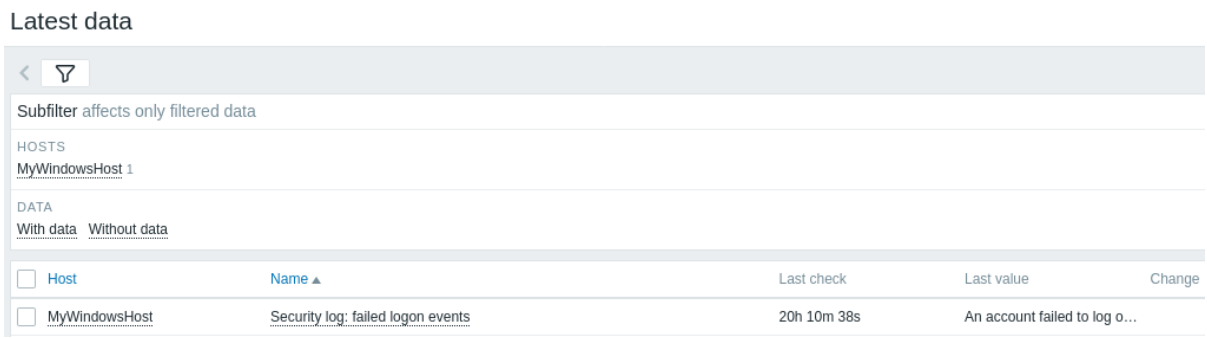

3. Нажмите *Add*, чтобы сохранить элемент данных.

Тестирование и просмотр собранных метрик

Поздравляем! Zabbix теперь настроен на сбор журналов событий Windows. Чтобы проверить, что журналы событий собираются, вы можете протестировать элемент данных ”Security log: failed logon events”, выйдя из своей учетной записи Windows и попытавшись войти с неверными учетными данными.

Затем просмотрите собранные журналы в веб-интерфейсе Zabbix:

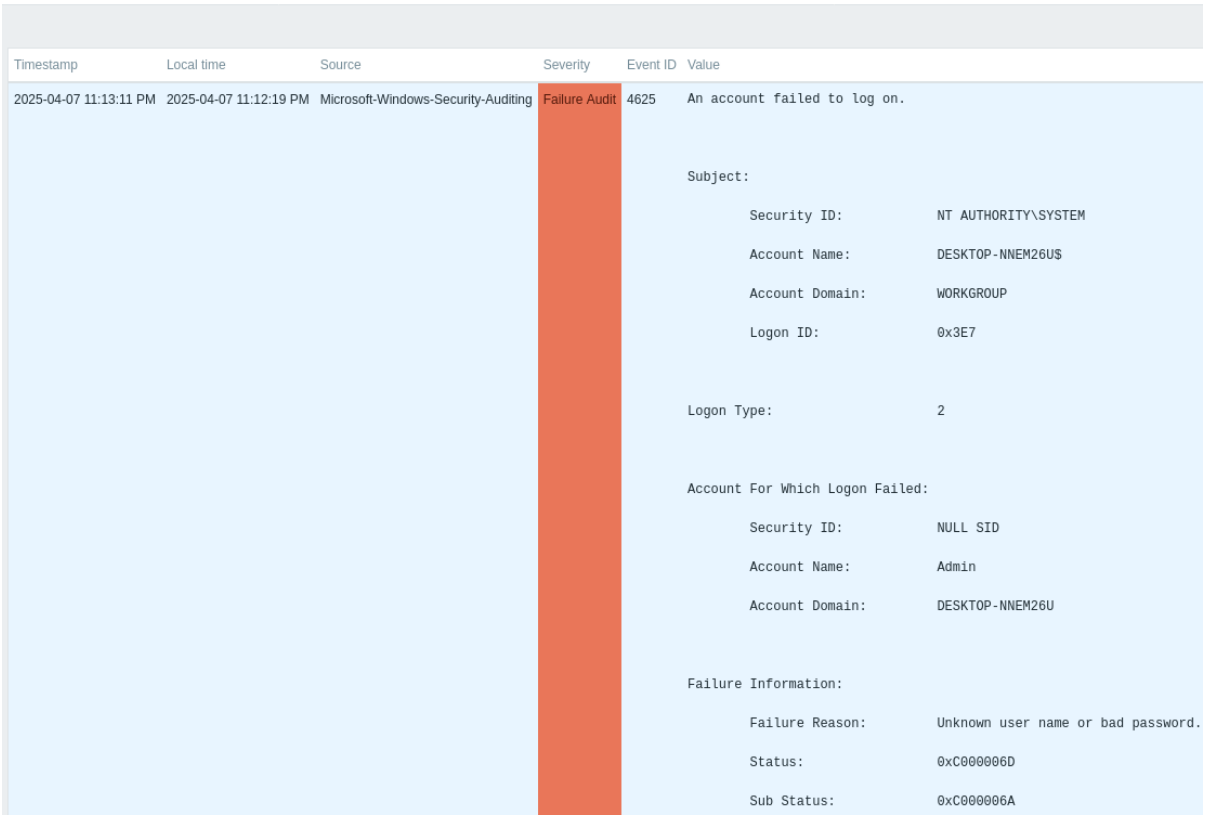
1. Перейдите в *Monitoring* > *Latest data* в веб-интерфейсе Zabbix.




2. Отфильтруйте по узлу сети ”MyWindowsHost” в поле *Name*.

3. Нажмите *History*, чтобы просмотреть записанные значения журнала.

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.<br><br>Subject:<br>Security ID: NT AUTHORITY\SYSTEM<br>Account Name: DESKTOP-NNEM26U$<br>Account Domain: WORKGROUP<br>Logon ID: 0x3E7<br><br>Logon Type: 2<br><br>Account For Which Logon Failed:<br>Security ID: NULL SID<br>Account Name: Admin<br>Account Domain: DESKTOP-NNEM26U<br><br>Failure Information:<br>Failure Reason: Unknown user name or bad password.<br>Status: 0xC000006D<br>Sub Status: 0xC000006A |

4. Если значения журнала отсутствуют, перейдите к разделу Troubleshooting этого руководства.

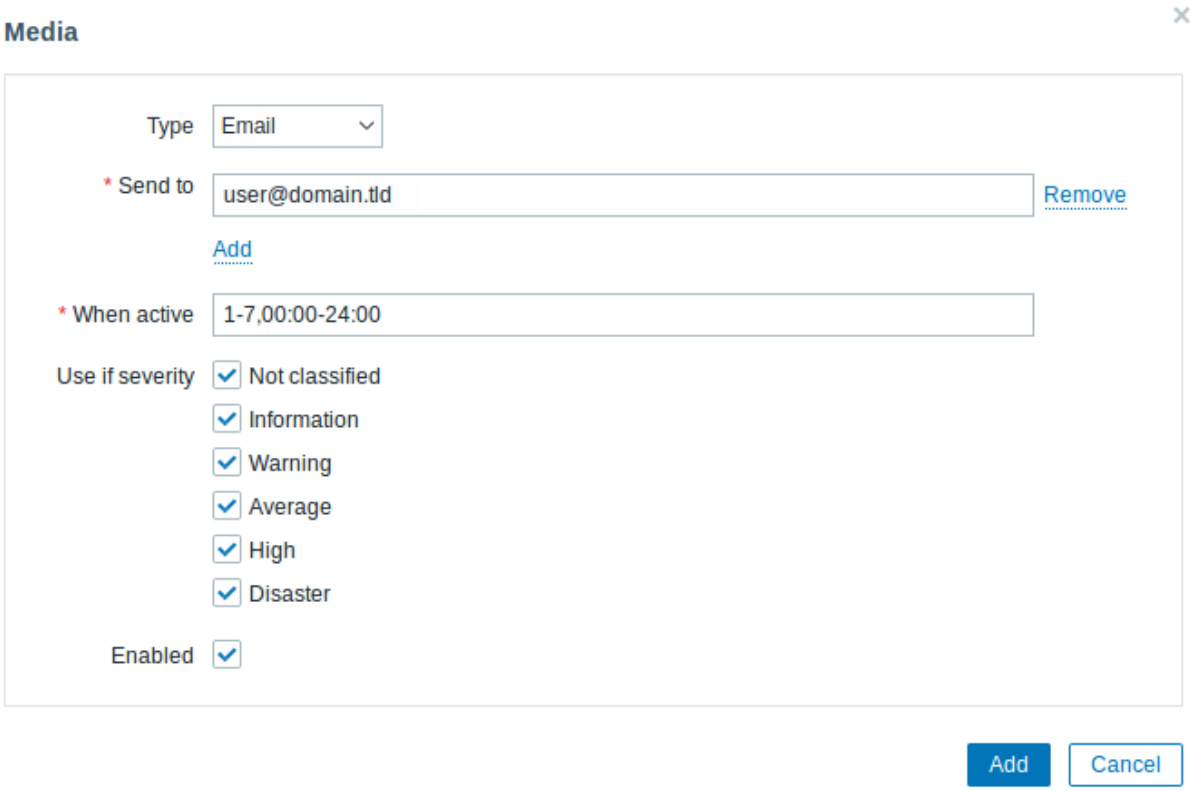
Настройка оповещений о проблемах

Это руководство содержит основные шаги настройки отправки оповещений по электронной почте.

1. Перейдите в Data collection > Hosts, чтобы определить триггер, который срабатывает, когда элемент данных журнала событий записывает нужный вам шаблон. Например, чтобы обнаруживать неудачные попытки входа в журнале Security, используйте функцию find():

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

2. Перейдите в *User settings > Profile*, откройте вкладку *Media* и добавьте свой email.




3. Следуйте руководству по получению уведомления о проблеме.

В следующий раз, когда Zabbix обнаружит проблему, вы должны получить оповещение по электронной почте.

Troubleshooting

If you encounter issues with collecting or viewing Windows event logs, use the tips below to identify and resolve common problems:

1. On Zabbix server (Linux) list your iptables rules with the following command:

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

and verify there's an ACCEPT rule for TCP port 10051.

2. Make sure your `eventlog[...]` key uses the exact log name (case-sensitive), event ID, mode (e.g. skip) and other parameters exactly as shown in the Windows-specific item keys.

**См. также:**

- Создание элемента данных - узнайте, как добавить дополнительные метрики.
- Агент Zabbix в Microsoft Windows - подробные инструкции по установке.
- Мониторинг Windows с помощью агента Zabbix - подробное руководство по настройке базового мониторинга для компьютеров Windows с использованием агента Zabbix.
- Ключи элементов данных, специфичные для Windows - подробная информация о ключах элементов данных для Windows, поддерживаемых агентами Zabbix, включая ключи для мониторинга журнала событий.
- Мониторинг файлов журналов - инструкции по настройке Zabbix для централизованного мониторинга и анализа файлов журналов, применимые к журналам событий Windows.

## Zabbix Cloud

## Zabbix Cloud

Ознакомьтесь с доступными параметрами конфигурации и средствами управления доступом для организаций и узлов в Zabbix Cloud.

Обратите внимание, что узлы Zabbix Cloud в настоящее время работают под управлением **Zabbix 7.0**.<br> Подробную информацию о Zabbix 7.0 см. в официальной документации.

## Настройка и управление Zabbix Cloud

**Начните использовать Zabbix Cloud** Пошаговые инструкции, которые помогут быстро запустить ваш экземпляр Zabbix Cloud, включая начальную настройку и подключение контролируемых устройств.

**Конфигурация узла сети** Узнайте, как настроить узел Zabbix Cloud, уделяя особое внимание доступу, обработке данных и настройкам ресурсов.

**Добавление пользователей** Узнайте, как добавлять пользователей и назначать им роли в организации Zabbix Cloud и ее узлах.

## Узнайте о Zabbix Cloud

**Zabbix Cloud vs. локальное развертывание** Сравните различия в управлении, масштабируемости и обслуживании между этими двумя вариантами развертывания и узнайте, какой из них лучше всего подходит для вашего рабочего процесса.

**Изучите Zabbix Cloud** Получите обзор Zabbix Cloud, включая его ключевые возможности, преимущества и то, что входит в подписку. Ответы на часто задаваемые вопросы по настройке и использованию см. в FAQ.

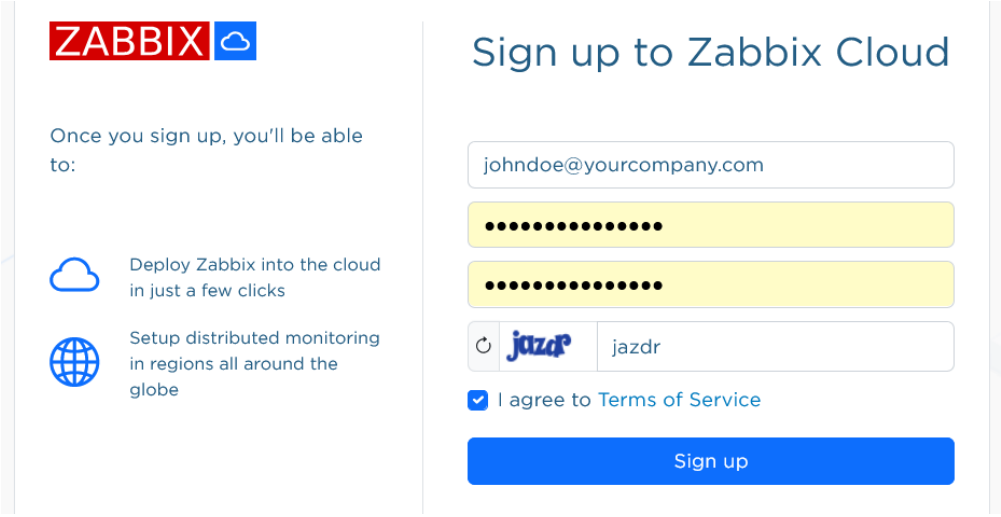
## Развертывание Zabbix в облаке

Чтобы начать работу с Zabbix в облаке, выполните следующие действия:

- Войдите в систему или зарегистрируйтесь на cloud.zabbix.com
- Создайте облачный узел
- Начните использовать Zabbix в облаке

**Вход в систему или регистрация** Вы можете войти на cloud.zabbix.com с существующей учетной записью Google, GitHub или Microsoft.

Либо перейдите на страницу регистрации, чтобы создать учетную запись.



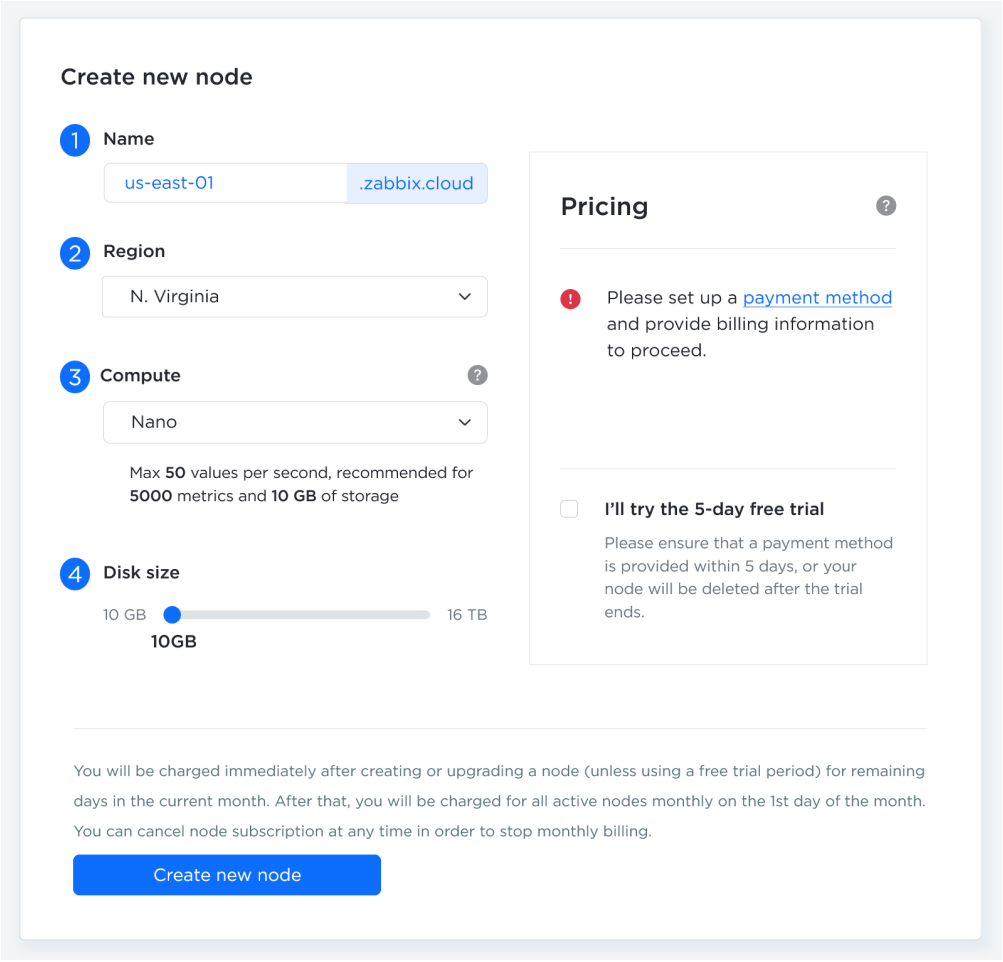

После регистрации вы получите подтверждение новой учетной записи по электронной почте.

Перейдите на cloud.zabbix.com, чтобы войти в систему. При входе вам будет предложен одноразовый пароль. Этот пароль будет отправлен вам по электронной почте.

**Создание облачного узла** Облачный узел — это экземпляр Zabbix, размещенный в облаке. Он включает сервер Zabbix, базу данных и веб-интерфейс.

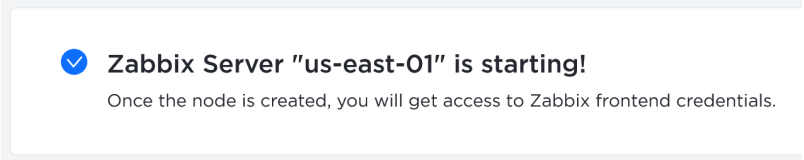
После входа в систему вы увидите, что для вашей учетной записи по умолчанию уже существует организация *Private organization*. Позже вы можете переименовать организацию по умолчанию.

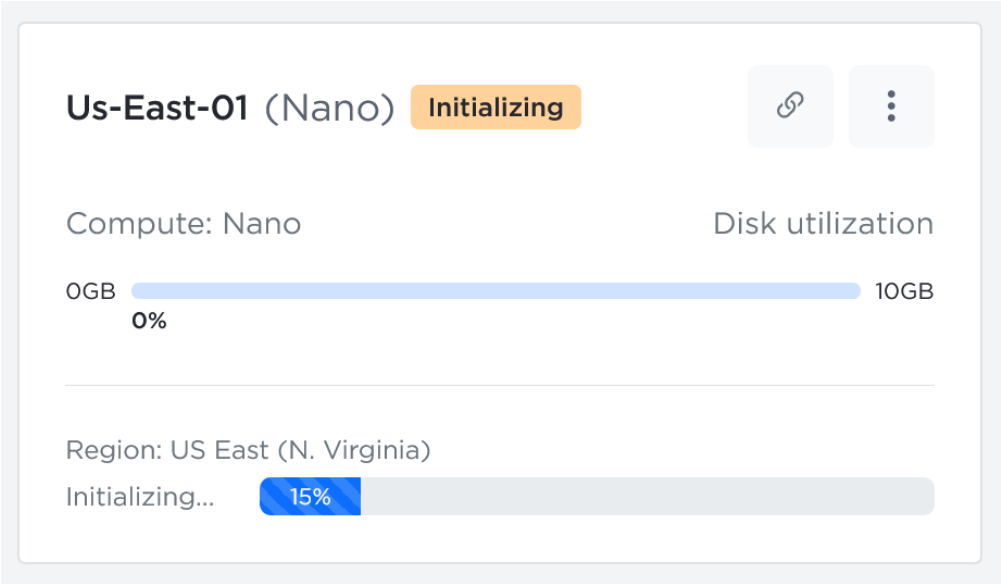
Нажмите *Create New Node*. Откроется форма настройки узла.

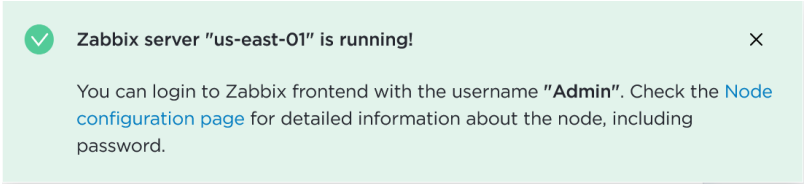



| Parameter | Description |
|---|---|
| *Name* | Выберите уникальное имя узла.<br>Имя узла формирует уникальный URL для облачного сервера/веб-интерфейса в формате `<your-name>.zabbix.cloud`. |
| *Region* | Выберите ближайшее расположение центра обработки данных для минимальной сетевой задержки. |
| *Compute* | Выберите вычислительную мощность для узла.<br>Чтобы определить, какой уровень ресурсов вам нужен, см. сведения о ценах. При необходимости вы всегда сможете позже увеличить мощность. Перед созданием не пробного узла потребуется способ оплаты. Добавьте платежную информацию через меню *Billing*.<br>Доступна **бесплатная пробная версия** на 5 дней для тестирования узла уровня *Nano*. В течение пробного периода плата взиматься не будет, но после его окончания узел будет удален, если его не перевести на платный тариф. |
| *Disk size* | Ползунок по умолчанию установлен на рекомендуемый минимальный размер в зависимости от уровня.<br>Увеличьте его, если вы храните большие объемы истории или трендов. Имейте в виду, что большие диски влекут за собой более высокие ежемесячные расходы и что уменьшить размер диска самостоятельно нельзя — для этого потребуется обратиться в службу поддержки. |

После заполнения формы нажмите *Create New Node*. Пока узел инициализируется, отображается соответствующее сообщение, и узел уже появляется в разделе Organizations.





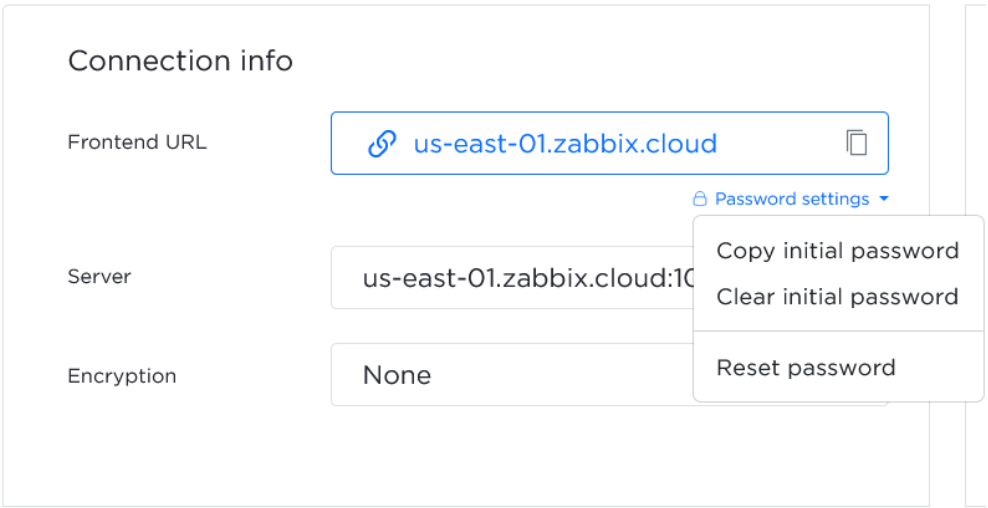




**Начало работы с Zabbix**

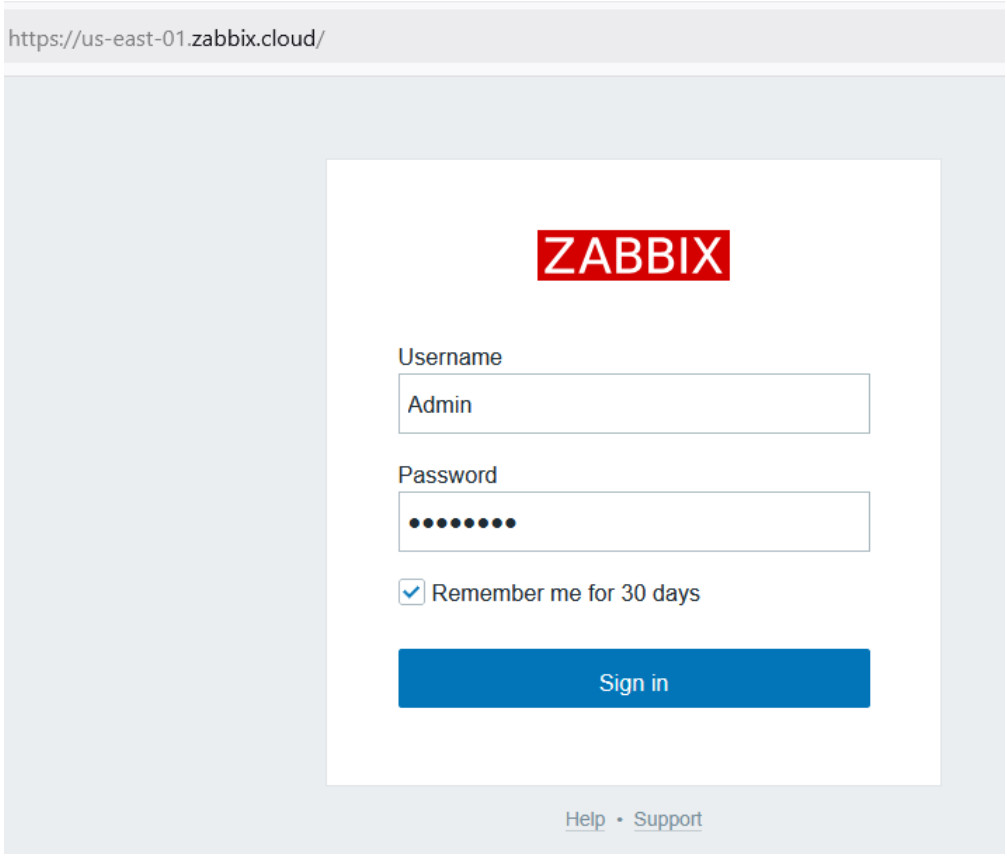
Когда статус узла меняется с *Initializing* на *Running*, вы можете войти в веб-интерфейс Zabbix.

Чтобы получить начальный пароль для веб-интерфейса, нажмите на созданный узел.

Вы будете перенаправлены в настройки узла (обзор, фильтры доступа, шифрование, резервные копии и т. д.). В разделе **Connection info** нажмите *Password settings* и выберите *Copy initial password* в раскрывающемся списке.




Нажмите URL веб-интерфейса, чтобы открыть веб-интерфейс Zabbix (`us-east-01.zabbix.cloud`).




Войдите, используя следующие учетные данные:

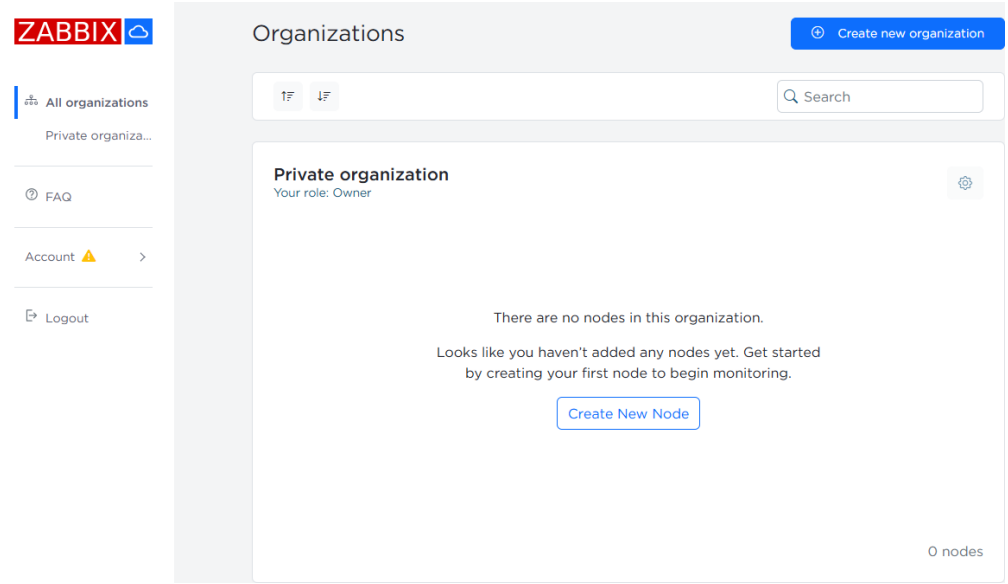
- *Username* - `Admin`
- *Password* - укажите начальный пароль (например, HzTG9t7Y)

Чтобы начать мониторинг, необходимо задать фильтры доступа - IP-адрес или адреса, которым разрешено подключаться к облачному серверу Zabbix. Должен быть задан как минимум один фильтр.

**Управление организацией** Организация — это контейнер для одного или нескольких узлов сети. Организации также помогают вам структурировать ресурсы и распределять обязанности, например:

- Управление узлами сети
- Выставление счетов

При входе в Zabbix Cloud отображается страница *Все организации*. По умолчанию для вашей учетной записи создается частная организация.




Как владелец этой организации, вы можете управлять пользователями, администраторами, узлами сети и названием

организации. Нажмите кнопку настроек , чтобы открыть параметры управления.

## Конфигурация узла сети

### Обзор

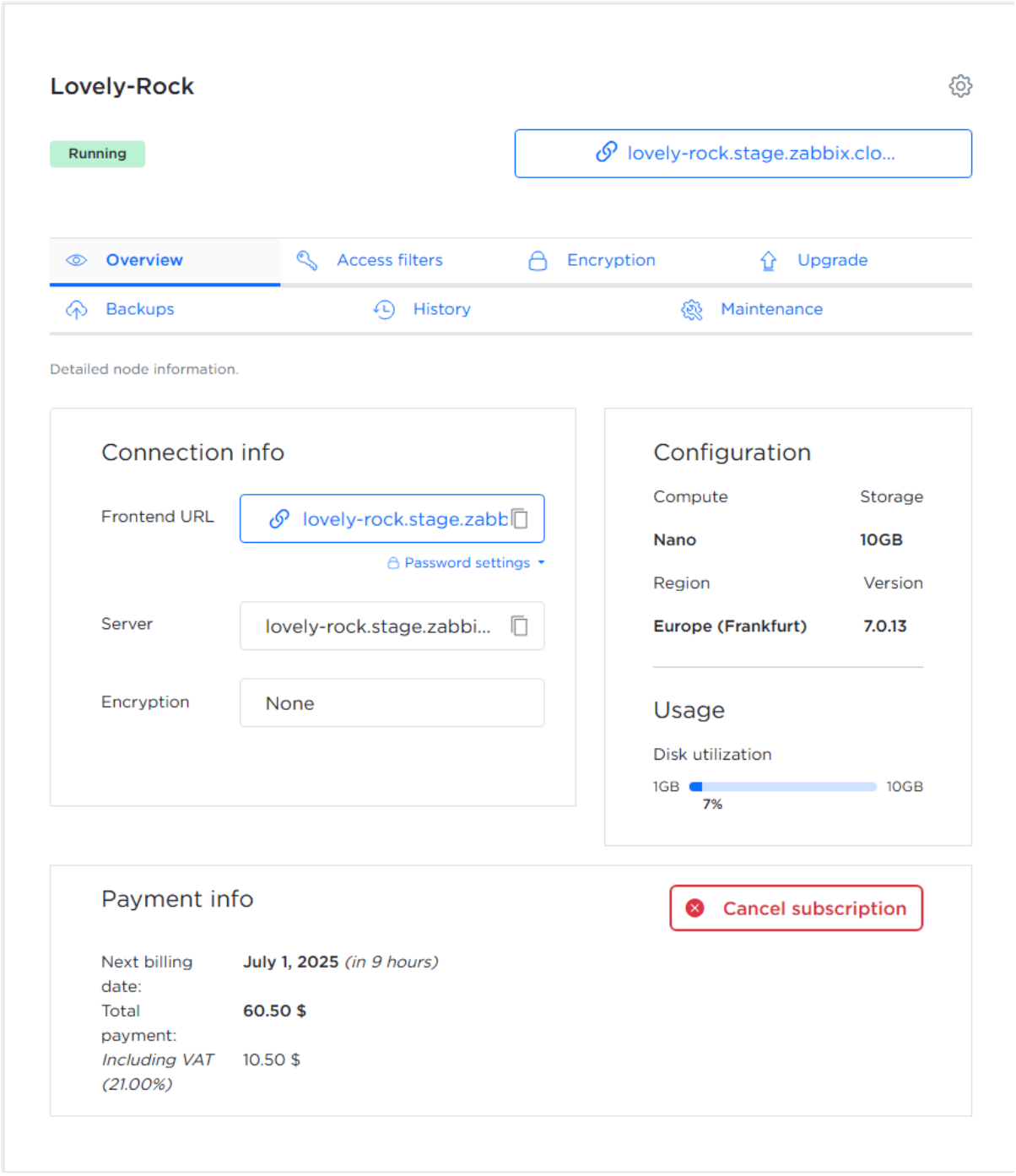
**Облачный узел** — это эквивалент экземпляра Zabbix, размещенный в облаке. Он включает сервер Zabbix, базу данных и веб-интерфейс. Облачный узел имеет выделенные вычислительные ресурсы (CPU, память) и дисковое хранилище.

Пользователи могут создавать один или несколько облачных узлов. Облачные узлы можно объединять в **организации** (структурные, региональные и т. д.), создавая распределенную систему мониторинга.

Облачные узлы и организации создаются через пользовательский интерфейс Zabbix Cloud. Чтобы открыть страницу конфигурации узла, щелкните по узлу в разделе *All organizations*, в меню вашей организации или по ссылке *Node configuration page*, которая отображается после запуска узла.

В представлении конфигурации узла каждая вкладка позволяет управлять определенной частью вашего узла:

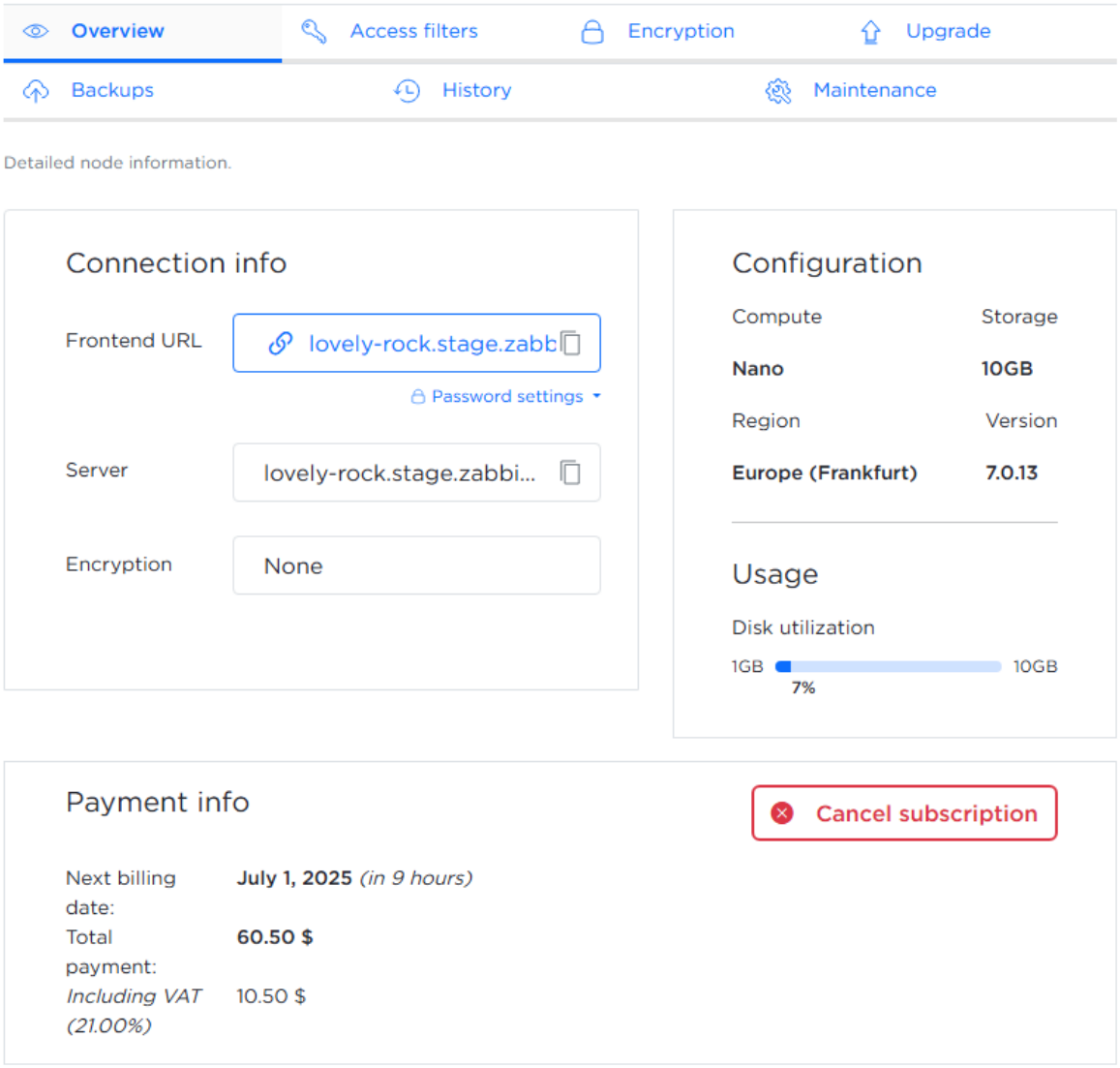
- Overview - просмотр сведений об узле, открытие его веб-интерфейса и управление пользователями узла.
- Access filters - управление доступом по IP-адресу или подсети.
- Encryption - настройка TLS путем загрузки сертификатов, ключей и необязательных CRL.
- Backups - создание, восстановление и удаление снимков узла; просмотр следующего запланированного резервного копирования.
- History - задание периодов хранения для журналов аудита, истории и трендов с предустановленными единицами и ограничением на обновление 3 раза в день.
- Maintenance - определение еженедельных окон обслуживания по часовому поясу, дню и часу.
- Upgrade - масштабирование вычислительных ресурсов/хранилища или управление статусом пробного периода/оплаты.




**Note:**
Некоторые вкладки могут быть недоступны в зависимости от типа пользователя, который просматривает страницу, и статуса узла (пробный/непробный).

**Вкладка Overview**

Вкладка *Overview* предоставляет подробную информацию об узле сети.




Она разделена на три раздела:

*Connection info*:

- URL веб-интерфейса, по которому можно перейти, с возможностью копирования в буфер обмена
- Настройки пароля
- Текущий метод шифрования

*Configuration and usage*:

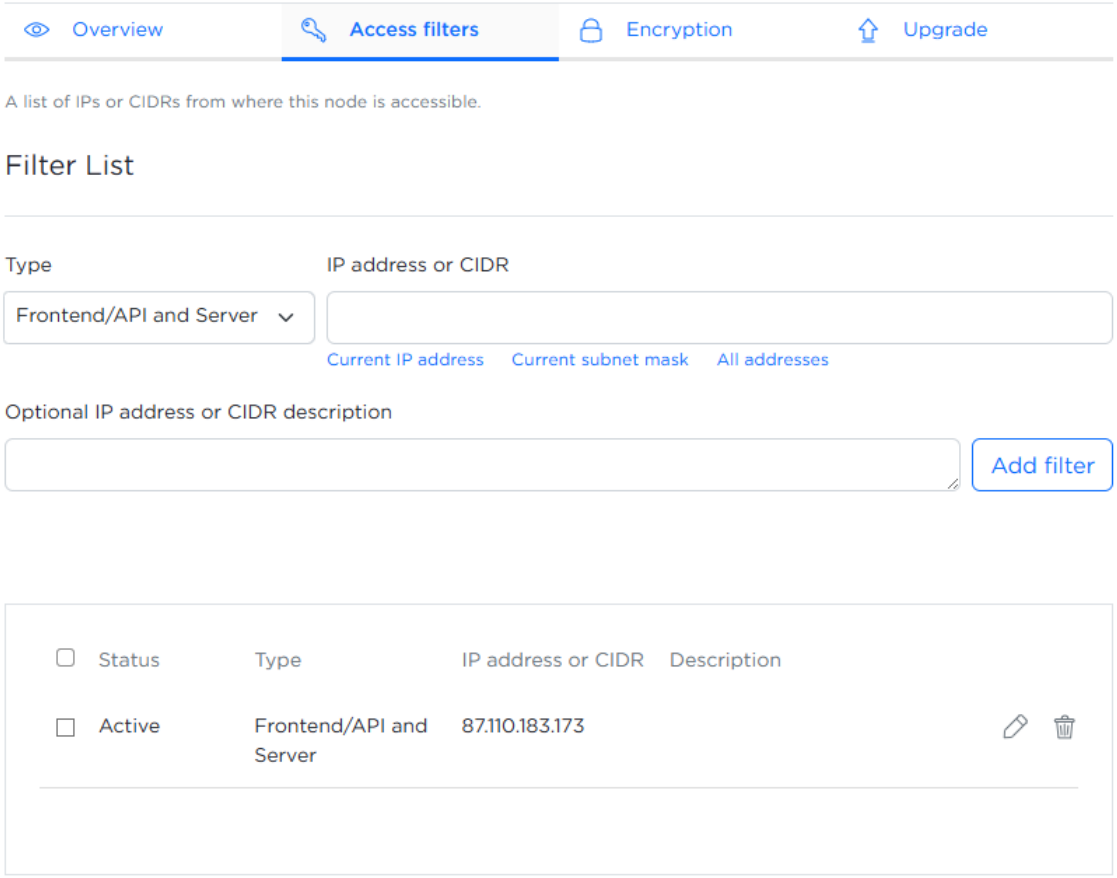
- Уровень вычислительных ресурсов, максимальный объем хранилища, регион
- Версия компонентов Zabbix
- Использование диска

*Node info/Payment Info*:

- Дата окончания срока действия (для пробных и предоплаченных узлов)
- В зависимости от типа узла после завершения инициализации отображается контекстно-зависимая кнопка: - *Upgrade to paid* - доступно для узлов на бесплатном пробном периоде - *Cancel subscription* - доступно для платных узлов с активной подпиской - *Resubscribe to paid* - отображается, если подписка была отменена - Для предоплаченных узлов кнопка действия не отображается, так как они имеют длительный срок действия и не могут быть отменены

**Вкладка фильтров доступа**

Вкладка *Фильтры доступа* позволяет ограничить доступ к вашему узлу сети Zabbix по IP-адресу или подсети.




Чтобы настроить фильтр доступа, заполните следующие поля:

- *Тип* - используйте раскрывающийся список, чтобы выбрать, какой компонент Zabbix ограничить: только веб-интерфейс/API, только сервер или оба.
- *IP-адрес или CIDR* - введите IP-адрес или CIDR вручную либо заполните поле автоматически, выбрав соответствующий вариант.
- *Необязательное описание IP-адреса или CIDR* - добавьте необязательное описание для каждого IP-адреса или подсети.

После заполнения полей нажмите *Добавить фильтр*. Добавленные фильтры отображаются в таблице ниже, где показаны статус, тип, IP-адрес или подсеть и описание.

**Warning:**
Во время инициализации узла сети фильтр доступа автоматически создается для IP-адреса, который вы использовали. Если позже вы подключитесь с другого IP-адреса или из другой сети, вам нужно добавить этот адрес здесь, иначе доступ будет заблокирован.

Вы можете:

- - изменить фильтр.
- - удалить фильтр или несколько фильтров с помощью флажков.

**Вкладка Encryption**

Вкладка *Encryption* позволяет настроить TLS шифрование между компонентами Zabbix путем загрузки необходимых файлов сертификатов.

Чтобы включить защищенную связь с использованием сертификатов, выберите *CERTIFICATE* и загрузите со своего компьютера следующие файлы:

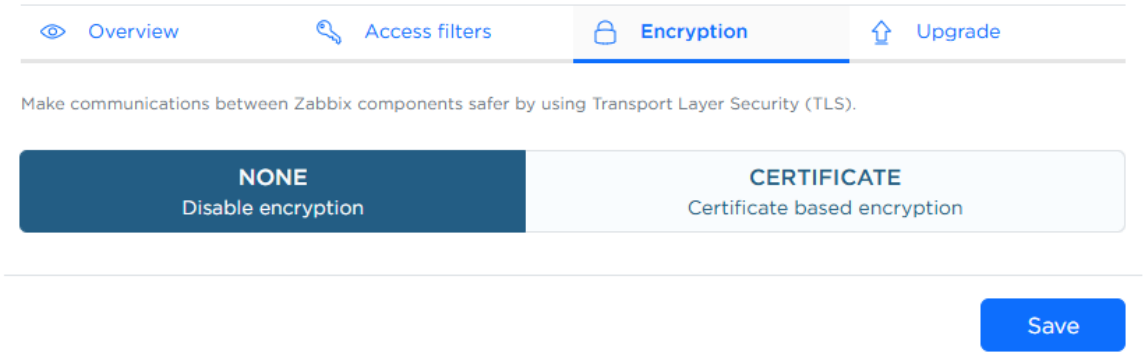
- *Root CA certificates* - используются для проверки подлинности сертификатов другой стороны.
  Нажмите *Choose file* и выберите файл, содержащий доверенные сертификаты Root CA (обычно файл `.crt` или `.pem`, в кодировке PEM).

- *Certificate chain* - представляет цепочку доверия для вашего сертификата.
  Нажмите *Choose file* и выберите файл в кодировке PEM (например, `.crt` или `.pem`), который включает ваш сертификат и любые промежуточные сертификаты CA.

- *Private key* - закрытый ключ, соответствующий вашему сертификату.
  Нажмите *Choose file* и выберите файл `.key` в кодировке PEM.

- *Revoked certificates* (необязательно) - если вы ведете список отозванных сертификатов (CRL), нажмите *Choose file* и выберите файл CRL, чтобы указать сертификаты, которым больше не следует доверять.

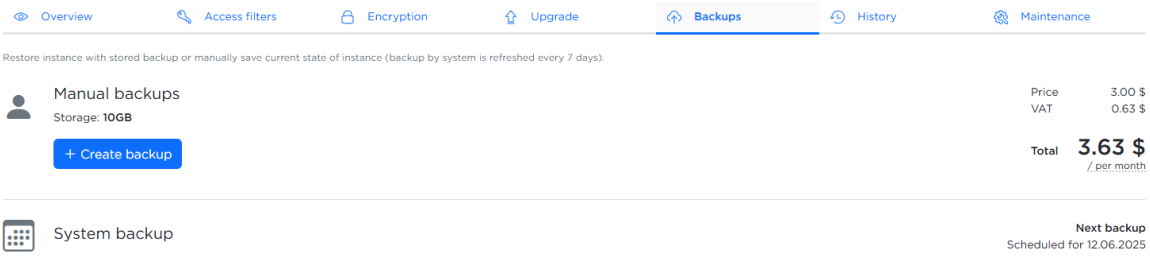
После выбора файлов нажмите *Save*. Файлы будут загружены, проверены и применены для настройки TLS-шифрования между компонентами Zabbix.




**Вкладка Backups**

Вкладка *Backups* позволяет просматривать, создавать, восстанавливать и удалять резервные копии вашего узла сети Zabbix.

Вы можете восстановить свой экземпляр из любой сохраненной резервной копии или в любое время создать новую резервную копию вручную.
На вкладке также отображается дата следующего запланированного резервного копирования.



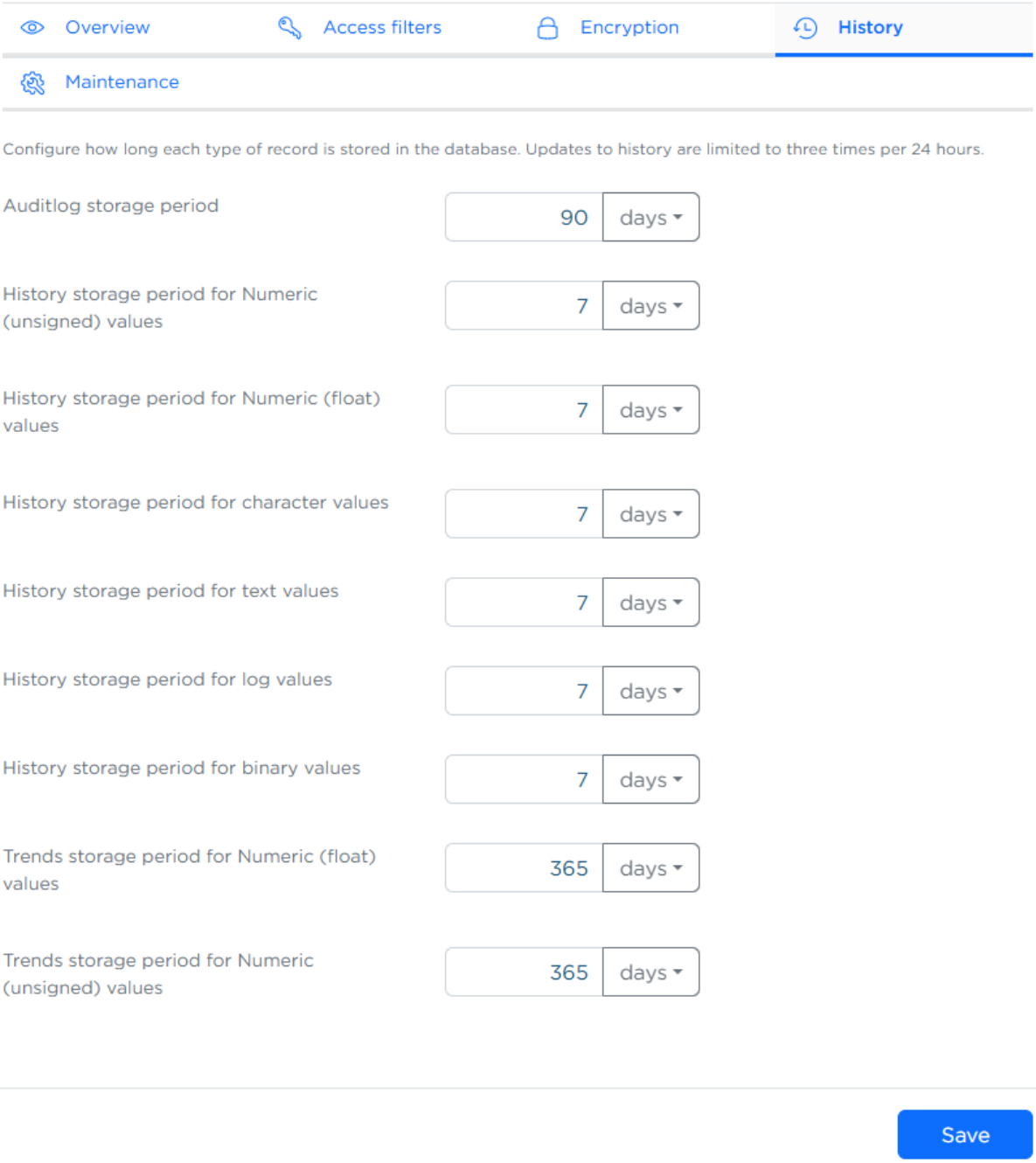

Чтобы создать резервную копию:

1. Нажмите кнопку *Create backup*.
2. Введите необязательное описание во всплывающем окне.
3. Нажмите *Create* для подтверждения.

**Warning:**
За дополнительные резервные копии ежемесячно взимается плата в зависимости от объема используемого хранилища.

**Вкладка History**

Вкладка *History* позволяет настроить, как долго хранятся различные типы исторических данных.



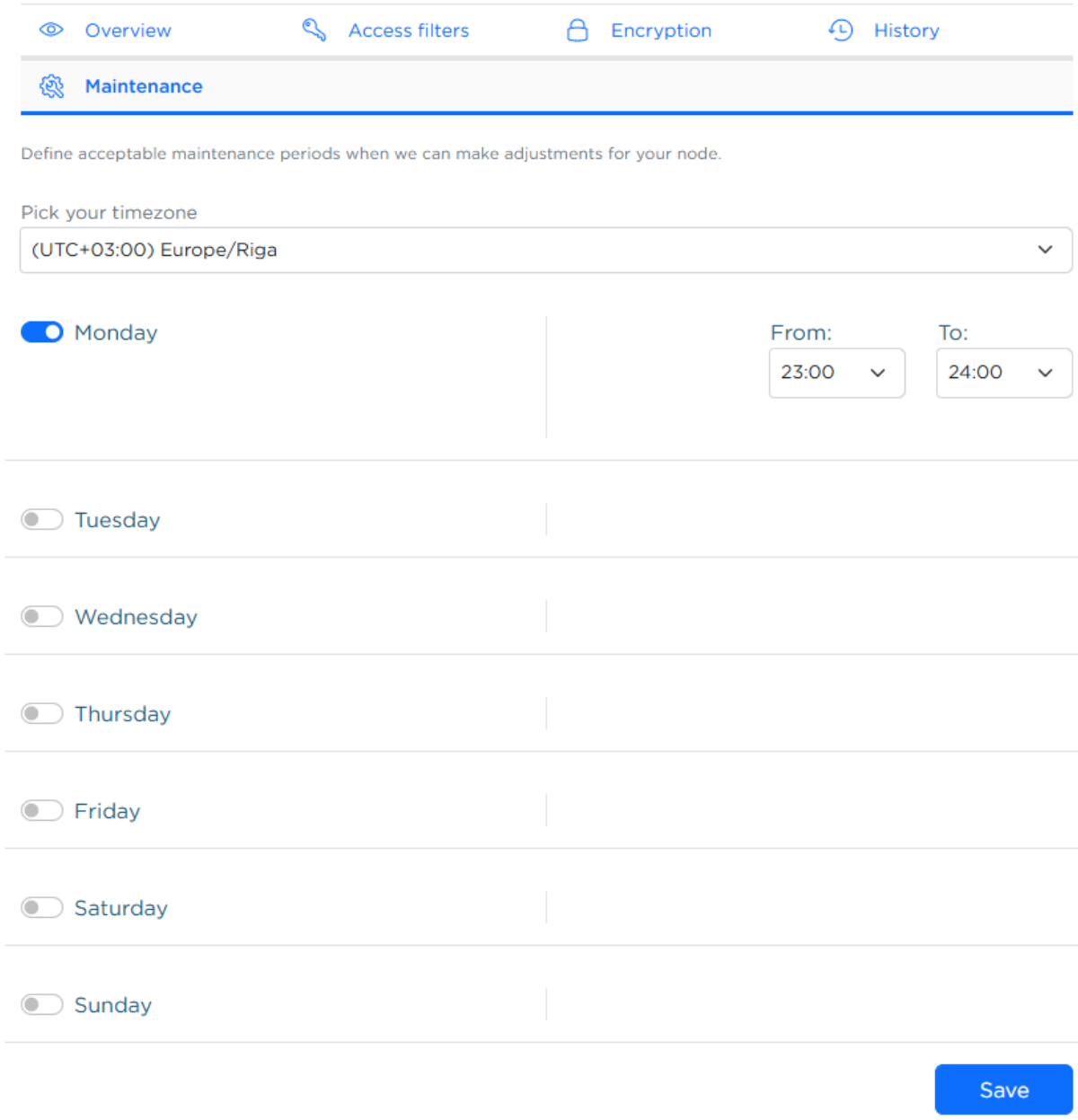

Вы можете задать период хранения для каждого типа данных (например, числовых, текстовых, журналов) и выбрать единицу измерения (дни, месяцы, годы). Изменения можно вносить до трех раз каждые 24 часа.

После обновления значений нажмите *Save*, чтобы применить изменения.

**Warning:**
Более длительные периоды хранения могут увеличить использование дискового пространства и повлиять на производительность. Рекомендуется настраивать периоды хранения с учетом ваших реальных данных и потребностей в анализе.<br> Zabbix Cloud использует разбиение базы данных на разделы для управления хранением данных вместо традиционной очистки данных Zabbix. Это обеспечивает более высокую производительность и более предсказуемую обработку сроков хранения.

**Вкладка Maintenance**

Вкладка *Maintenance* позволяет задать допустимые окна обслуживания, в которые на вашем узле сети можно выполнять обновления или изменения.

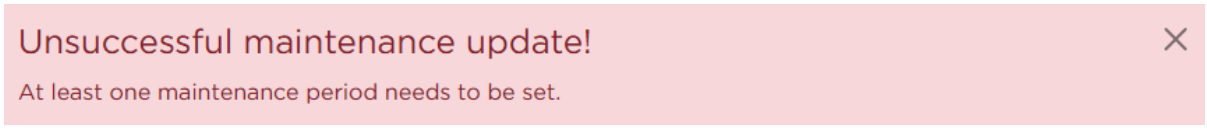



Необходимо:

- Выбрать свой часовой пояс в раскрывающемся списке.
- Выбрать как минимум один день и одно почасовое окно для обслуживания.

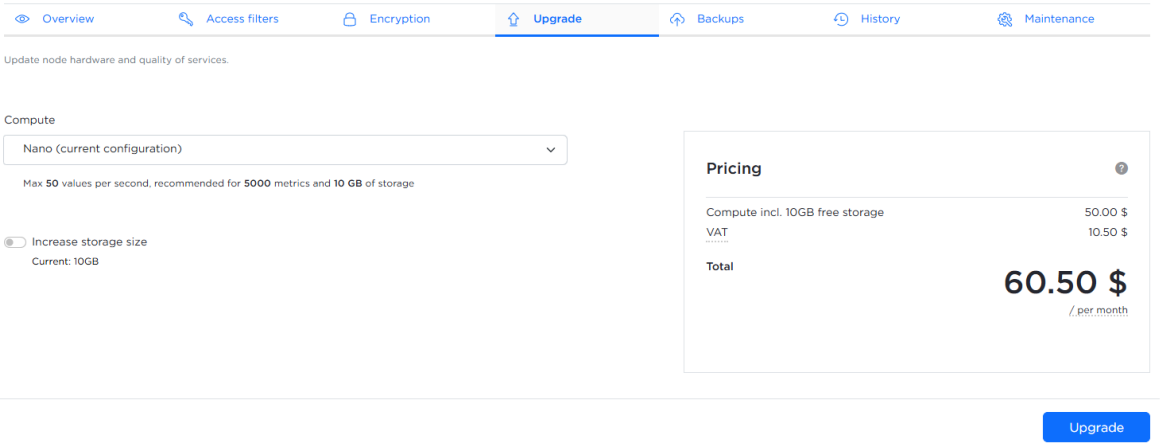
Нажмите *Save*, чтобы применить изменения.

Если не выбран ни день, ни временное окно, будет отображено сообщение об ошибке.




**Вкладка Upgrade**

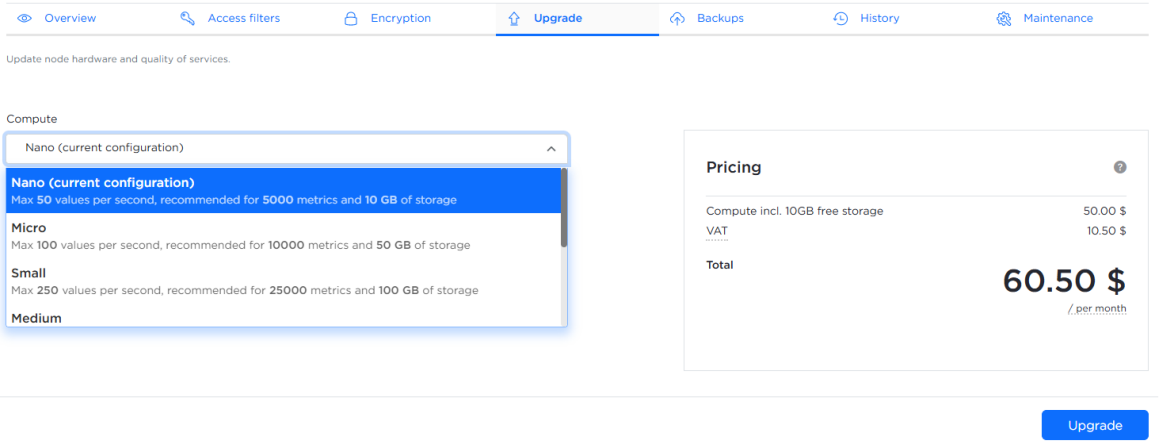
Вкладка *Upgrade* позволяет масштабировать ресурсы вашего узла или управлять его подпиской.




Вы можете увеличить уровень вычислительных ресурсов или размер диска — обратите внимание, что размер диска можно увеличивать только один раз в 24 часа, а уменьшить его через веб-интерфейс нельзя (для любых запросов на уменьшение обратитесь в службу поддержки Zabbix).
Тестовые узлы можно перевести на платную подписку.

Чтобы увеличить уровень вычислительных ресурсов:

1. Нажмите раскрывающийся список *Compute*.



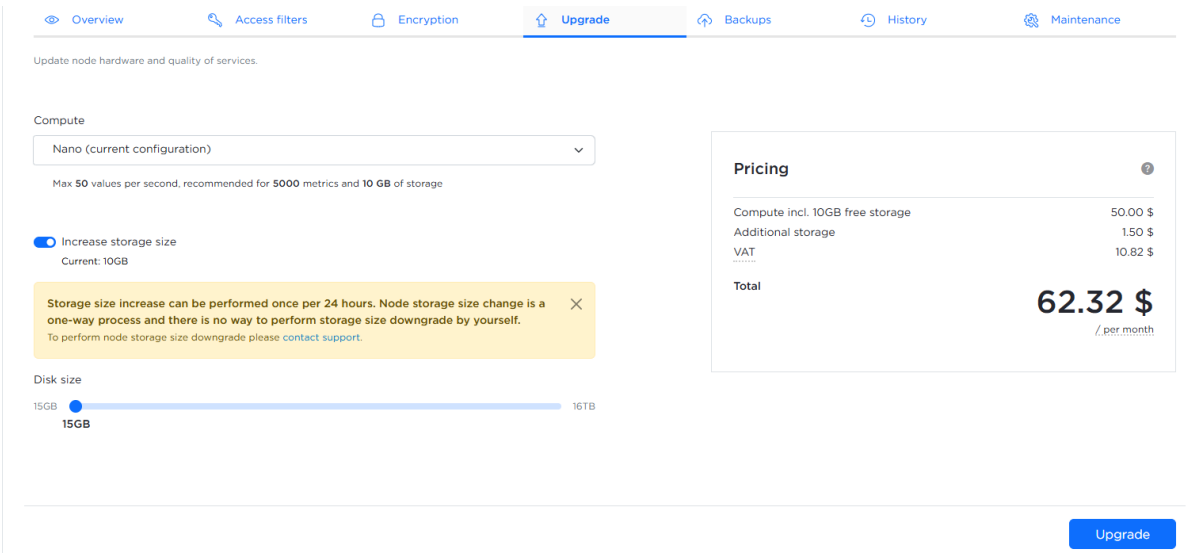

2. Выберите нужный уровень конфигурации.

3. Нажмите кнопку *Upgrade*.

Чтобы увеличить размер диска:

1. Выберите радиокнопку *Increase storage size*.

2. Переместите ползунок *Disk size* до нужного значения.



3. Нажмите кнопку *Upgrade*.

**Note:**
Обновленная общая ежемесячная цена отображается в таблице *Pricing* перед подтверждением изменений.

## Добавление пользователей

### Введение

Следующие примеры описывают, как добавить пользователей в вашу организацию и узлы сети.

**Note:**
Копирование, очистка и сброс пароля веб-интерфейса могут выполняться только владельцем.

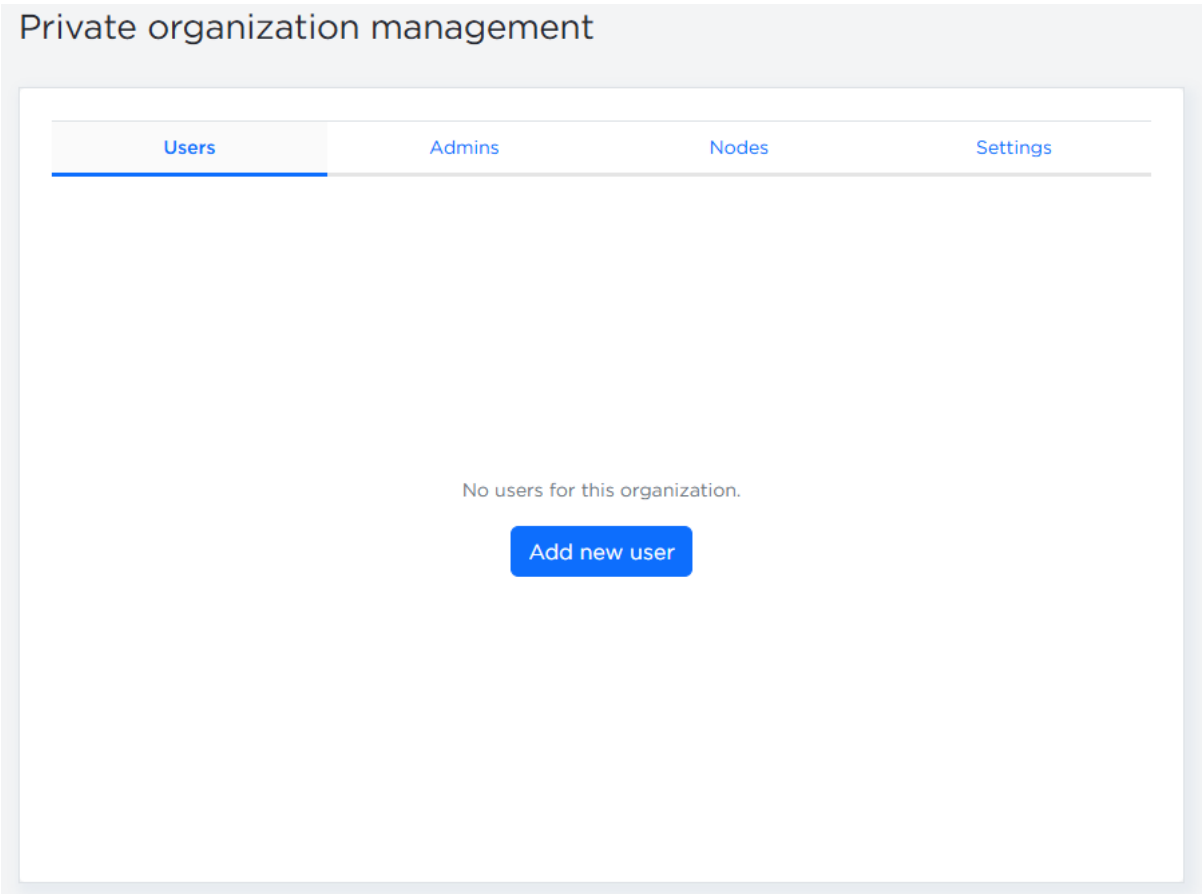
### Добавление пользователей в организации

Подобно пользователям в Zabbix, в Zabbix Cloud можно создавать пользователей и назначать им роли, определяющие их права в организации и её нодам.

Чтобы добавить пользователей в организацию:

1. Откройте страницу *Все организации*.

2. Нажмите кнопку настроек нужной организации, чтобы открыть меню управления.




3. На вкладке *Users* или *Admins* нажмите *Add new user*.

4. Введите адрес электронной почты пользователя и выберите роль: *Administrator* или *User*.

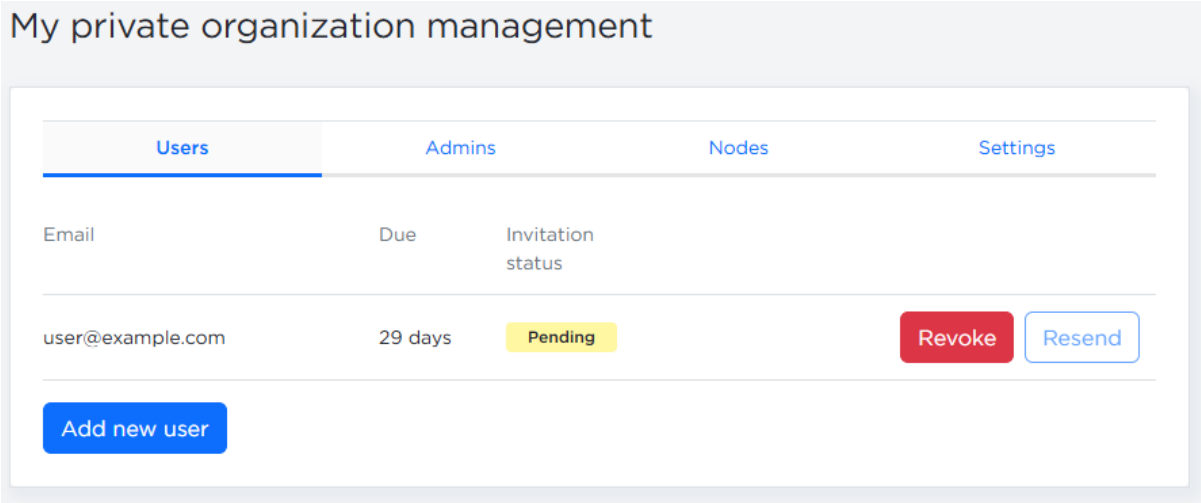
Обратите внимание: выбор роли *User* добавляет пользователя в организацию, но не предоставляет доступ ни к каким нодам.

Каждая роль имеет свой уровень доступа:

- **User** - может работать только с назначенными ему нодами и выполнять конфигурационные задачи, не влияющие на оплату, например:
  - Управление фильтрами доступа
  - Настройка шифрования
  - Задание сроков хранения истории и трендов
  - Планирование технического обслуживания
- **Administrator** - имеет все права пользователя, а также может выполнять действия, влияющие на оплату, например:
  - Создавать и восстанавливать резервные копии
  - Изменять уровень ресурсов ноды (вычислительный уровень и размер диска)
  - Отменять подписки
- **Owner** - полный контроль над организацией. Помимо всех прав администратора, может:
  - Создавать и удалять ноды
  - Добавлять или удалять пользователей и администраторов
  - Переименовывать или удалять организацию
  - Управлять платёжными данными

> **Attention:**
> Платёжный метод владельца используется для всех услуг. Если требуется подтверждение операции банком, именно владелец должен её авторизовать, даже если действие инициировал администратор.

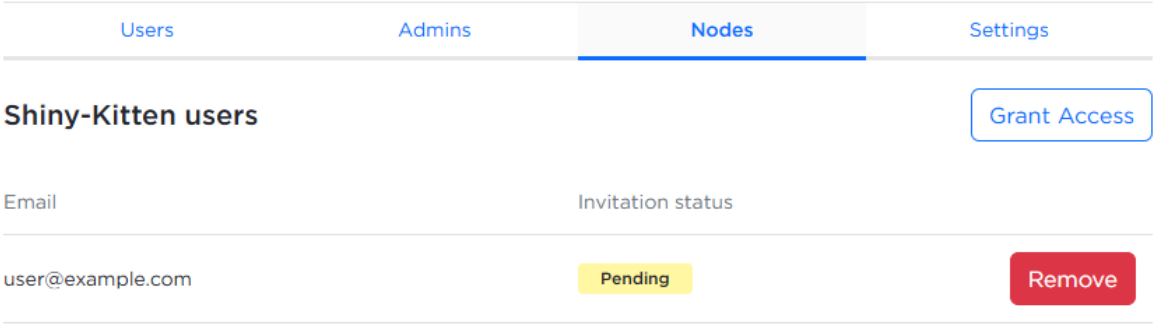
Приглашённый пользователь будет добавлен со статусом *Pending*. Приглашение можно отозвать или отправить повторно (повторная отправка возможна не чаще одного раза в 10 минут — кнопка будет неактивна при превышении лимита).




**Добавление пользователей к нодам**

Чтобы добавить пользователя к ноде:

1. Перейдите на вкладку *Ноды*.
2. Нажмите *Предоставить доступ*.
3. Выберите пользователя, добавленного в организацию из раскрывающегося списка.
4. Нажмите *Предоставить доступ*.




Кроме того, вы можете предоставить доступ в окне *Конфигурация ноды*:

1. Нажмите кнопку настроек .
2. Выберите *Добавить пользователя*.
3. Выберите пользователя и нажмите *Предоставить доступ*.

Приглашенные пользователи должны принять приглашение, создав учетную запись Zabbix Cloud, используя тот же адрес электронной почты, на который было отправлено приглашение. Если у пользователя уже есть учетная запись Zabbix Cloud с этим адресом электронной почты, он должен войти в систему и подтвердить приглашение. Приглашения привязаны к определенному адресу электронной почты и не могут быть приняты с другой учетной записи.

## Основные отличия Zabbix Cloud

**Введение**

Zabbix Cloud — это облачная платформа, позволяющая пользователям запускать автономные экземпляры Zabbix в облаке (облачные узлы).

По сравнению с обычным локальным развертыванием Zabbix:

- Не требуется установка или обслуживание.
- Облачные узлы полностью управляются Zabbix.
- Облачные узлы можно запустить за несколько минут.

Эта услуга предоставляется по модели оплаты по мере использования (доступна **бесплатная пробная версия**).

**Сравнительная таблица** Хотя большая часть функциональности совпадает между локальными установками Zabbix и облачными узлами, существуют и различия. Ниже приведена сводная сравнительная таблица. В последующих разделах отдельные пункты описаны подробнее.

| Feature / Aspect | Zabbix Cloud | Zabbix on-premises |
|---|---|---|
| *Release cycle* | В соответствии с LTS релизами | Вы выбираете версию Zabbix (stable, beta, custom builds) |
| *Ease of use* | Готовая к использованию платформа, полностью управляемая Zabbix | Требует установки, настройки и обслуживания |
| *Vulnerability patching* | Автоматическое обновление ОС/Zabbix | Ручное обновление пользователем |
| *HTTPS certificate* | Действительный сертификат из коробки (без Let's Encrypt) | По умолчанию самоподписанный (можно настроить Let's Encrypt или другой CA) |
| *DB performance* (INSERT/UPDATE/SELECT) | Автоматическое разбиение на разделы для максимальной скорости; без длительных DELETE | Ручное разбиение на разделы и очистка (возможны длительные транзакции DELETE) |
| *Performance tuning* | Предварительно настроено командой Zabbix Cloud (без changelogs) | Полная видимость и контроль над параметрами настройки |
| *Firewall control* | Управляется командой Zabbix Cloud; порты GUI/trapper не доступны публично | Полный контроль над правилами firewall и открытием портов |
| *DB isolation & protection* | Высокая изоляция, прямой доступ к БД/SSH отсутствует | Зависит от вашей сети и безопасности узла сети |
| *High availability (HA)* | Работает на высокодоступной облачной платформе,<br>прокси Zabbix можно использовать для повышения устойчивости сбора данных | HA под управлением пользователя |
| *SNMP traps* | Только через выделенный прокси (без HA/балансировки нагрузки для приемника traps) | Напрямую или через прокси, возможны HA/балансировка нагрузки |
| *SNMP polling* | Сложно без прокси (для каждого устройства нужен NAT/пользовательский порт) | Нативный SNMP polling; прокси необязателен |
| *Retention settings* | History/Trends/Audit только через веб-интерфейс (без API или файла конфигурации) | Настраивается в файле конфигурации сервера Zabbix или через API |
| *Custom scripts* | `AlertScriptsPath`, `ExternalScripts`, веб-интерфейс и модули сообщества не поддерживаются | Полностью поддерживается (scripts path, modules, интеграции) |

| Feature / Aspect | Zabbix Cloud | Zabbix on-premises |
|---|---|---|
| *ODBC monitoring* | Только PostgreSQL (драйвер `{postgresql}`); плагин MariaDB присутствует как заглушка | ODBC для PostgreSQL, MySQL, Oracle и т. д. - полностью настраивается |
| *Limiting ODBC calls* | Невозможно (`StartODBCPollers=1` только; тяжелые синхронные запросы блокируют другую работу) | Полностью настраиваемое количество poller'ов и планирование |
| *SAML certificate upload* | Не поддерживается | Поддерживается через веб-интерфейс или API |
| *Scheduled reports* | Необходимо создать собственный тип media Email (script media не поддерживается) | Поддерживаются как script, так и Email media из коробки |
| *Host interface for active checks* | IP-адрес, выданный платформой (создается автоматически) | Интерфейсы управляются вами; IP-адреса находятся под вашим контролем |

**Различия в функциональности** SNMP-трапы

SNMP-трапы поддерживаются только через выделенный прокси Zabbix. Если требуется мониторинг SNMP-трапов, автоматическое балансирование нагрузки или высокая доступность для прокси использовать нельзя, поскольку SNMP-трапы должны отправляться на один IP-адрес.

SNMP-опрос

Без прокси SNMP-опрос требует, чтобы каждое устройство было доступно через NAT и использовало нестандартные порты.

Настройки хранения

Периоды хранения истории, трендов и журнала аудита можно настраивать только через веб-интерфейс в облаке. Эти параметры нельзя настроить через `zabbix_server.conf` или API.

Ручные переопределения истории для отдельных элементов данных не поддерживаются (разбиение на разделы управляется глобально). URL-адреса облачных узлов не принимают те же параметры запроса, что и в локальных установках.

Пользовательские скрипты

Следующие типы пользовательских скриптов не поддерживаются в Zabbix Cloud:

- Скрипты оповещений (`AlertScriptsPath`)
- Внешние скрипты (`ExternalScripts`)
- Скрипты веб-интерфейса на сервере Zabbix

Модули веб-интерфейса, разработанные сообществом, не могут быть установлены.

Мониторинг ODBC

Zabbix Cloud поддерживает мониторинг ODBC только для PostgreSQL. Используйте официальный шаблон ODBC Zabbix и задайте строку подключения в шаблоне следующим образом:

```
Driver={postgresql}
```

Также установлен плагин MariaDB, но в настоящее время он не функционирует. Если он используется, задайте:

```
Driver={mysql}
```

**Attention:**
В Zabbix Cloud нет известной рабочей конфигурации для Oracle MySQL 8.0. Хотя простые запросы (например, `SELECT 1`) могут выполняться успешно, более сложные запросы возвращают `SQL_ERROR`.<br><br> Ограничить вызовы ODBC невозможно (`StartODBCPollers=1` только). Интенсивная синхронная отчетность может повлиять на производительность, и одновременно может выполняться только один `SELECT`.

Доступ к инфраструктуре

Zabbix Cloud не предоставляет SSH-доступ к базовым узлам и не допускает прямых подключений к базе данных (например, к порту 3306). Все действия по настройке, мониторингу и устранению неполадок должны выполняться через облачный веб-интерфейс или API, что обеспечивает изоляцию и защиту уровней ОС и базы данных.

Интерфейс активной проверки

Активные проверки в облаке автоматически создают интерфейс узла сети с IP-адресом, не связанным с вашей локальной сетью. По умолчанию этот IP-адрес может быть внешним. Чтобы сохранить согласованность, вы можете вручную удалить или изменить интерфейс узла сети после создания.

Загрузка сертификата SAML

SAML-аутентификация не поддерживается, поскольку лишь немногие поставщики принимают незашифрованные или неподписанные сообщения SAML.

Запланированные отчеты

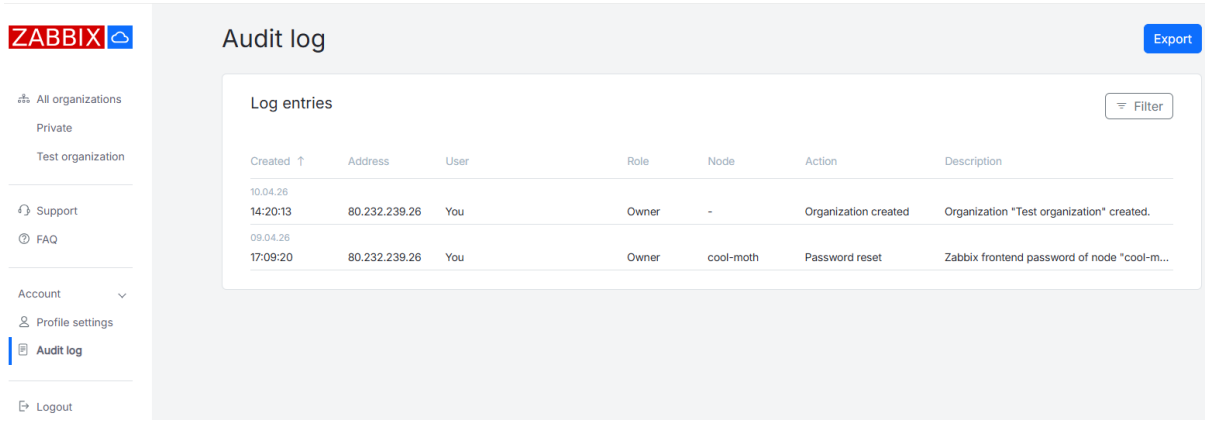
Медиа-тип *Cloud Email* по умолчанию использует транспорт script и не может использоваться для запланированных отчетов. Чтобы отправлять отчеты по электронной почте:

1. Создайте новый медиа-тип Email.
2. Назначьте этот медиа-тип электронной почты вашему пользователю в *User settings* > *Media*.
3. Выберите этот медиа-тип при настройке запланированных отчетов.

## Журнал аудита

**Обзор** В Zabbix Cloud журнал аудита записывает действия пользователей и системы.

После входа в систему в учетную запись Zabbix Cloud перейдите в *Account* > *Audit log*, чтобы просмотреть записи журнала аудита. Доступные пользователю записи журнала аудита зависят от роли пользователя в организации.




**Просмотр записей журнала аудита** Записи журнала аудита отображаются в таблице.

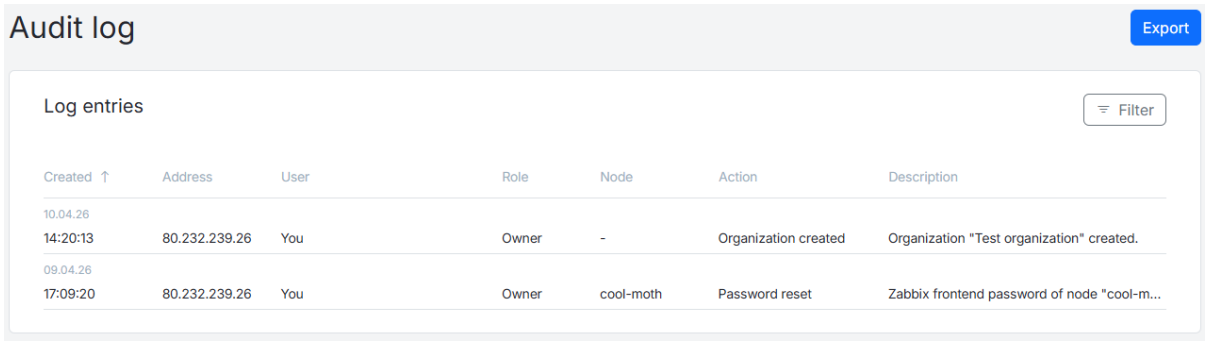



Таблица содержит следующую информацию:

| Column | Description |
|---|---|
| *Created* | Дата и время создания записи журнала. Отображаемое время соответствует часовому поясу пользователя, выбранному в раскрывающемся списке *User timezone* в *Account* > *Profile settings*. |
| *User* | Имя пользователя организации, который выполнил действие. |
| *Role* | Роль пользователя, который выполнил действие. |
| *Node* | Имя узла, связанного с действием. |
| *Action* | Имя выполненного действия. |
| *Description* | Описание выполненного действия. |

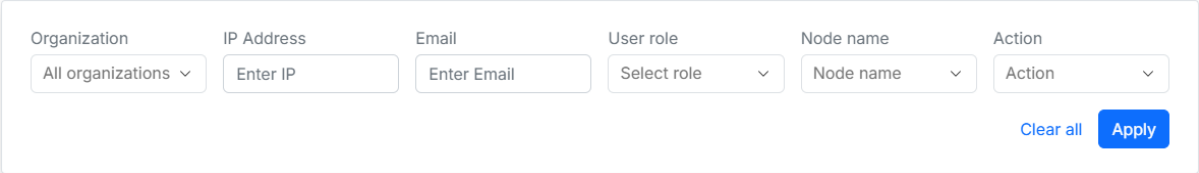
По именам столбцов можно щелкнуть, чтобы отсортировать записи по возрастанию или убыванию.

Отфильтрованные записи журнала аудита можно экспортировать в формате `.csv` с помощью кнопки *Export*.

Использование фильтров

Используйте кнопку *Filter*, чтобы открыть фильтр.

Чтобы отобразить отфильтрованные результаты, нажмите *Apply* после выбора одного или нескольких параметров фильтрации. Чтобы очистить фильтр, нажмите *Clear all*.




Можно фильтровать записи по одному или нескольким параметрам. Доступны следующие параметры фильтрации:

| Parameter | Description |
|---|---|
| *Organization* | Выберите все организации или конкретную. |
| *IP Address* | Укажите IP-адрес. |
| *Email* | Укажите адрес электронной почты, привязанный к пользователю. |
| *User role* | Выберите все роли или конкретную: Owner, Admin или User. |
| *Node name* | Выберите все узлы сети или конкретный. |
| *Action* | Выберите конкретный тип действия. |

Журнал аудита отслеживает следующие действия:

- *Node created*
- *Node deleted*
- *Node upgraded*
- *Node version updated*
- *Password reset*
- *Maintenance updated*
- *History setting updated*
- *Node expiration date extended*
- *Backup created*
- *Backup deleted*
- *Backup restored*
- *Backup updated*
- *Access filter added*
- *Access filter updated*
- *Access filter deleted*
- *Encryption enabled*
- *Encryption disabled*
- *Subscription cancelled*
- *Subscription renewed*
- *Subscription payment failed*
- *Payment requires confirmation*
- *Organization created*
- *Organization deleted*
- *Admin invited*
- *User invited*
- *Admin joined*
- *User joined*
- *Admin removed from organization*
- *User removed from organization*
- *Node added to organization*
- *Node removed from organization*
- *Node access granted*
- *Node access revoked*
- *User settings updated*
- *Payment created*
- *Payment completed*
- *Payment failed*

- *VAT charge applied*
- *VAT validation failed*
- *Billing info updated*
- *Payment method updated*
- *Payment method removed*
- *Prepaid node created*
- *Prepaid node updated*

# Центр разработчика

# Центр разработчика

Начните разрабатывать собственные расширения для Zabbix с помощью руководств по плагинам, модулям и виджетам.

## Сборка на Zabbix

**Плагины** Обзор того, как разрабатывать и управлять плагинами Zabbix для расширения функциональности или интеграции с внешними системами.

**Библиотека Python для Zabbix** Ресурсы для использования официальной библиотеки `zabbix_utils` для взаимодействия с API Zabbix и создания интеграций.

**Модули веб-интерфейса** Руководства и справочные материалы по созданию пользовательских модулей, которые расширяют или изменяют веб-интерфейс Zabbix в соответствии с конкретными сценариями использования.

**Виджеты** Разбор структуры и логики виджетов с инструкциями по созданию пользовательских элементов панели мониторинга, адаптированных под ваши потребности.

## Модули

#### Что такое модуль PHP веб-интерфейса?

- Модуль - это объект с уникальным идентификатором, именем, описанием, автором и другими полями, определенными в его файле манифеста, а также PHP, Javascript и другими файлами, расположенными в одном каталоге внутри подкаталога *modules* каталога установки веб-интерфейса Zabbix (например, *zabbix/ui/modules*).
- Модуль должен соответствовать простым правилам, чтобы гарантировать правильную работу.
- Модуль должен быть установлен (распакован) и включен в веб-интерфейсе администратором.

#### Для чего можно использовать модуль

- Добавление нового функционала через пользовательские разделы веб-интерфейса;
- Создание пользовательских типов виджетов панели (см. модули виджетов);
- Переопределение или расширение существующего функционала.

#### Для чего нельзя использовать модуль

- Регистрация нового метода API или изменение существующего.

#### Как работают модули

- Включенный модуль запускается при каждом HTTP-запросе перед выполнением кода действия.
- Модуль зарегистрирует новые действия или переопределит существующие.
- Модуль добавит новые разделы интерфейса и удалит или переопределит существующие.
- При необходимости модуль будет подключаться к событиям внешнего интерфейса, таким как onBeforeAction и on-Terminate.
- Запрошенное действие выполняется путем запуска кода действия - либо кода по умолчанию, либо определенного в модуле.

**Куда идти дальше** Независимо от того, предпочитаете ли вы учиться на практике или сначала прочитать рекомендации, эти страницы содержат информацию и шаги, необходимые для создания собственных модулей:

- Пошаговые руководства по написанию вашего первого модуля
- Файловая структура модуля
- Особенности модуля виджетов
- Примеры модулей для повторного использования

**Файловая структура модуля**

Весь код, связанный с модулем, хранится в одном каталоге внутри каталога **modules** вашей установки веб-интерфейса Zabbix (например, *zabbix/ui/modules*).

```
example_module_directory/          (      )
   manifest.json                   (      )                    .
   Module.php                                                        .
   actions/                                                    .
       SomethingView.php
       SomethingCreate.php
       SomethingDelete.php
       data_export/
           ExportAsXml.php
           ExportAsExcel.php
   views/                                                 .
       example.something.view.php
       example.something.delete.php
   assets/                                                      ,                    .                 n
       js/                                        JavaScript,                .
           example.something.view.js.php
       css/                                    CSS-   ,                .
           example.something.css
        image.png                                      ,                .
        example.something.file                                            .
```

**Дерево файлов модуля**

**Написание модуля** Процесс написания примера модуля состоит из следующих шагов (если доступно, нажмите на имя файла или папки, чтобы просмотреть дополнительные сведения о шаге):

1. Создайте новый каталог для модуля внутри **zabbix/ui/modules/**.
2. Добавьте файл manifest.json с метаданными модуля.
3. Создайте папку *views* и определите представление(я) модуля.
4. Создайте папку *actions* и определите действие(я) модуля.
5. Создайте файл Module.php (или Widget.php для виджетов панели) и определите правила инициализации и обработки событий.
6. Создайте папку *assets* для файлов JavaScript (поместите их в *assets/js*), CSS-стилей (поместите их в *assets/css*) или любых других дополнительных файлов.
7. Убедитесь, что в manifest.json указаны необходимые представления, действия и файлы ресурсов.
8. Зарегистрируйте модуль во веб-интерфейсе Zabbix и начните его использовать.

**Note:**
Вы можете зарегистрировать и включить модуль сразу после создания файла manifest.json. После включения модуля вы сможете сразу просматривать все изменения, внесенные в файлы модуля, просто обновив веб-интерфейс Zabbix.

**manifest.json**

Любому модулю нужен файл manifest.json. Файл должен находиться в основном каталоге модуля (например, *zabbix/ui/modules/module_name/manifest.json*).

Минимально необходимый набор полей manifest.json:

```
{
    "manifest_version": 2.0,
    "id": "my_ip_address",
    "name": "My IP Address",
    "namespace": "MyIPAddress",
    "version": "1.0"
}
```

Параметры, поддерживаемые в **manifest.json** (щелкните имя параметра, чтобы открыть подробное описание):

| Parameter | Description | Required |
|---|---|---|
| manifest_version | Версия манифеста модуля. | Yes |
| id | Уникальный ID модуля. | |
| name | Имя модуля, которое будет отображаться в разделе администрирования. | |
| namespace | Пространство имен PHP для классов модуля. | |
| version | Версия модуля. | |
| type | Тип модуля. Для виджета должно быть установлено значение *widget* | Yes for widgets, otherwise no |
| widget | Конфигурация виджета. Используется только для виджетов. | |
| actions | Действия, которые нужно зарегистрировать вместе с модулем. | |
| assets | CSS-стили и файлы JavaScript для включения. | No |
| author | Автор модуля. | |
| config | Значения по умолчанию для пользовательских параметров модуля. | |
| description | Описание модуля. | |
| url | Ссылка на описание модуля. | |

manifest_version

Версия манифеста модуля. На данный момент поддерживается версия **2.0**. *Тип*: Double

*Пример*:

```
"manifest_version": 2.0
```

id

Идентификатор модуля. Должен быть уникальным. Чтобы избежать конфликта имён в будущем, рекомендуется использовать для модулей префикс (имя автора или название компании, или любой другой). Например, если модуль является примером для уроков и имя модуля «My module», то идентификатор будет «example_my_module».

*Тип*: String

*Пример*:

```
"id": "example_my_module"
```

name

Имя модуля, которое будет отображаться в секторе Администрирование.

*Тип*: String

*Пример*

```
"name": "My module"
```

namespace

Пространство имён PHP для классов модулей

*Тип*: String

*Пример*:

```
"namespace": "ClockWidget"
```

version

Версия модуля. Версия будет отображаться в секторе Администрирование.

*Тип*: String

*Пример*:

```
"version": "1.0"
```

type

Тип модуля. Требуется для виджетов и должен быть равен ”widget”.

*Тип*: String

*По умолчанию*: ”module”

*Пример*:

```
"type": "widget"
```

actions

Действия для регистрации в модуле. Требуется определение ключа объекта *class* для каждого действия, остальные ключи действий не являются обязательными.

*Тип*: Object

Поддерживаемые ключи объекта, если *type* равно *module*:

- **write.your.action.name** (object) - название действия следует писать строчными буквами [a-z], разделяя слова точками. Поддерживает ключи:
  - **class** (string; обязательный) - имя класса действия.
  - **layout** (string) - макет действия. Поддерживаемые значения: *layout.json*, *layout.htmlpage* (по умолчанию), *null*.
  - **view** (string) - вид действия.

*Пример*:

```
"actions": {
    "module.example.list": {
        "class": "ExampleList",
        "view": "example.list",
        "layout": "layout.htmlpage"
        }
    }
```

Поддерживаемые ключи объекта, если *type* равно *widget*:

- **widget.{id}.view** (object) - имя файла и класса для просмотра виджета. Замените **{id}** значением id виджета (например, *widget.example_clock.view*). Поддерживает ключи:
  - **class** (string; обязательный) - имя класса действия для режима просмотра виджета для расширения класса CControllerDashboardWidgetView по умолчанию. Исходный файл класса должен находиться в каталоге *actions*.
  - **view** (string) - вид виджета. Должен находиться в каталоге *views*. Если файлом представления является *widget.view.php*, который ожидается по умолчанию, этот параметр можно опустить. Если вы используете другое имя, укажите его здесь.
- **widget.{id}.edit** (object) - имя файла для просмотра конфигурации виджета. Замените **{id}** значением id виджета (например, *widget.example_clock.edit*). Поддерживает ключи:
  - **class** (string; обязательный) - имя класса действия для режима просмотра конфигурации виджета. Исходный файл класса должен находиться в каталоге *actions*.
  - **view** (string) - вид конфигурации виджета. Должен находиться в каталоге *views*. Если файлом представления является *widget.edit.php*, который ожидается по умолчанию, этот параметр можно опустить. Если вы используете другое имя, укажите его здесь.

*Пример*:

```
"actions": {
    "widget.tophosts.view": {
        "class": "WidgetView"
    },
    "widget.tophosts.column.edit": {
        "class": "ColumnEdit",
        "view": "column.edit",
        "layout": "layout.json"
    }
}
```

assets

Стили CSS и файлы JavaScript для включения.

*Тип*: Object

*Поддерживаемые ключи объектов:*

- **css** (array) – файлы CSS для включения. Файлы должны находиться в папке *assets/css*.
- **js** (array) — файлы JavaScript для включения. Файлы должны находиться в папке *assets/js*.

*Пример*:

```
"assets": {
    "css": ["widget.css"],
    "js": ["class.widget.js"]
}
```

author

Автор модуля. Автор будет отображаться в разделе Администрирование.

*Тип*: String

*Пример*:

```
"author": "John Smith"
```

config

Значения по умолчанию для опций модуля. Объект может содержать любые пользовательские ключи. Если указано, эти значения будут записаны в базу данных во время регистрации модуля. Новые переменные, добавленные позже, будут записаны при первом вызове. В дальнейшем значения переменных можно будет изменить только непосредственно в базе данных.

*Тип*: Object

*Пример*:

```
"config": {
    "username": "Admin",
    "password": "",
    "auth_url": "https://example.com/auth"
}
```

description

Описание модуля.

*Тип*: String

*Пример*:

```
"description": "          ."
```

widget

Настройка виджета. Используется, если для *type* установлено значение *widget*.

*Тип*: Object

*Поддерживаемые ключи объекта:*

- **имя** (string) — используется в списке виджетов и в качестве заголовка по умолчанию. Если пусто, будет использоваться параметр «name» из модуля.
- **size** (object) – размеры виджета по умолчанию. Поддерживает ключи:
  - *width* (integer) — ширина виджета по умолчанию.
  - *height* (integer) — высота виджета по умолчанию.
- **form_class** (string) - класс с формой полей виджета. Должен находиться в каталоге *includes*. Если классом является *WidgetForm.php*, который ожидается по умолчанию, этот параметр можно опустить. Если вы используете другое имя, укажите его здесь.

- **js_class** (string) — имя класса JavaScript для режима просмотра виджета для расширения класса CWidget по умолчанию. Класс будет загружен с информационной панелью. Исходный файл класса должен находиться в каталоге assets/js. См. также: assets.
- **use_time_selector** (boolean) — определяет, требует ли виджет выбора времени на панели управления. Поддерживаемые значения: *true*, *false* (по умолчанию).
- **refresh_rate** (integer) — частота обновления виджета в секундах (по умолчанию: 60).

*Пример*:

```
"widget": {
    "name": "",
    "size": {
    "width": 12,
    "height": 5
    },
    "form_class": "WidgetForm",
    "js_class": "CWidget",
    "use_time_selector": false,
    "refresh_rate": 60
}
```

url

Ссылка на описание модуля. Для виджетов эта ссылка будет открываться при нажатии на значок справки ? в окне *Добавить виджет* или *Редактировать виджет*. Если **url** не указан, при нажатии на значок справки откроется общая страница Виджеты панели.

*Тип*: String

*Пример*:

```
"url": "http://example.com"
```

**Действия**

Действия отвечают за 'бизнес-логику' модуля. Действие обычно состоит из controller и action view.

Модуль может:

- Вызвать действия, которые уже определены в веб-интерфейсе Zabbix.
- Переопределить действия по умолчанию настраиваемыми действиями.
- Определить совершенно новые действия.

Чтобы переопределить поведение действия по умолчанию некоторым пользовательским поведением, определите действие с тем же именем в конфигурации модуля. При вызове действия вместо действия Zabbix по умолчанию будет выполнено действие модуля.

Файлы действий должны храниться в папке *actions*. Действия необходимо указать в файле manifest.json.

Контроллер

Рабочий процесс контроллера действия:

1. Проверьте, что все параметры, переданные в HTTP-запросе, допустимы:
   - Вызовите метод контроллера *checkInput()*;
   - Используйте правила проверки, определенные в CNewValidator.php;
   - Вызовите метод *validateInput()*.
2. Проверьте права пользователя.
3. Подготовьте данные в соответствии с переданными параметрами: если *checkInput()* возвращает true, Zabbix вызывает метод контроллера *doAction()*.
4. Подготовьте массив **$data** для представления. Используйте *CControllerResponseData* и метод *setResponse()* для сохранения ответа в массиве **$data**.

Пример:

```php
/**
 * 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:**
Полный список доступных классов контроллеров можно посмотреть в исходном коде Zabbix.

#### Представления

Файл представления получает данные от контроллера, а затем формирует HTML-вид.

**Note:**
Определение представления(й) для модуля веб-интерфейса является необязательным, если только модуль не является виджетом.<br> Виджеты панели мониторинга должны иметь как минимум два представления: одно для режима редактирования и одно для режима просмотра (их следует хранить в каталоге *views*).

В представлении можно использовать предопределенные HTML-классы Zabbix (из */zabbix/ui/include/classes/html*), а также добавлять новые HTML- и CSS-классы. Новые классы следует хранить в папке *assets* модуля.

Пример:

```
...
(new CColHeader(_('Name')))
```

Это добавит новый столбец с именем *Name* и оформит верхнюю строку таблицы так же, как на других страницах Zabbix.

Представление действия

Это справочный файл для определения представления действий.

```
<?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();
```

**Ресурсы**

Каталог *assets* может содержать любые файлы и подкаталоги, которые не принадлежат другим директориям. Вы можете использовать её для:

- JavaScript стилей (должны находиться внутри *assets/js*);
- CSS стилей (должны находиться внутри *assets/css*);
- Изображений;
- Шрифтов;
- Всего, что вам нужно добавить.

assets/js

Каталог *assets/js* зарезервирован и должен содержать только JavaScript-файлы. Чтобы использовать их в виджете, укажите эти файлы в *manifest.json*.

Например:

```
"assets": {
    "js": ["class.widget.js"]
}
```

assets/css

*assets/css* зарезервирован и должен содержать только файлы CSS стилей. Чтобы использовать их в виджете, укажите эти файлы в *manifest.json*.

Пример:

```
"assets": {
    "css": ["mywidget.css"]
}
```

CSS стили

Файлы CSS могут содержать настраиваемый атрибут «тема», определяющий другой стиль для определенных тем внешнего интерфейса.

Доступные темы и значения их атрибутов:

- **Синий** - [theme='blue-theme']
- **Темный** - [theme='dark-theme']
- **Высококонтрастный свет** - [theme='hc-light']
- **Высококонтрастный темный** - [theme='hc-dark']

Пример:

```
.widget {
    background-color: red;
}

[theme='dark-theme'] .widget {
    background-color: green;
}
```

#### Зарегистрируем новый модуль

В этом разделе объясняется, как добавить новый модуль во внешний интерфейс Zabbix.

**Предварительные требования** Прежде чем продолжить, убедитесь, что:

- Модуль расположен внутри каталога *modules* вашей установки внешнего интерфейса Zabbix (например, *zabbix/ui/modules*).
- Модуль имеет как минимум базовую версию файла manifest.json.
- У вас есть доступ к разделу меню «Администрирование» в Zabbix (требуется тип роли пользователя «Супер-администратор»).

> **Note:**
> Интерфейс не будет устанавливать или даже распознавать несовместимые модули.

**Добавление модуля** Откройте страницу *Administration→General→Modules* и нажмите *Scan directory*.

Найдите свой модуль в списке и активируйте его.

Чтобы активировать модуль, нажмите ссылку *Disabled* - состояние модуля изменится на *Enabled*.

Нажмите на имя модуля, чтобы просмотреть дополнительную информацию о модуле, такую как автор, версия или краткое описание (если оно определено в манифесте).

**Предварительный просмотр виджета** Модули виджетов после добавления сразу становятся видимыми в списке виджетов информационной панели.

Вы можете открыть дашборд, переключиться в режим редактирования и добавить виджет на дашборд как обычно.

При внесении изменений в виджет обновите панель мониторинга, чтобы увидеть, как виджет выглядит с учетом последних обновлений.

#### Виджеты

Виджеты - это внешние модули Zabbix, используемые для информационных панелей. Если не указано иное, все рекомендации по модулям также применимы к виджетам.

Однако виджет заметно отличается от модуля. Чтобы создать виджет:

- указать тип ”виджет” в файле manifest.json (”тип”: ”виджет”);
- включить как минимум два представления: одно для режима представления виджета и одно для режима конфигурации виджета (example.widget.view.php и example.widget.edit.php);
- и контроллер для представления виджета (WidgetView.php);
- используйте и расширяйте стандартные классы виджетов.

#### Конфигурация

На этой странице описаны классы, которые можно использовать для создания представления конфигурации виджета с настраиваемыми полями конфигурации. Представление конфигурации виджета - это часть виджета, которая позволяет пользователю настраивать параметры виджета для презентации.

Виджет

Основной класс виджетов, расширяет базовый класс всех виджетов информационной панели - *CWidget*. Требуется для переопределения поведения виджета по умолчанию.

Класс *Widget* должен находиться в корневом каталоге виджета (например, *zabbix/ui/modules/my_custom_widget*).

**Пример 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

Класс *WidgetForm* расширяет класс по умолчанию *CWidgetForm* и содержит набор полей *CWidgetField*, которые необходимы для определения структуры хранения конфигурации виджета в базе данных и обработки ввода.

Класс *WidgetForm* должен находиться в каталоге *includes*. Если класс имеет другое имя, это имя должно быть указано в параметре *widget/form_class* в файле *manifest.json*.

**пример 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 function addFields(): self {
        return $this
            ->addField(
                (new CWidgetFieldMultiSelectItem('itemid', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
                    ->setFilterParameter('numeric', true)
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            )
            ->addField(
```

```
                (new CWidgetFieldColor('chart_color', _('Color')))->setDefault('FF0000')
            );
    }
}
```

CWidgetFormView

Класс *CWidgetFormView* необходим для определения логики представления полей, определенных в классе *WidgetForm*, а также для задания их внешнего вида и поведения при отображении в представлении конфигурации.

Класс *CWidgetFormView* поддерживает следующие методы:

- *addField()* — получает в качестве параметра экземпляр класса *CWidgetFieldView*; каждый класс *CWidgetField* имеет соответствующий класс *CWidgetFieldView* для использования в представлении конфигурации виджета.
- *includeJsFile()* — позволяет добавить файл JavaScript в представление конфигурации виджета.
- *addJavaScript()* — позволяет добавить встроенный JavaScript, который будет выполнен сразу после загрузки представления конфигурации виджета.

Класс *CWidgetFormView* должен находиться в каталоге *views*.

**пример views/widget.edit.php**

```
<?php

/**
 * My custom widget form view.
 *
 * @var CView $this
 * @var array $data
 */

use Zabbix\Widgets\Fields\CWidgetFieldGraphDataSet;

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldMultiSelectItemView($data['fields']['itemid'], $data['captions']['items']['itemid'
    )
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description']),
        'js-advanced-configuration'
    )
    ->addField(
        new CWidgetFieldColorView($data['fields']['chart_color']),
        'js-advanced-configuration'
    )
    ->includeJsFile('widget.edit.js.php')
    ->addJavaScript('my_widget_form.init('.json_encode([
        'color_palette' => CWidgetFieldGraphDataSet::DEFAULT_COLOR_PALETTE
    ], JSON_THROW_ON_ERROR).');')
    ->show();
```

JavaScript

Класс JavaScript можно использовать для добавления динамического поведения и интерактивности в представление конфигурации виджета. Например, вы можете инициализировать палитру цветов, определенную в классе *CWidgetFormView*.

Класс JavaScript должен быть загружен вместе с формой, поэтому на него следует ссылаться в классе *CWidgetFormView* с помощью методов *includeJsFile()* и *addJavaScript()*.

В приведенном ниже примере экземпляр одноэлементного класса сразу создается и сохраняется под именем *window.my_custom_widget_form*. Таким образом, открытие формы во второй раз приведет к воссозданию экземпляра.

Класс JavaScript должен находиться в каталоге *views*.

**пример views/widget.edit.js.php**

```
<?php
```

```
use Modules\LessonGaugeChart\Widget;

?>

window.widget_lesson_gauge_chart_form = new class {

    init({color_palette}) {
        this._advanced_configuration = document.getElementById('adv_conf');

        this._advanced_configuration.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() {
        const show_advanced_configuration = this._advanced_configuration.checked;

        for (const element of this._form.querySelectorAll('.js-advanced-configuration')) {
            element.style.display = show_advanced_configuration ? '' : 'none';
        }
    }
};
```

CWidgetField

Класс *CWidgetField* — это базовый класс, от которого наследуются все классы полей формы (*CWidgetFieldCheckBox*, *CWidgetFieldTextArea*, *CWidgetFieldRadioButtonList* и т. д.). Классы, расширяющие *CWidgetField*, отвечают за получение, сохранение и проверку значений конфигурации виджета.

Доступны следующие классы *CWidgetField*.

| Класс *CWidgetField* | Тип поля базы данных | Описание |
|---|---|---|
| *CWidgetFieldCheckBox* | int32 | Одиночный флажок. |
| *CWidgetFieldCheckBoxList* | array of int32 | Несколько флажков в одном поле конфигурации. |
| *CWidgetFieldColor* | string | Поле выбора цвета. |
| *CWidgetFieldDatePicker* | string | Поле выбора даты. |
| *CWidgetFieldIntegerBox* | int32 | Поле для ввода целого числа. Может использоваться для настройки минимального и максимального значений. |
| *CWidgetFieldLatLng* | string | Текстовое поле, позволяющее вводить через запятую широту, долготу и уровень масштабирования карты. |
| *CWidgetFieldMultiSelect* | - | Базовый класс для полей множественного выбора, расширяемый всеми классами *CWidgetFieldMultiSelect**. |
| *CWidgetFieldMultiSelectActionID* | | Поле множественного выбора для выбора действий из списка действий, определенных в *Alerts → Actions*. |
| *CWidgetFieldMultiSelectGraphID* | | Поле множественного выбора для выбора пользовательских графиков. |
| *CWidgetFieldMultiSelectGraphPrototypeID* | | Поле множественного выбора для выбора прототипов пользовательских графиков. |
| *CWidgetFieldMultiSelectGroupID* | | Поле множественного выбора для выбора групп узлов сети. |
| *CWidgetFieldMultiSelectHostID* | | Поле множественного выбора для выбора узлов сети. |

| Класс *CWidgetField* | Тип поля базы данных | Описание |
|---|---|---|
| *CWidgetFieldMultiSelectHostInventory* | array of int32 | Поле множественного выбора для выбора полей инвентаря узла сети. |
| *CWidgetFieldMultiSelectItem* | ID | Поле множественного выбора для выбора элементов данных. |
| *CWidgetFieldMultiSelectItemPrototype* | ID | Поле множественного выбора для выбора прототипов элементов данных. |
| *CWidgetFieldMultiSelectMap* | ID | Поле множественного выбора для выбора карт. |
| *CWidgetFieldMultiSelectMediaType* | ID | Поле множественного выбора для выбора типов оповещений. |
| *CWidgetFieldMultiSelectOverrideHost* | ID | Поле множественного выбора для выбора источника данных для узлов сети с панели или другого виджета. |
| *CWidgetFieldMultiSelectService* | ID | Поле множественного выбора для выбора сервисов. |
| *CWidgetFieldMultiSelectSla* | ID | Поле множественного выбора для выбора SLA. |
| *CWidgetFieldMultiSelectUser* | ID | Поле множественного выбора для выбора пользователей. |
| *CWidgetFieldNavTree* | array of (multiple mixed) | Сохраняет структуру дерева навигации по картам для использования в виджете *Map navigation tree*. |
| *CWidgetFieldNumericBox* | string | Поле для ввода числа с плавающей точкой. |
| *CWidgetFieldPatternSelect* | - | Базовый класс для полей выбора по шаблону, расширяемый всеми классами *CWidgetPatternSelect**. |
| *CWidgetFieldPatternSelectHost* | string | Поле множественного выбора для выбора узлов сети. Поддерживает шаблоны имен узлов сети. |
| *CWidgetFieldPatternSelectItem* | string | Поле множественного выбора для выбора элементов данных. Поддерживает шаблоны имен элементов данных. |
| *CWidgetFieldRadioButtonList* | int32 | Группа переключателей, состоящая из одного или нескольких переключателей. |
| *CWidgetFieldRangeControl* | int32 | Ползунок для выбора значения целого типа. |
| *CWidgetFieldReference* | string | Сохраняет уникальный идентификатор этого виджета на панели; используется для взаимодействия между виджетами. |
| *CWidgetFieldSelect* | int32 | Выпадающий список. |
| *CWidgetFieldSeverities* | array of int32 | *CWidgetFieldCheckBoxList* с предустановленными уровнями важности триггеров. |
| *CWidgetFieldSparkline* | array of (multiple mixed) | Позволяет настраивать график sparkline. |
| *CWidgetFieldTags* | array of (string, int32, string) | Позволяет настраивать одну или несколько строк фильтра тегов. |
| *CWidgetFieldTextArea* | string | Многострочное текстовое поле. |
| *CWidgetFieldTextBox* | string | Текстовое поле для ввода одной строки. |
| *CWidgetFieldThresholds* | array of (string, string) | Позволяет настраивать пары цвет и число. |
| *CWidgetFieldTimePeriod* | array of string | Поле выбора временного периода. |
| *CWidgetFieldTimeZone* | string | Выпадающий список с часовыми поясами. |
| *CWidgetFieldUrl* | string | Текстовое поле, позволяющее вводить URL. |

**Note:**
Максимальная длина для полей `int32` составляет 10 символов, для полей `ID` — 20 символов, а для полей `string` — 2048 символов. Для составных типов (например, `array of (string, int32, string)`) ограничение применяется к каждому элементу отдельно. Ограничения длины для отдельных полей можно задать с помощью методов, таких как `setMaxLength()`.

**Презентация**

На этой странице описаны компоненты, которые можно использовать для создания представления виджета. Представление виджета - это часть виджета, которая получает данные в соответствии со своей конфигурацией и отображает их на информационной панели в контейнере.

Представление презентации состоит из трех частей:

- Действие виджета
- Просмотр виджета
- JavaScript

Действие виджета

Класс действий виджета (*WidgetView*) содержит методы для работы с виджетами в режиме просмотра презентации. Большинство действий виджетов используют и/или расширяют класс контроллера по умолчанию *CControllerDashboard-WidgetView*.

Класс действия виджета должен находиться в каталоге *actions* и указываться в параметре *actions* (*actions/widget.{id}.view/class*) в файле *manifest.json*.

**Пример actions/WidgetView.php (реализован в родном для Zabbix виджете *Информация о системе*)**

```
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()
            ]
        ]));
    }
}
```

Виджет

Класс представления виджета (*CWidgetView*) отвечает за построение представления виджета.

Класс представления виджета должен находиться в каталоге *views*. Если файл, содержащий класс представления виджета, имеет имя, отличное от имени по умолчанию (*widget.view.php*), то его необходимо указать в файле *manifest.json* *actions* параметр (*actions/widget.{id}.view/view*).

**пример views/widget.view.php**

```
<?php

/**
 * My custom widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['name'])
    )
    ->show();
```

JavaScript

Класс JavaScript отвечает за определение поведения виджета, например, обновление данных виджета, изменение размера виджета, отображение элементов виджета и т. д.

Все операции JavaScript используют и/или расширяют базовый класс JavaScript всех виджетов панели - *CWidget*. Класс *CWidget* содержит набор методов с реализацией по умолчанию для поведения виджета. В зависимости от сложности виджета эти методы можно использовать как есть или расширить.

Класс *CWidget* содержит следующие методы:

- Методы, определяющие жизненный цикл виджета: *onInitialize()*, *onStart()*, *onActivate()*, *onDeactivate()*, *onDestroy()*, *onEdit()*.
- Методы, обрабатывающие обновление и отображение данных виджета: *promiseUpdate()*, *getUpdateRequestData()*, *processUpdateResponse(response)*, *processUpdateErrorResponse(error)*, *setContents(response)*.
- Методы, изменяющие внешний вид виджета: *onResize()*, *hasPadding()*.

Класс JavaScript должен находиться в каталоге *assets/js* и указываться в параметре *assets* (*assets/js*) в *manifest.json* файл.

Методы жизненного цикла

Методы жизненного цикла виджета вызываются панелью и на разных этапах жизненного цикла виджета во время его существования на панели.

Метод ***onInitialize()*** определяет исходное состояние и/или значения виджета без выполнения каких-либо манипуляций с HTML или данными. Этот метод вызывается при создании виджета (создается экземпляр объекта виджета), обычно путем добавления виджета на страницу панели или загрузки страницы панели.

Пример:

```
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;
}
```

Метод ***onStart()*** определяет HTML-структуру виджета, не выполняя никаких манипуляций с данными. Этот метод вызывается перед первой активацией страницы панели мониторинга, то есть до того, как панель мониторинга и ее виджеты полностью отобразятся пользователю.

Пример:

```
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`
        );
    }
}
```

Метод ***onActivate()*** делает виджет активным и интерактивным, включая настраиваемые прослушиватели событий (для реагирования на действия пользователя) и запуская цикл обновления виджета (для поддержания актуальности его содержимого). Этот метод вызывается при активации страницы информационной панели, то есть когда она полностью отображается в пользовательском интерфейсе.

Обратите внимание, что до вызова метода *onActivate()* виджет находится в неактивном состоянии («WIDGET_STATE_INACTIVE»). После успешного вызова виджет переходит в активное состояние («WIDGET_STATE_ACTIVE»). В активном состоянии виджет реагирует, слушает события, периодически обновляет свое содержимое и может взаимодействовать с другими виджетами.

Пример:

```
onActivate() {
    this._startClock();

    this._resize_observer = new ResizeObserver(this._events.resize);
    this._resize_observer.observe(this._target);
}
```

Метод ***onDeactivate()*** останавливает любую активность и интерактивность виджета, деактивируя пользовательские прослушиватели событий и останавливая цикл обновления виджета. Этот метод вызывается, когда страница панели деактивируется, то есть отключается или удаляется, или когда виджет удаляется со страницы панели.

Обратите внимание, что до вызова метода *onDeactivate()* виджет находится в активном состоянии (WIDGET_STATE_ACTIVE). После успешного вызова виджет переходит в неактивное состояние («WIDGET_STATE_INACTIVE»).

Пример:

```
onDeactivate() {
    this._stopClock();
    this._resize_observer.disconnect();
}
```

Метод ***onDestroy()*** выполняет задачи очистки перед удалением виджета с панели управления, что может включать в себя закрытие соединения с базой данных, которое было установлено во время инициализации виджета, очистка временных данных для освобождения системной памяти и предотвращения утечек ресурсов, отмена регистрации прослушивателей событий, связанных с событиями изменения размера или нажатиями кнопок, чтобы предотвратить ненужную обработку событий и утечки памяти и т. д. Этот метод вызывается при удалении виджета или страницы информационной панели, которая его содержит.

Обратите внимание, что перед вызовом метода *onDestroy()* виджет в активном состоянии (`WIDGET_STATE_ACTIVE`) всегда деактивируется с помощью вызова метода *onDeactivate()*.

Пример:

```
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);
    }
}
```

Метод ***onEdit()*** определяет внешний вид и поведение виджета при переходе панели управления в режим редактирования. Этот метод вызывается, когда панель мониторинга переходит в режим редактирования, обычно когда пользователь взаимодействует с кнопкой *Редактировать* виджета или кнопкой *Редактировать панель* панели мониторинга.

Пример:

```
onEdit() {
    this._deactivateGraph();
}
```

Методы обновления виджета

Методы процесса обновления виджета отвечают за получение обновленных данных с сервера Zabbix или любого другого источника данных и их отображение в виджете.

Метод ***promiseUpdate()*** инициирует процесс обновления данных путем получения данных, обычно с помощью веб-запросов или вызовов API. Этот метод вызывается при отображении страницы панели и периодически после этого, пока страница панели не переключится на другую страницу панели.

Ниже приведен пример реализации по умолчанию метода *promiseUpdate()*, используемого большинством родных виджетов Zabbix. В реализации по умолчанию метод *promiseUpdate()* соответствует общему шаблону получения данных с сервера. Он создает новый объект Curl с соответствующим URL-адресом и параметрами запроса, отправляет запрос POST с помощью метода *fetch()* с объектом данных, созданным методом *getUpdateRequestData()*, и обрабатывает ответ (или ответ об ошибке) с помощью *processUpdateResponse(response)* или *processUpdateErrorResponse(error)* соответственно. Эта реализация подходит для большинства виджетов, поскольку они обычно получают данные в формате JSON и обрабатывают их согласованным образом.

```
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);
        });
    }
```

Метод ***getUpdateRequestData()*** подготавливает данные запроса сервера для обновления виджета, собирая различные свойства и соответствующие им значения (идентификаторы виджета, настройки фильтра, временные диапазоны и т. д.) из состояния и конфигурации виджета, и создание объекта данных, который представляет необходимую информацию, подлежащую отправке на сервер в запросе на обновление. Этот метод вызывается только как часть метода *promiseUpdate()* по умолчанию, то есть во время процесса обновления виджета.

Реализация по умолчанию:

```
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
    };
}
```

Метод ***processUpdateResponse(response)*** обрабатывает ответ, полученный от сервера после запроса на обновление, и, если процесс обновления прошел успешно и без ошибок, очищает данные виджета и отображает новое содержимое с помощью метода *setContents()*. Этот метод вызывается только как часть метода *promiseUpdate()* по умолчанию, то есть во время процесса обновления виджета.

Реализация по умолчанию:

```
processUpdateResponse(response) {
    this._setHeaderName(response.name);

    this._updateMessages(response.messages);
    this._updateInfo(response.info);
    this._updateDebug(response.debug);

    this.setContents(response);
}
```

Метод ***processUpdateErrorResponse(error)*** обрабатывает ответ, полученный от сервера после запроса на обновление, если ответ является ошибкой, и отображает сообщения об ошибках. Этот метод вызывается только как часть метода *promiseUpdate()* по умолчанию, то есть во время процесса обновления виджета.

Реализация по умолчанию:

```
processUpdateErrorResponse(error) {
    this._updateMessages(error.messages, error.title);
}
```

Метод ***setContents(response)*** отображает содержимое виджета, если процесс обновления виджета прошел успешно и без ошибок, что может включать в себя манипулирование элементами DOM, обновление компонентов пользовательского интерфейса, применение стилей или форматирования и т. д. Этот метод вызывается только как часть метода *processUpdateResponse(response)* по умолчанию, то есть в процессе обработки ответа, полученного от сервера после запроса на обновление.

Реализация по умолчанию:

```
setContents(response) {
    this._body.innerHTML = response.body ?? '';
}
```

Методы модификации представления

Методы модификации представления виджета отвечают за изменение внешнего вида виджета.

Метод ***onResize()*** отвечает за настройку визуальных элементов виджета в соответствии с новым размером виджета, которые могут включать перестановку элементов, настройку размеров элементов, усечение текста, реализацию отложенной загрузки для улучшения реагирования при изменении размера и т. д. Этот метод вызывается при изменении размера виджета, например, когда пользователь вручную изменяет размер виджета или когда изменяется размер окна браузера.

Пример:

```
onResize() {
    if (this.getState() === WIDGET_STATE_ACTIVE) {
        this._startUpdating();
    }
}
```

Метод ***hasPadding()*** отвечает за применение вертикального заполнения шириной 8 пикселей в нижней части виджета, когда он настроен на показ заголовка. Этот метод вызывается, когда страница информационной панели активируется, то есть когда она становится отображаемой страницей в пользовательском интерфейсе.

Реализация по умолчанию:

```
hasPadding() {
    return this.getViewMode() !== ZBX_WIDGET_VIEW_MODE_HIDDEN_HEADER;
}
```

Для некоторых виджетов необходимо использовать все доступное пространство виджетов, например, для настройки собственного цвета фона. Ниже приведен пример реализации метода *hasPadding()*, используемого в родном для Zabbix виджете *значение элемента данных*.

```
hasPadding() {
    return false;
}
```

**Руководства**

Этот раздел содержит практические пошаговые руководства, иллюстрирующие, как создать собственный модуль и виджет в Zabbix.

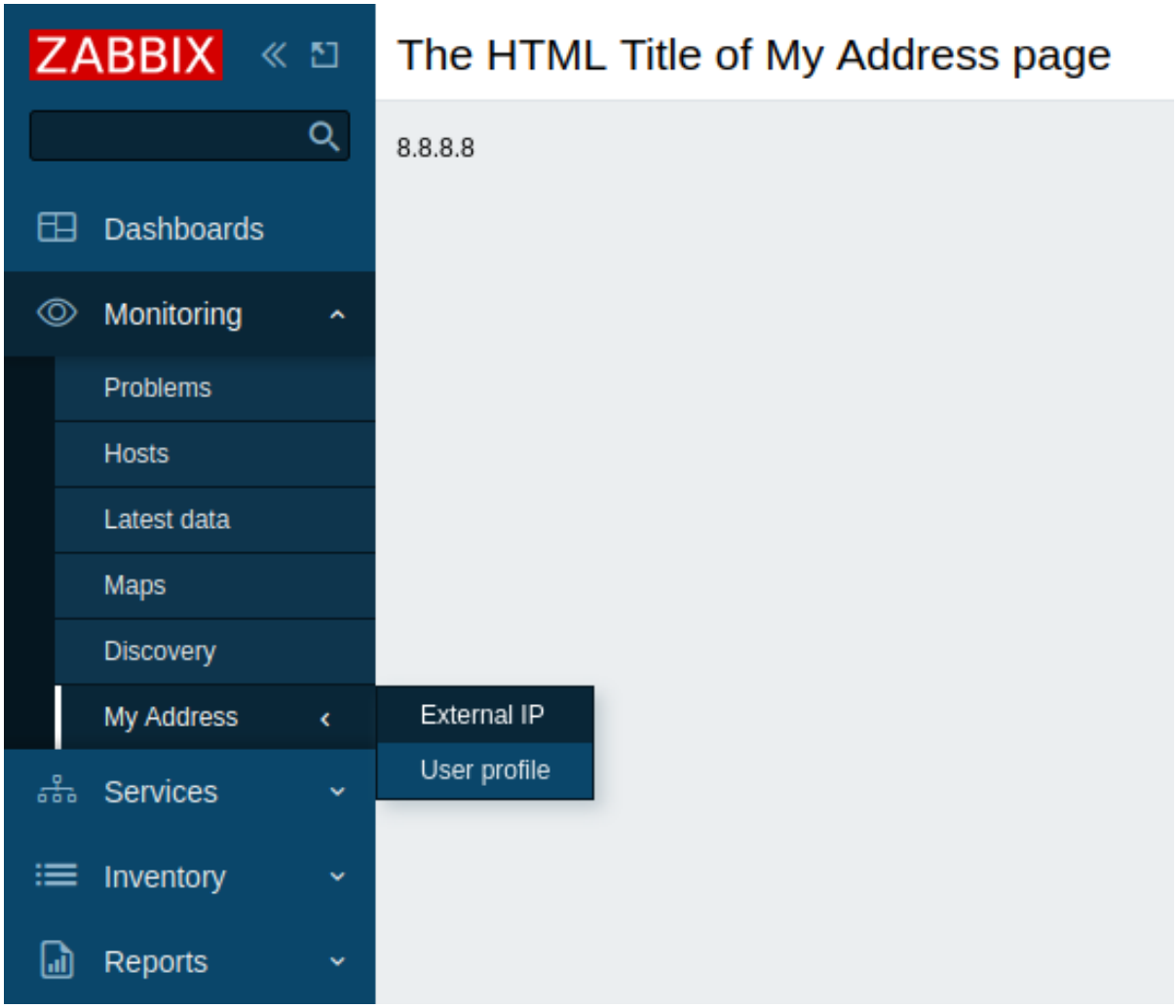
**Создание модуля (руководство)**

Эта пошаговая инструкция показывает, как создать простой модуль веб-интерфейса Zabbix. Вы можете скачать все файлы этого модуля в виде ZIP-архива: MyAddress.zip.

Что вы создадите

В ходе этого руководства вы сначала создадите модуль веб-интерфейса, который добавляет новый раздел меню *My Address*, а затем преобразуете его в более продвинутый модуль веб-интерфейса, который выполняет HTTP-запрос к *https://api.seeip.org* и отображает ответ — IP-адрес вашего компьютера — на новой странице в новом разделе меню *My Address*.

Вот как будет выглядеть готовый модуль:




Часть I - Новый раздел меню

Добавление пустого модуля в веб-интерфейс Zabbix

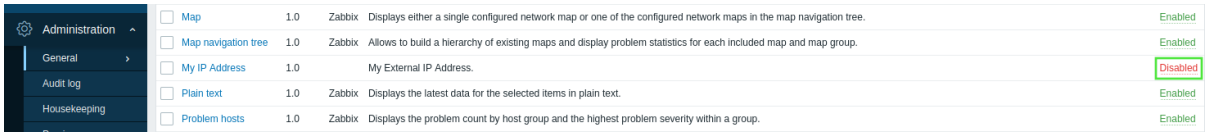
1. Создайте каталог *MyAddress* в каталоге *modules* вашей установки веб-интерфейса Zabbix (например, *zabbix/ui/modules*).
2. Создайте файл *manifest.json* с базовыми метаданными модуля (см. описание поддерживаемых параметров).

**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. В веб-интерфейсе Zabbix перейдите в раздел *Administration* → *General* → *Modules* и нажмите кнопку *Scan directory*.

4. Найдите новый модуль *My IP Address* в списке и нажмите гиперссылку "Disabled", чтобы изменить статус модуля с "Disabled" на "Enabled" (если модуль не отображается в списке, см. раздел troubleshooting).




Теперь модуль зарегистрирован в веб-интерфейсе. Однако он пока не отображается, поскольку вам еще нужно определить функциональность модуля. После добавления содержимого в каталог модуля изменения сразу станут видны

в веб-интерфейсе Zabbix после обновления страницы.

Создание раздела меню

1. Создайте файл *Module.php* в каталоге *MyAddress*.

Этот файл реализует новый класс *Module*, который наследуется от стандартного класса *CModule*. Класс *Module* добавит новый раздел меню *My Address* в главное меню.

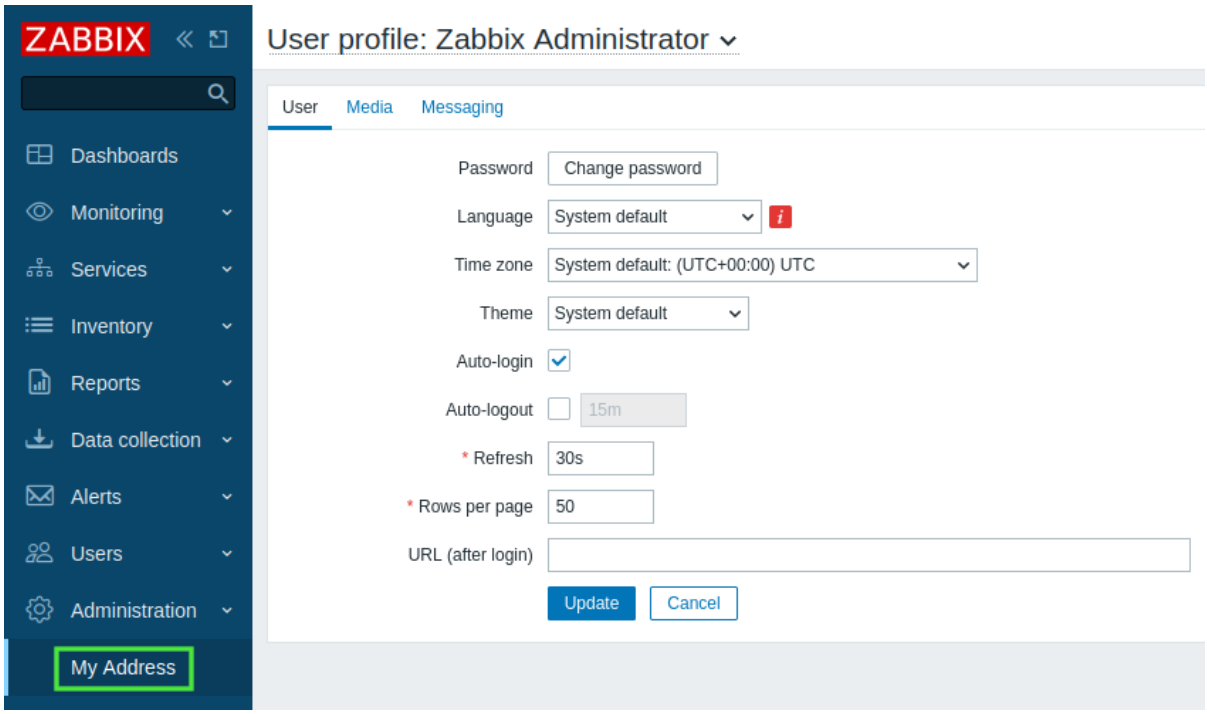
Метод *setAction()* задает действие, которое будет выполнено при щелчке по разделу меню. Для начала можно использовать предопределенное действие *userprofile.edit*, которое откроет страницу *Профиль пользователя*. В части III этого руководства вы узнаете, как создать собственное действие.

**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:**
Вы можете заменить `'userprofile.edit'` на другие действия, например, `'charts.view'` (открывает пользовательские графики), `'problems.view'` (открывает *Мониторинг → Проблемы*) или `'report.status'` (открывает отчет *Сведения о системе*).

3. Обновите веб-интерфейс Zabbix. Теперь в нижней части главного меню Zabbix появился новый раздел *My Address*. Нажмите *My Address*, чтобы открыть страницу *Профиль пользователя*.




Часть II - Изменение расположения раздела меню

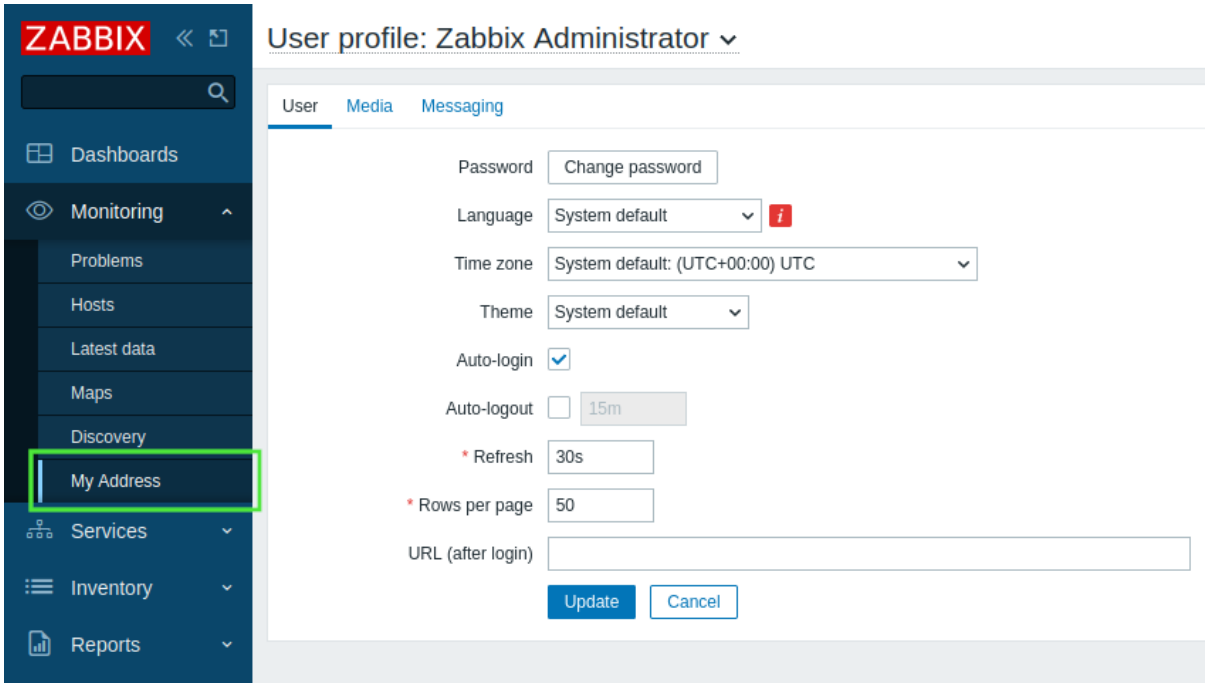
В этой части вы переместите раздел меню *Мой адрес* в раздел *Мониторинг*, а затем добавите в него вложенное меню. В результате пользователи смогут получить доступ к двум страницам подменю из раздела меню *Мониторинг → Мой адрес*.

1. Откройте и отредактируйте файл *Module.php*.

**ui/modules/MyAddress/Module.php**

```
<?php

namespace Modules\MyAddress;

use Zabbix\Core\CModule,
    APP,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->findOrAdd(_('Monitoring'))
            ->getSubmenu()
            ->insertAfter(_('Discovery'),
                (new CMenuItem(_('My Address')))->setAction('userprofile.edit')
            );
    }
}
```

2. Обновите интерфейс Zabbix. Разверните раздел меню *Мониторинг* и обратите внимание, что раздел *Мой адрес* теперь расположен под разделом *Обнаружение*.




3. Чтобы добавить вложенные страницы в раздел меню *Мой адрес*, снова откройте и отредактируйте файл *Module.php*.

На этом этапе создаются два подраздела:

- *Внешний IP*, который выполняет новое действие «my.address», которое будет определено на следующих шагах;
- *Профиль пользователя*, который выполняет предопределенное действие «userprofile.edit», чтобы открыть страницу *Профиль пользователя*.

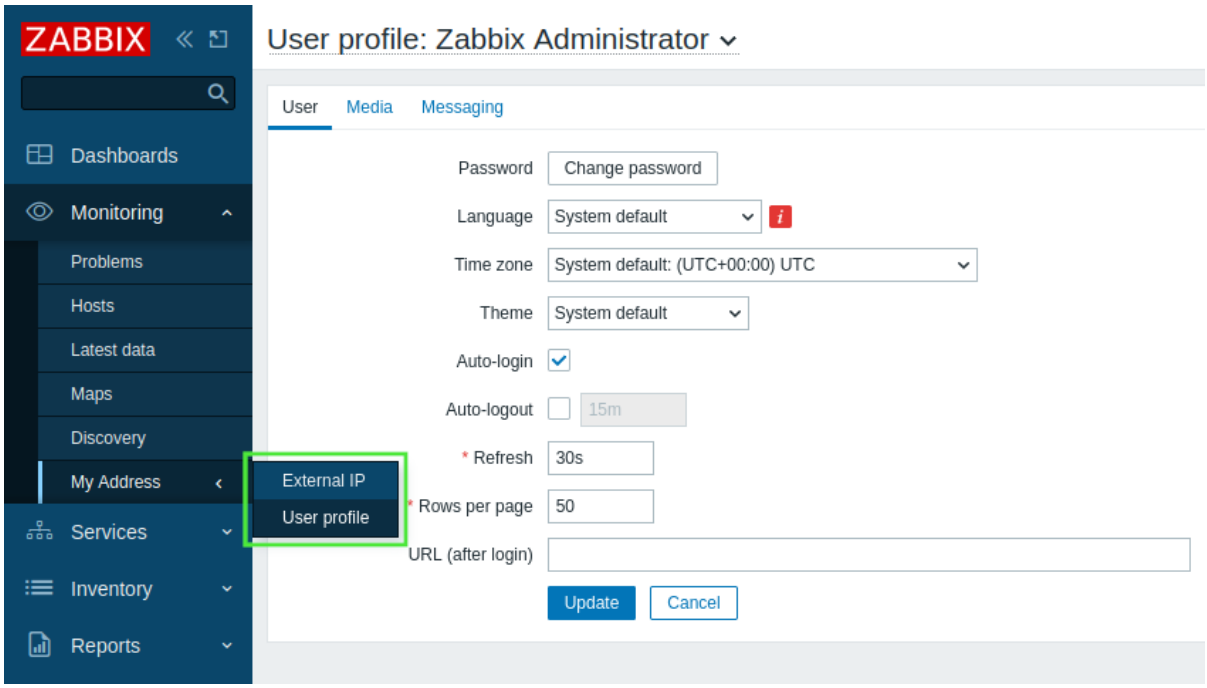
Обратите внимание, что для вложенного меню вам необходимо использовать класс *CMenu* в дополнение к классам, использованным на предыдущих шагах.

**ui/modules/MyAddress/Module.php**

```
<?php

namespace Modules\MyAddress;

use Zabbix\Core\CModule,
```

```
    APP,
    CMenu,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->findOrAdd(_('Monitoring'))
            ->getSubmenu()
            ->insertAfter(_('Discovery'),
                (new CMenuItem(_('My Address')))->setSubMenu(
                    new CMenu([
                        (new CMenuItem(_('External IP')))->setAction('my.address'),
                        (new CMenuItem(_('User profile')))->setAction('userprofile.edit')
                    ])
                )
            );
    }
}
```

4. Обновите интерфейс Zabbix. Обратите внимание, что раздел меню *Мой адрес* теперь содержит меню третьего уровня с двумя страницами — *Внешний IP* и *Профиль пользователя*.




Часть III - Действие модуля

Действие реализуется в двух файлах - *actions/MyAddress.php* и *views/my.address.php*. Файл ***actions/MyAddress.php*** отвечает за реализацию бизнес-логики, а файл ***views/my.address.php*** - за представление.

1. Создайте каталог *actions* в каталоге *MyAddress*.
2. Создайте файл *MyAddress.php* в каталоге *actions*.

Логика действия будет определена в классе *MyAddress*. Этот класс действия будет реализовывать четыре функции: *init()*, *checkInput()*, *checkPermissions()* и *doAction()*. Веб-интерфейс Zabbix вызывает функцию *doAction()* при запросе действия. Эта функция отвечает за бизнес-логику модуля.

**Attention:**
Данные должны быть организованы в виде ассоциативного массива. Массив может быть многомерным и может содержать любые данные, ожидаемые представлением.

**ui/modules/MyAddress/actions/MyAddress.php**

```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. Создайте новый каталог *views* в каталоге *MyAddress*.
4. Создайте файл *my.address.php* в каталоге *views* и определите представление модуля.

Обратите внимание, что переменная $data доступна в представлении без явного определения. Фреймворк автоматически передает ассоциативный массив в представление.

**ui/modules/MyAddress/views/my.address.php**

```php
<?php

(new CHtmlPage())
    ->setTitle(_('The HTML Title of My Address Page'))
    ->addItem(new CDiv($data['my-ip']))
    ->show();
```

5. Действие модуля должно быть зарегистрировано в файле *manifest.json*. Откройте *manifest.json* и добавьте новый объект `actions`, который содержит:

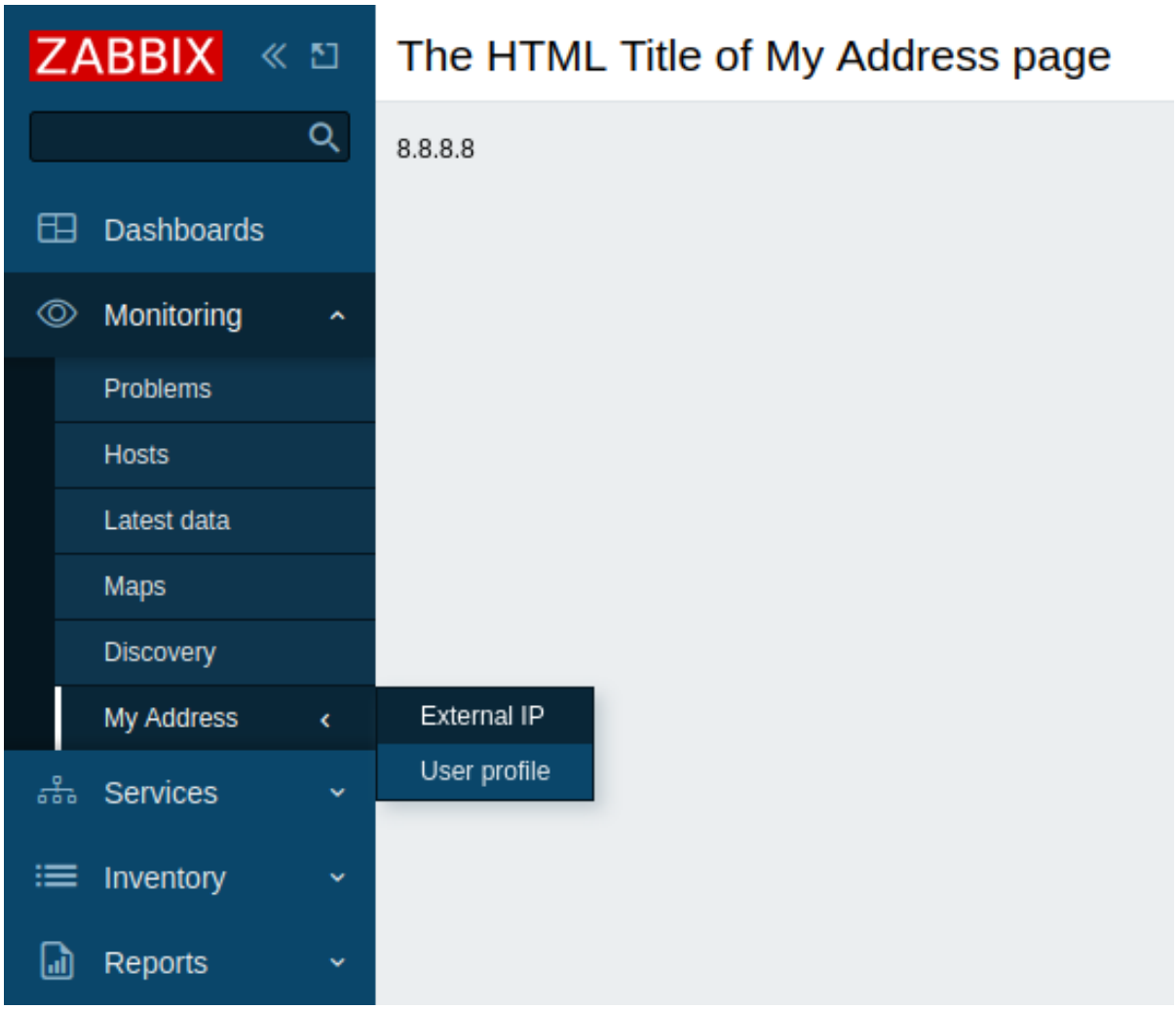
- ключ действия с именем действия, записанным в нижнем регистре (a-z) и с разделением слов точками (например, `my.address`);
- имя класса действия (`MyAddress`) в качестве значения ключа `class` объекта `my.address`;
- имя представления действия (`my.address`) в качестве значения ключа `view` объекта `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"
        }
```

```
    }
}
```

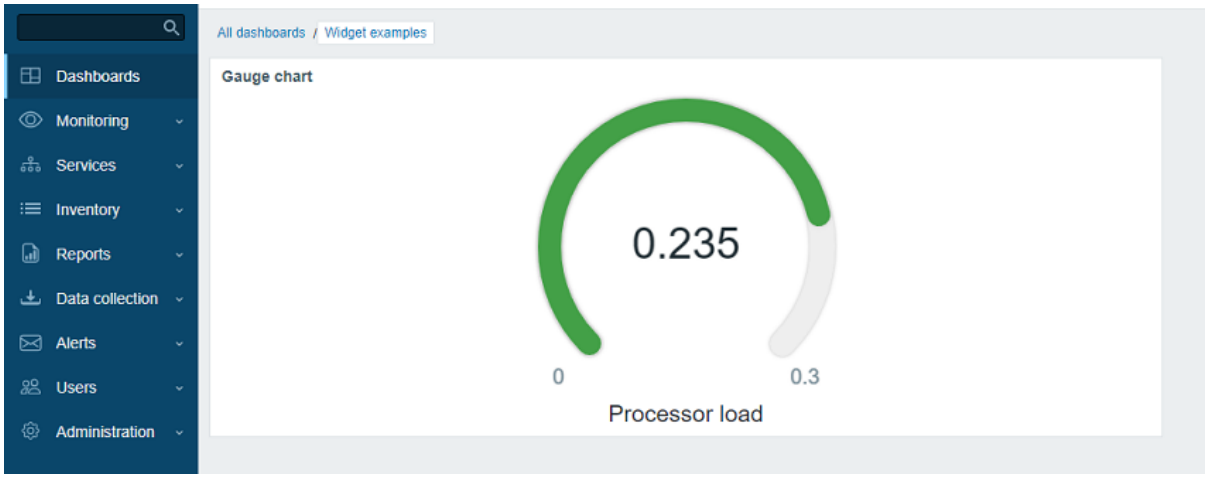
6. Обновите веб-интерфейс Zabbix. Нажмите *My address* → *External IP*, чтобы увидеть IP-адрес вашего компьютера.




**Создание виджета (руководство)**

Это пошаговое руководство, которое показывает, как создать простой виджет информационной панели. Вы можете скачать все файлы этого виджета в виде ZIP-архива: lesson_gauge_chart.zip.

**Что вы создадите** В ходе этого руководства вы сначала создадите базовый виджет "Hello, world!", а затем преобразуете его в более продвинутый виджет, который отображает значение элемента данных в виде диаграммы-датчика. Вот как будет выглядеть готовый виджет:



**Часть I - ”Hello, world!”** В этом разделе вы узнаете, как создать минимально необходимые элементы виджета и добавить новый виджет в веб-интерфейс Zabbix.

Добавление пустого виджета в веб-интерфейс Zabbix

1. Создайте каталог *lesson_gauge_chart* в каталоге *modules* вашей установки веб-интерфейса Zabbix (например, *zabbix/ui/modules*).

> **Note:**
> Все пользовательские виджеты рассматриваются как внешние модули и должны быть добавлены в каталог *modules* вашей установки веб-интерфейса Zabbix (например, *zabbix/ui/modules*). Каталог *zabbix/ui/widgets* зарезервирован для встроенных виджетов Zabbix и обновляется вместе с веб-интерфейсом Zabbix.

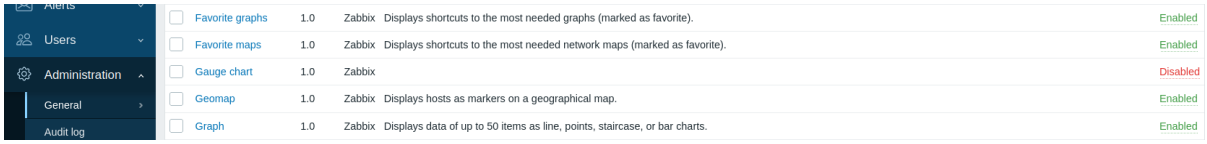
2. Создайте файл *manifest.json* с базовыми метаданными виджета (см. описание поддерживаемых параметров).

**ui/modules/lesson_gauge_chart/manifest.json**

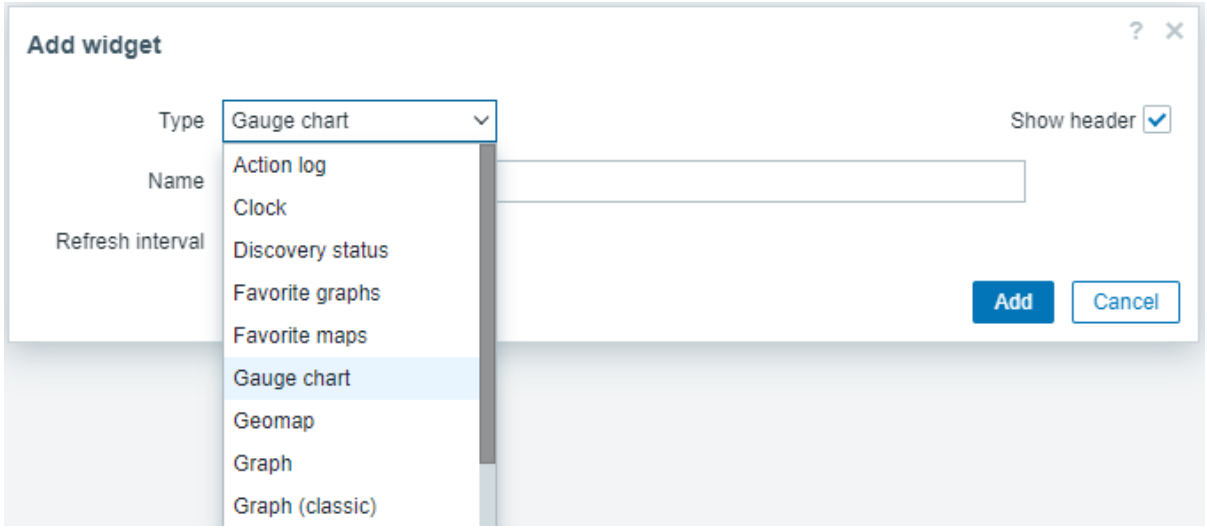
```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.1",
    "author": "Zabbix"
}
```

3. В веб-интерфейсе Zabbix перейдите в раздел *Administration* → *General* → *Modules* и нажмите кнопку *Scan directory*.

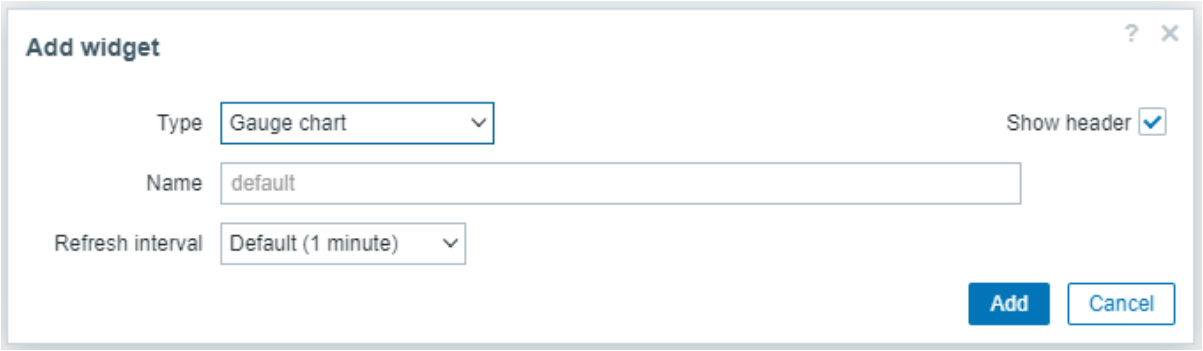
4. Найдите новый модуль *Gauge chart* в списке и нажмите гиперссылку ”Disabled”, чтобы изменить статус модуля с ”Disabled” на ”Enabled” (если модуль не отображается в списке, см. раздел troubleshooting).



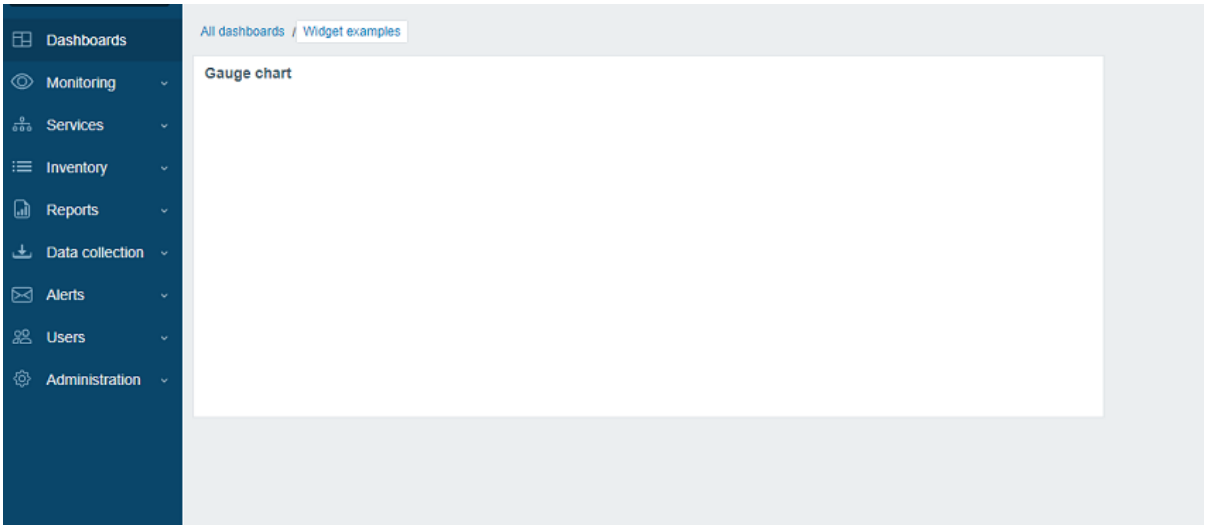

5. Откройте панель, переключите ее в режим редактирования и добавьте новый виджет. В поле *Type* выберите ”Gauge chart”.




6. На этом этапе конфигурация виджета *Gauge chart* содержит только общие поля виджета *Name* и *Refresh interval*. Нажмите *Add*, чтобы добавить виджет на панель.




7. На панели должен появиться пустой виджет. Нажмите *Save changes* в правом верхнем углу, чтобы сохранить панель.




Добавление представления виджета

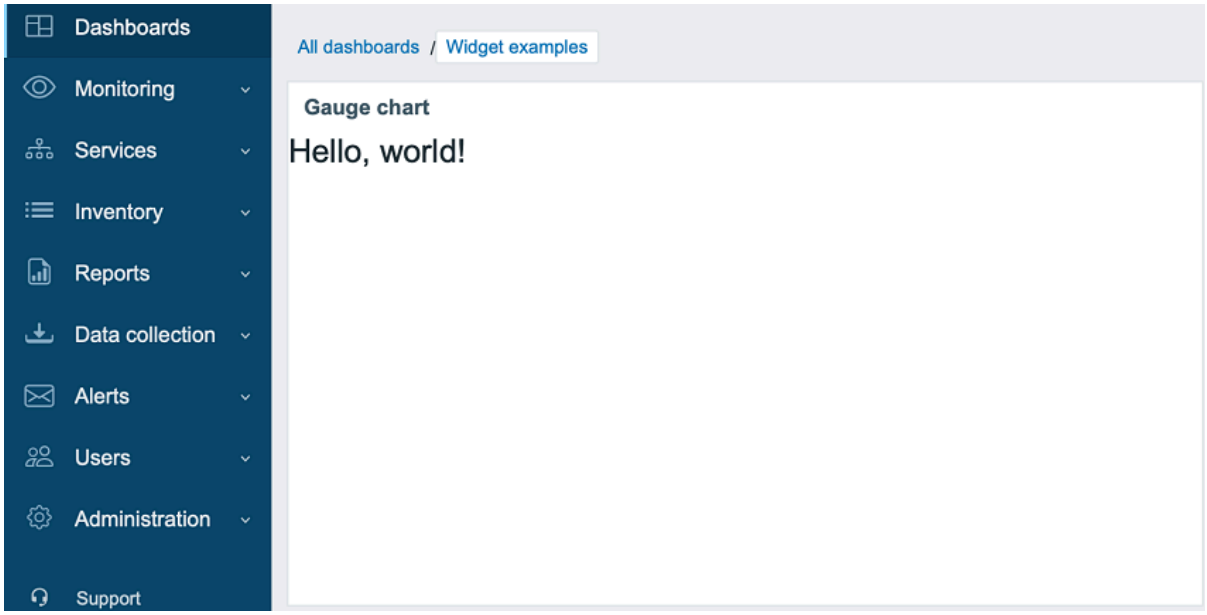
**Note:**
Файл **view** виджета должен находиться в каталоге *views* (для этого руководства — *ui/modules/lesson_gauge_chart/views/*). Если файл имеет имя по умолчанию *widget.view.php*, регистрировать его в файле *manifest.json* не нужно. Если файл имеет другое имя, укажите его в разделе *actions/widget.lesson_gauge_chart.view* файла manifest.json.

1. Создайте каталог *views* в каталоге *lesson_gauge_chart*.
2. Создайте файл *widget.view.php* в каталоге *views*.

**ui/modules/lesson_gauge_chart/views/widget.view.php**

```
<?php

/**
 *              Gauge chart.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, 'Hello, world!')
    )
    ->show();
```

3. Обновите панель мониторинга. Теперь виджет *Gauge chart* отображает ”Hello, world!”.




**Часть II - Диаграмма измерений** Добавление настроек в представление конфигурации и использование их в представлении виджета

В этом разделе вы узнаете, как добавить поле конфигурации виджета и вывести введенное значение в представлении виджета в виде текста.

Конфигурация виджета состоит из формы (*Zabbix\Widgets\CWidgetForm*) и представления формы виджета (*widget.edit.php*). Чтобы добавить поля (*Zabbix\Widgets\CWidgetField*), нужно создать класс *WidgetForm*, который будет расширять *Zabbix\Widgets\CWidgetForm*.

Форма содержит набор полей (*Zabbix\Widgets\CWidgetField*) различных типов, которые используются для проверки значений, введенных пользователем. Поле формы (*Zabbix\Widgets\CWidgetField*) для каждого типа элемента ввода преобразует значение в единый формат для хранения в базе данных.

> **Note:**
> Файл **form** виджета должен находиться в каталоге *includes* (для этого руководства — *ui/modules/lesson_gauge_chart/includes/*). Если файл имеет имя по умолчанию *WidgetForm.php*, регистрировать его в файле *manifest.json* не нужно. Если файл имеет другое имя, укажите его в разделе *widget/form_class* файла manifest.json.

1. Создайте новый каталог *includes* в каталоге *lesson_gauge_chart*.
2. Создайте файл *WidgetForm.php* в каталоге *includes*.

**ui/modules/lesson_gauge_chart/includes/WidgetForm.php**

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\CWidgetForm;

class WidgetForm extends CWidgetForm {
}
```

3. Добавьте поле *Description* в форму конфигурации виджета. Это обычное текстовое поле, в которое пользователь может ввести любой набор символов. Для него можно использовать класс *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'))
            );
   }
}
```

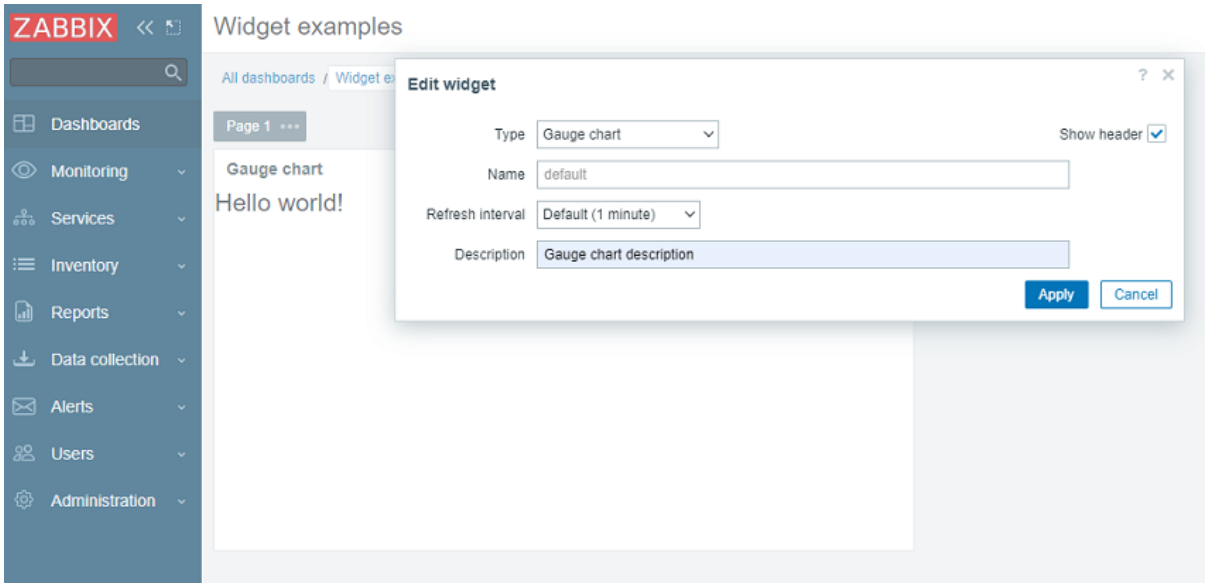
4. В каталоге *views* создайте файл представления конфигурации виджета *widget.edit.php* и добавьте представление для нового поля *Description*. Для класса поля *CWidgetFieldTextBox* представлением является *CWidgetFieldTextBoxView*.

**ui/modules/lesson_gauge_chart/views/widget.edit.php**

```
<?php

/**
 *                  Gauge chart.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

5. Перейдите на панель и нажмите значок шестеренки в виджете, чтобы открыть форму конфигурации виджета.

6. Теперь форма конфигурации виджета содержит новое текстовое поле *Description*. Введите любое значение, например *Gauge chart description*.




7. Нажмите *Apply* в форме конфигурации виджета. Затем нажмите *Save changes* в правом верхнем углу, чтобы сохранить панель. Обратите внимание, что новое описание нигде не отображается, и виджет по-прежнему показывает "Hello, world!".

Чтобы новое описание появилось в виджете, значение поля *Description* нужно получить из базы данных и передать в представление виджета. Для этого необходимо создать класс действия.

8. Создайте новый каталог *actions* в каталоге *lesson_gauge_chart*.

9. Создайте файл *WidgetView.php* в каталоге *actions*. Класс действия *WidgetView* будет расширять класс *CControllerDashboardWidgetView*.

Значения полей конфигурации виджета хранятся в свойстве **$fields_values** класса действия.

**ui/modules/lesson_gauge_chart/actions/WidgetView.php**

```
<?php

namespace Modules\LessonGaugeChart\Actions;

use CControllerDashboardWidgetView,
    CControllerResponseData;

class WidgetView extends CControllerDashboardWidgetView {

    protected function doAction(): void {
        $this->setResponse(new CControllerResponseData([
            'name' => $this->getInput('name', $this->widget->getName()),
            'description' => $this->fields_values['description'],
            'user' => [
                'debug_mode' => $this->getDebugMode()
            ]
        ]));
    }
}
```

10. Откройте *manifest.json* и зарегистрируйте *WidgetView* как класс действия в разделе *actions/widget.lesson_gauge_chart.view*.

**ui/modules/lesson_gauge_chart/manifest.json**

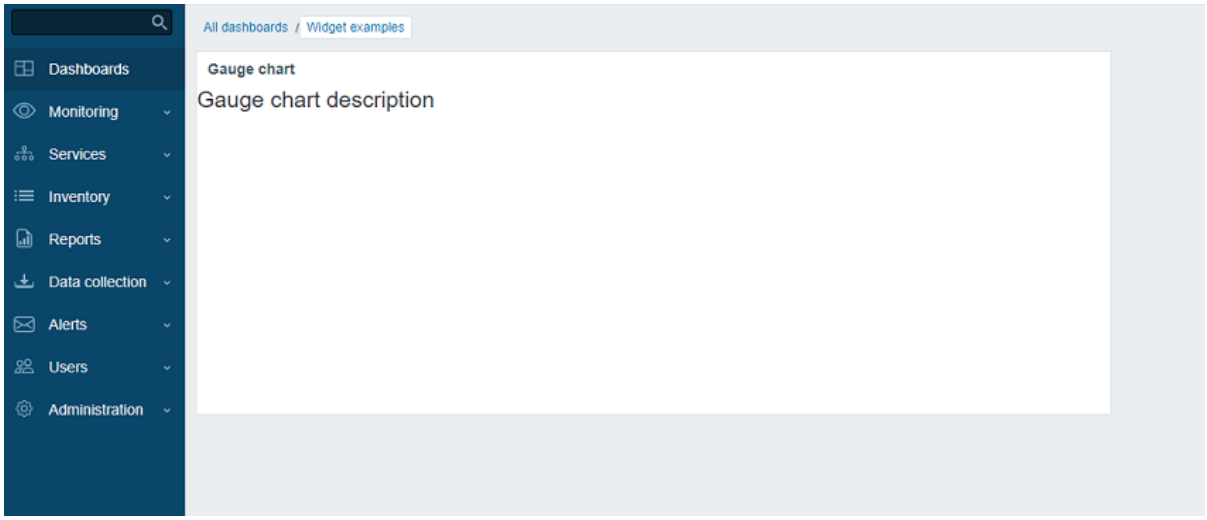
```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.0",
    "author": "Zabbix",
    "actions": {
        "widget.lesson_gauge_chart.view": {
            "class": "WidgetView"
        }
    }
}
```

11. Теперь вы можете использовать значение поля description, содержащееся в *$data['description']*, в представлении виджета. Откройте *views/widget.view.php* и замените статический текст "Hello, world!" на *$data['description']*.

**ui/modules/lesson_gauge_chart/views/widget.view.php**

```
<?php

/**
 *              Gauge chart.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['description'])
    )
    ->show();
```

12. Обновите страницу панели. Теперь вместо "Hello, world!" вы должны видеть текст описания виджета.




Получение значения элемента данных через API

Виджет должен отображать последнее значение элемента данных, выбранного пользователем. Для этого нужно добавить возможность выбора элементов данных в конфигурации виджета.

В этом разделе вы узнаете, как добавить поле выбора элемента данных в форму виджета и как добавить визуальную часть этого поля в представление конфигурации. Затем контроллер виджета сможет получать данные элемента данных и его значение через API-запрос. После получения значение можно отобразить в представлении виджета.

1. Откройте *includes/WidgetForm.php* и добавьте поле *CWidgetFieldMultiSelectItem*. Это позволит выбирать элемент данных в форме конфигурации.

**ui/modules/lesson_gauge_chart/includes/WidgetForm.php**

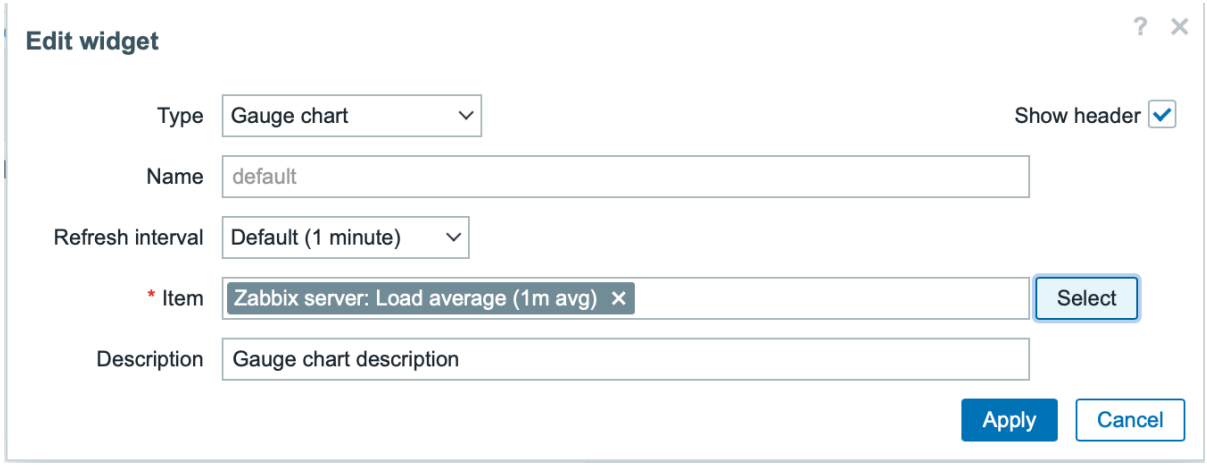
```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\{
    CWidgetField,
    CWidgetForm
};

use Zabbix\Widgets\Fields\{
    CWidgetFieldMultiSelectItem,
    CWidgetFieldTextBox
};

/**
 *              -    .
 */
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. Откройте *views/widget.edit.php* и добавьте визуальный компонент поля в представление конфигурации.

**ui/modules/lesson_gauge_chart/views/widget.edit.php**

```php
<?php

/**
 *                      -    .
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldMultiSelectItemView($data['fields']['itemid'])
    )
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

3. Вернитесь на панель и нажмите значок шестеренки в виджете, чтобы открыть форму конфигурации виджета.
4. Теперь форма конфигурации виджета содержит новое поле ввода *Item*. Выберите узел сети "Zabbix server" и элемент данных "Load average (1m avg)".




5. Нажмите *Apply* в форме конфигурации виджета. Затем нажмите *Save changes* в правом верхнем углу, чтобы сохранить панель.
6. Откройте и измените *actions/WidgetView.php*.

Теперь идентификатор элемента данных будет доступен в контроллере виджета в *$this->fields_values['itemid']*. Метод контроллера *doAction()* получает данные элемента данных (имя, тип значения, единицы измерения) с помощью API-метода *item.get* и последнее значение элемента данных с помощью API-метода *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]
```

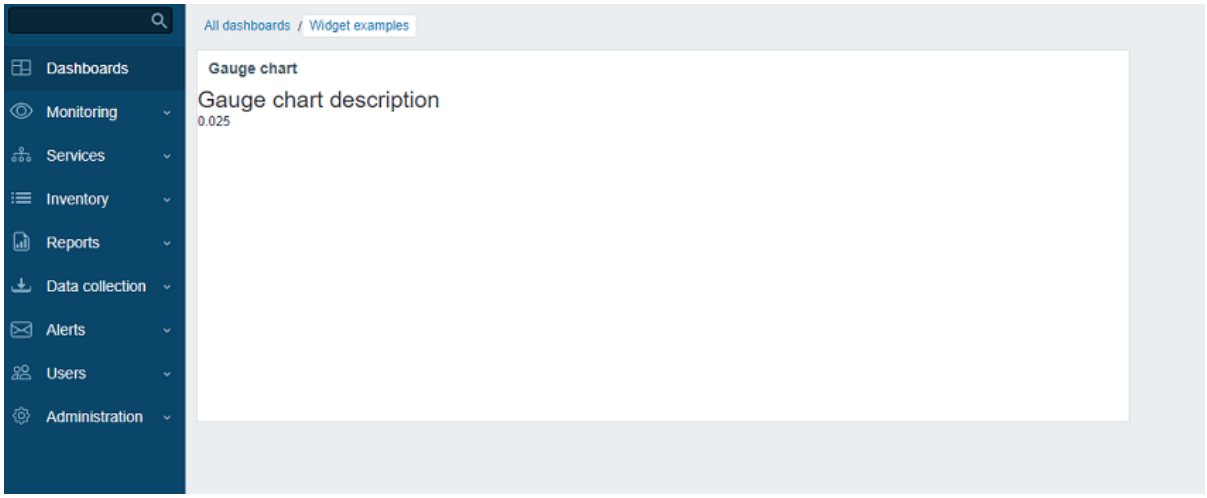
```
            ]
        ]);

        $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. Откройте *views/widget.view.php* и добавьте значение элемента данных в представление виджета.

**ui/modules/lesson_gauge_chart/views/widget.view.php**

```
<?php

/**
 *                  -    .
 *
 * @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. Обновите страницу панели. Виджет будет отображать последнее значение элемента данных.




Добавление расширенных настроек конфигурации в представление конфигурации

В этом разделе вы узнаете, как добавить сворачиваемый/разворачиваемый раздел *Advanced configuration* с необязательными параметрами, такими как цвет, минимальное и максимальное значения, единицы измерения и поле *Description*, созданное ранее.

1. Создайте файл *Widget.php* в основном каталоге виджета *lesson_gauge_chart*, чтобы создать новый класс *Widget*.

Класс *Widget* будет расширять базовый класс *CWidget*, чтобы добавлять/переопределять настройки виджета по умолчанию (в данном случае - переводы). JavaScript, приведенный ниже, отображает строку ”No data” в случае отсутствия данных. Строка ”No data” присутствует в файлах перевода интерфейса Zabbix.

Если у виджета есть какие-либо константы, рекомендуется также указать их в классе *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. Откройте *includes/WidgetForm.php* и добавьте новые поля *Color* (выбор цвета), *Min* (числовое поле), *Max* (числовое поле) и *Units* (список), а также задайте палитру цветов по умолчанию для выбора цвета, чтобы ее можно было использовать на следующих шагах.

**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
};

/**
 *              -    .
 */
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. Откройте *views/widget.edit.php* и добавьте визуальные компоненты полей в представление конфигурации.

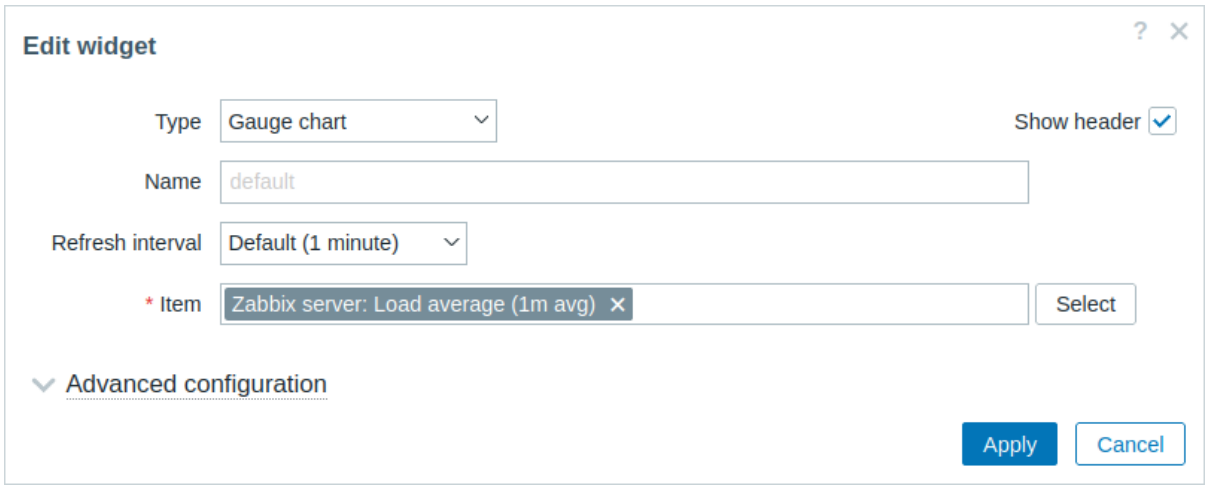
**ui/modules/lesson_gauge_chart/views/widget.edit.php**

```
<?php

/**
 *                      -    .
 *
```

```
 * @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();
```

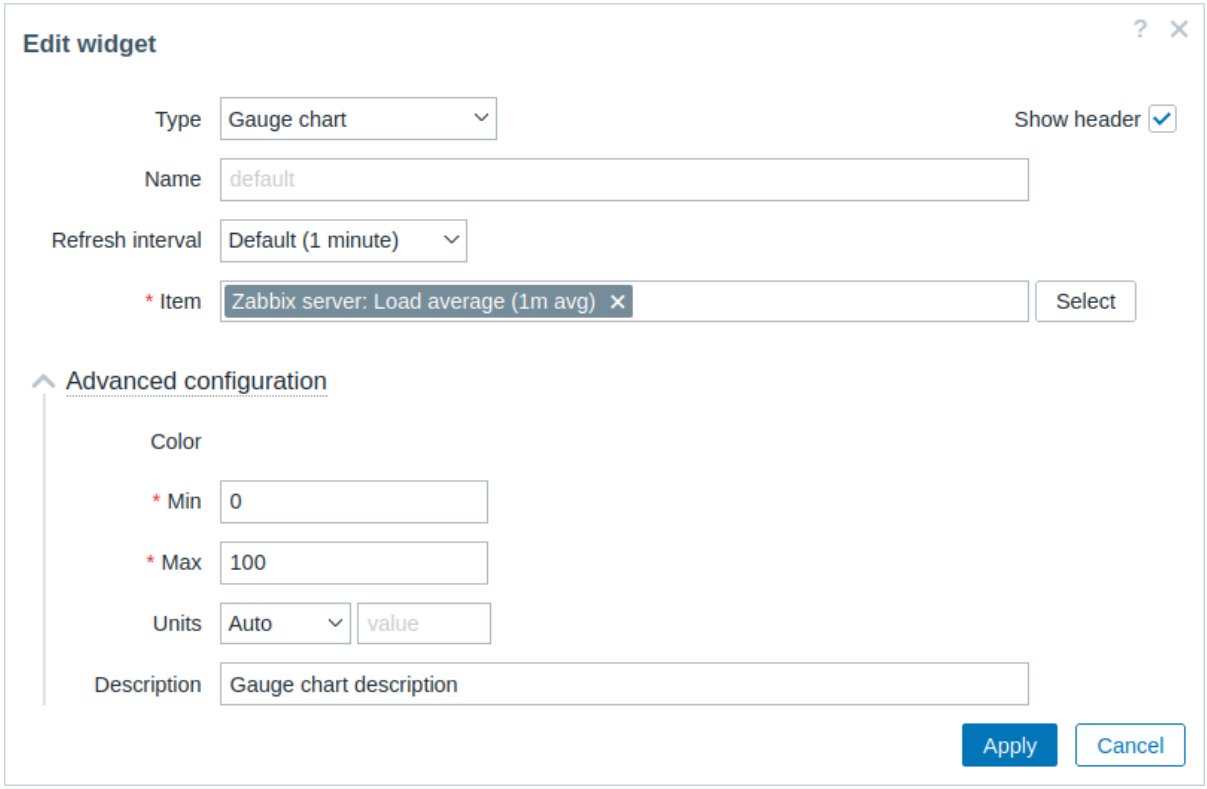
**Note:**
Метод *addField()* класса *CWidgetFormView* принимает строку с CSS-классом в качестве второго параметра.

4. Вернитесь на панель, переключитесь в режим редактирования и нажмите значок шестеренки в виджете, чтобы открыть форму конфигурации виджета. Теперь форма конфигурации виджета содержит новый сворачиваемый/разворачиваем раздел *Advanced configuration*.




5. Разверните раздел *Advanced configuration*, чтобы увидеть дополнительные поля конфигурации виджета. Обратите внимание, что у поля *Color* пока нет выбора цвета. Это связано с тем, что выбор цвета должен инициализироваться

с помощью JavaScript, который будет добавлен в следующем разделе - *Add JavaScript to the widget*.




Добавление JavaScript в виджет

В этом разделе вы узнаете, как добавить диаграмму-датчик, созданную с помощью JavaScript, которая показывает, является ли последнее значение нормальным или слишком высоким/слишком низким.

1. Создайте файл *widget.edit.js.php* в каталоге *views*.

JavaScript будет отвечать за инициализацию палитры цветов в окне конфигурации.

**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();
    }
```

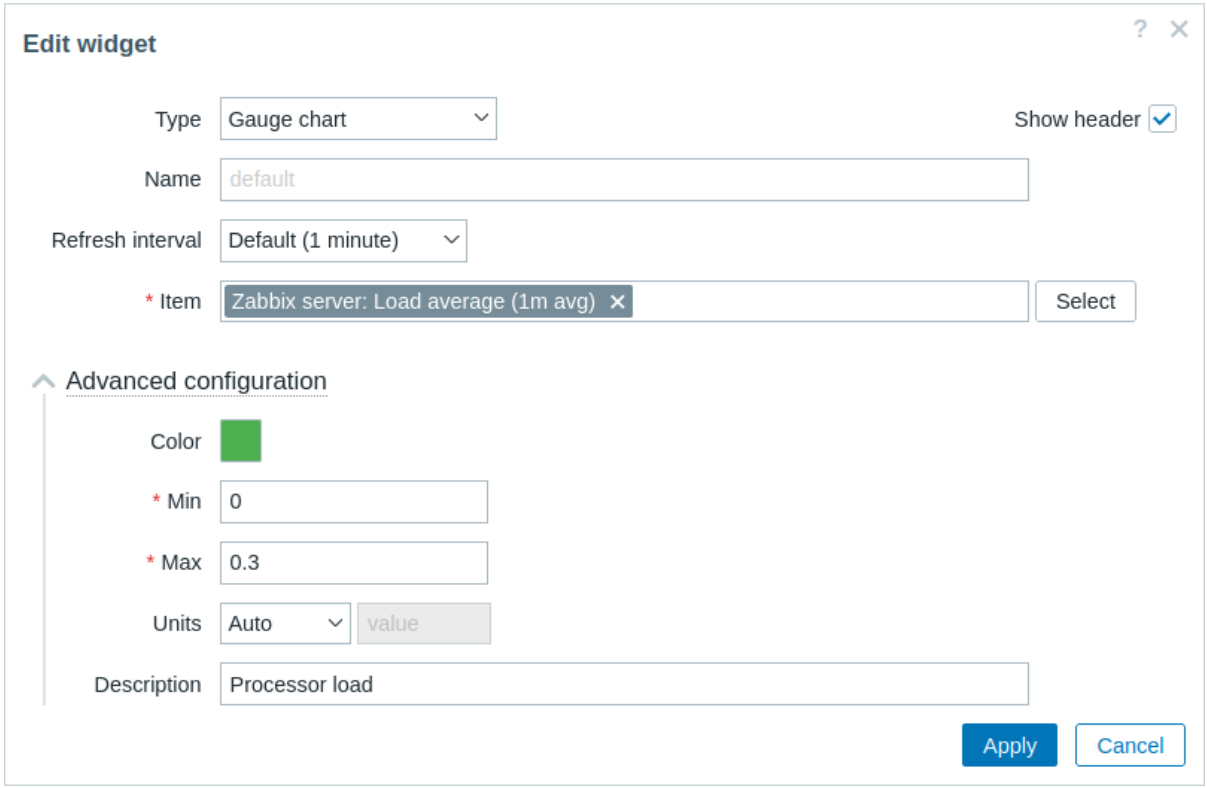
```
    updateForm() {
        this._unit_value.disabled = this._unit_select.value == <?= Widget::UNIT_AUTO ?>;
    }
};
```

2. Откройте *views/widget.edit.php* и добавьте файл *widget.edit.js.php* с JavaScript в представление конфигурации. Для этого используйте метод *includeJsFile()*. Чтобы добавить встроенный JavaScript, используйте метод *addJavaScript()*.

**ui/modules/lesson_gauge_chart/views/widget.edit.php**

```
<?php

/**
 * Gauge chart widget form view.
 *
 * @var CView $this
 * @var array $data
 */

use Modules\LessonGaugeChart\Includes\WidgetForm;

$lefty_units = new CWidgetFieldSelectView($data['fields']['value_units']);
$lefty_static_units = (new CWidgetFieldTextBoxView($data['fields']['value_static_units']))
    ->setPlaceholder(_('value'))
    ->setWidth(ZBX_TEXTAREA_TINY_WIDTH);

(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))
            ->setPopupParameter('numeric', true)
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Advanced configuration')))
            ->addField(
                new CWidgetFieldColorView($data['fields']['chart_color'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_min'])
            )
            ->addField(
                new CWidgetFieldNumericBoxView($data['fields']['value_max'])
            )
            ->addItem([
                $lefty_units->getLabel(),
                (new CFormField([
                    $lefty_units->getView()->addClass(ZBX_STYLE_FORM_INPUT_MARGIN),
                    $lefty_static_units->getView()
                ]))
            ])
            ->addField(
                new CWidgetFieldTextBoxView($data['fields']['description'])
            )
    )
    ->includeJsFile('widget.edit.js.php')
    ->addJavaScript('widget_lesson_gauge_chart_form.init('.json_encode([
        'color_palette' => WidgetForm::DEFAULT_COLOR_PALETTE
    ], JSON_THROW_ON_ERROR).');')
    ->show();
```

3. Вернитесь на панель, нажмите значок шестеренки в виджете, чтобы открыть форму конфигурации виджета. Теперь разверните раздел *Advanced configuration*, чтобы увидеть инициализированную палитру цветов. Заполните поля значениями и выберите цвет для диаграммы-датчика.




4. Нажмите *Apply* в форме конфигурации виджета. Затем нажмите *Save changes* в правом верхнем углу, чтобы сохранить панель.

5. Откройте *actions/WidgetView.php* и обновите контроллер.

Свойство **$this->fields_values** теперь содержит значения всех полей *Advanced configuration*. Завершите настройку контроллера, чтобы передавать конфигурацию и выбранное значение элемента данных в представление виджета.

**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. Откройте и измените *views/widget.view.php*.

Вам нужно создать контейнер для диаграммы-датчика, которую вы нарисуете на следующих шагах, и контейнер для описания.

Чтобы передать значения в JavaScript как объект JSON, используйте метод *setVar()*.

**ui/modules/lesson_gauge_chart/views/widget.view.php**

```
<?php

/**
 * Gauge chart widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem([
        (new CDiv())->addClass('chart'),
        $data['fields_values']['description']
            ? (new CDiv($data['fields_values']['description']))->addClass('description')
            : null
    ])
    ->setVar('history', $data['history'])
    ->setVar('fields_values', $data['fields_values'])
    ->show();
```

7. Создайте новый каталог *assets* в каталоге *lesson_gauge_chart*. Этот каталог будет использоваться для хранения JavaScript, CSS и, возможно, других ресурсов, таких как шрифты или изображения.
8. Для JavaScript представления виджета создайте каталог *js* в каталоге *assets*.
9. Создайте файл *class.widget.js* в каталоге *assets/js*.

Этот JavaScript-класс виджета будет расширять базовый JavaScript-класс всех виджетов панели - *CWidget*.

Панель зависит от корректной реализации виджета и передает ему всю необходимую информацию через вызов соответствующих JavaScript-методов. Панель также ожидает, что виджет будет генерировать события при возникновении определенного взаимодействия. Поэтому класс *CWidget* содержит набор методов с реализацией поведения виджета по умолчанию, которое можно настраивать путем расширения класса.

В данном случае требуется некоторая настройка, поэтому пользовательская логика будет реализована для следующего поведения виджета:

- инициализация виджета, отвечающая за определение начального состояния виджета (см. метод *onInitialize()*);
- отображение содержимого виджета (то есть отрисовка диаграммы-датчика), если процесс обновления виджета завершился успешно и без ошибок (см. метод *processUpdateResponse(response)* и связанные с ним методы *_resizeChart()* и *_updatedChart()*)
- изменение размера виджета (см. метод *onResize()* и связанный с ним метод *_resizeChart()*)

Для остальных аспектов виджета диаграммы-датчика будет использоваться реализация поведения виджета по умолчанию. Подробнее о JavaScript-методах класса *CWidget* см.: JavaScript.

Поскольку этот JavaScript требуется для представления виджета, он должен загружаться вместе со страницей панели. Чтобы включить загрузку JavaScript, необходимо обновить параметры *assets/js* и *js_class* в файле ***manifest.json***, как показано в шаге 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. Откройте *manifest.json* и добавьте:

- имя файла (*class.widget.js*) в массив в разделе *assets/js*;
- имя класса (*WidgetLessonGaugeChart*) в параметр *js_class* в разделе *widget*.

Теперь класс *WidgetLessonGaugeChart* будет автоматически загружаться вместе с панелью.

**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"]
    }
}
```

Добавление стилей CSS в виджет

В этом разделе вы узнаете, как добавлять собственные стили CSS, чтобы виджет выглядел более привлекательно.

1. Для стилей виджетов создайте новый каталог *css* в каталоге *assets*.
2. Создайте файл *widget.css* в каталоге *assets/css*. Чтобы стилизовать элементы виджета, используйте селектор *div.dashboard-widget-{widget id}*. Чтобы настроить CSS для всего виджета, используйте селектор *form.dashboard-widget-{idget 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;
}
```

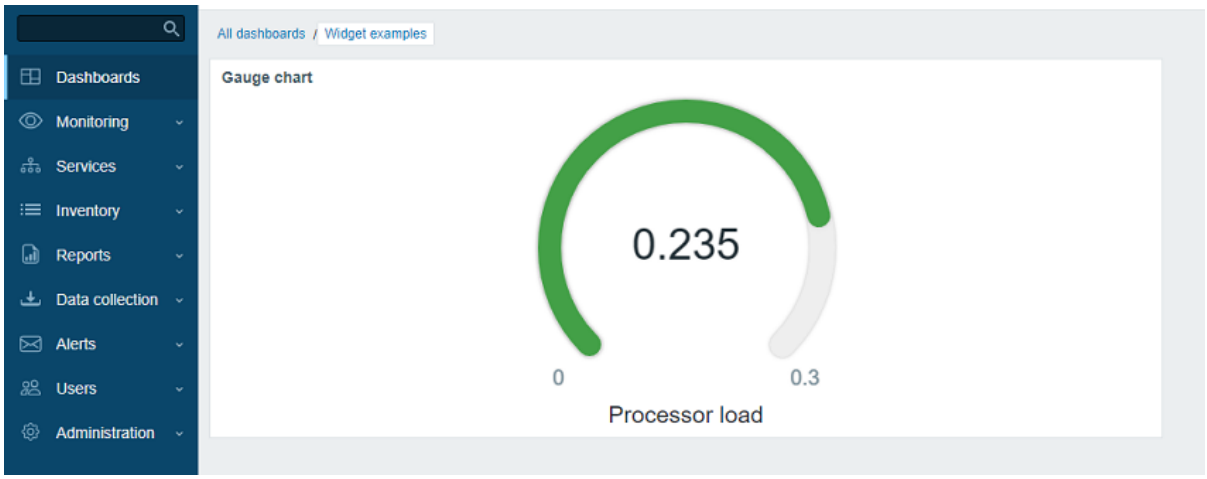
3. Откройте *manifest.json* и добавьте имя файла CSS (*widget.css*) в массив в разделе *assets/css*. Это позволит стилям CSS, определенным в *widget.css*, загружаться вместе со страницей панели.

**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. Обновите страницу панели, чтобы увидеть готовую версию виджета.




**Примеры**

В этом разделе представлены файлы образцов модулей и виджетов, которые вы можете использовать в качестве основы для своих собственных модулей.

Чтобы использовать модуль:

1. Загрузите ZIP-архив.
2. Распакуйте содержимое в отдельную папку внутри папки /zabbix/ui/modules.
3. Зарегистрируйте модуль в веб-интерфейсе Zabbix.

**Пример модуля**

- При создании группы хостов предоставьте разрешения на чтение настроенным группам пользователей - hg_auto_perm.zip

**Примеры виджетов**

- Минимальный виджет - widget_min.zip
- Виджет "Hello, world" только на CSS - hello_world_css.zip
- Виджет "Hello, world" только на JavaScript - hello_world_js.zip
- Виджет "Hello, world" на PHP - hello_world_php.zip

**Note:**
В качестве примеров вы также можете использовать нативные виджеты Zabbix.

## Библиотека Python для Zabbix

zabbix_utils — это официальная библиотека Python для Zabbix.

С помощью zabbix_utils вы можете:

- Использовать API Zabbix для управления объектами Zabbix (создавать узлы сети, обновлять элементы данных, получать события и т. д.).
- Собирать данные с агента Zabbix (аналогично Zabbix get).
- Отправлять данные на сервер Zabbix или прокси (аналогично Zabbix sender).

Библиотека поддерживает как синхронное, так и асинхронное выполнение скриптов, что позволяет использовать ее как для простых рабочих процессов, так и для крупномасштабных параллельных операций, таких как массовый сбор данных, пакетный экспорт, обработка оповещений и эффективное масштабирование системы.

Библиотека совместима с поддерживаемыми версиями сервера Zabbix и современными выпусками Python.

Чтобы начать работу, установите библиотеку, а затем следуйте руководству quickstart.

### Установка

zabbix_utils можно установить любым из способов, описанных ниже.

Требования:

- Zabbix 6.0 или новее (проверено на 6.0, 7.0, 7.2, 7.4)
- Python 3.8 или новее (проверено на 3.8–3.13)
- aiohttp — асинхронный HTTP-фреймворк (требуется только для асинхронного режима)

Python Package Index (PyPI)

Это наиболее распространенный способ для большинства сред:

```
pip install zabbix_utils
pip install zabbix_utils[async] #             async
```

Репозиторий Zabbix

Используйте этот способ, если вы предпочитаете управлять зависимостями через менеджер пакетов вашей системы.

Скачайте пакеты Zabbix для вашего дистрибутива Linux, а затем выполните следующие команды.

В RHEL и производных дистрибутивах:

```
dnf install python3-zabbix-utils
dnf install epel-release #             async
dnf install python3-aiohttp #             async
```

В Debian/Ubuntu и производных дистрибутивах:

```
apt install python3-zabbix-utils
apt install python3-aiohttp #             async
```

Исходный код (GitHub)

Используйте этот метод, если вы предпочитаете последнюю версию разработки:

```
git clone https://github.com/zabbix/python-zabbix-utils
cd python-zabbix-utils/
pip install -r requirements.txt #             async
python3 setup.py install
```

Чтобы установить зависимости, необходимые для асинхронного режима, вы также можете использовать один из методов, описанных выше.

**Руководство по быстрому началу работы**

После того как вы установили zabbix_utils, вы можете использовать его в своем скрипте.

Это руководство по быстрому началу работы показывает, как:

- Получить имя узла сети из агента Zabbix.
- Создать узел сети и элемент данных типа trapper с помощью API Zabbix.
- Отправить значение в элемент данных.

В руководстве скрипт приводится пошагово, с объяснением каждой части по мере ее появления. Полный скрипт приведен в конце страницы.

Также предполагается, что ваш сервер Zabbix, агент и API запущены локально.

Получение имени узла сети из агента Zabbix

Начните с получения имени узла сети, на котором запущен агент Zabbix. Это имя узла сети понадобится вам для создания узла сети с помощью API Zabbix.

1. Импортируйте класс `Getter` из zabbix_utils. Этот класс работает как Zabbix get и позволяет запрашивать данные у агента Zabbix.

2. Создайте экземпляр `Getter`, который подключается к локальному агенту Zabbix по адресу `127.0.0.1` на порту 10050.

3. Вызовите метод `get()` у экземпляра `Getter`, чтобы запросить имя узла сети у агента Zabbix.

- Метод `get()` принимает в качестве параметра ключ элемента данных и отправляет запрос агенту Zabbix для этого элемента данных.
- Метод `get()` возвращает объект, а имя узла сети хранится в атрибуте `value` этого объекта.
- `system.hostname` — это встроенный ключ агента Zabbix, который возвращает имя узла сети.

```
from zabbix_utils import Getter

agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value
```

Подключение и вход в Zabbix API

Далее подключитесь к Zabbix API. Это позволит вашему скрипту управлять объектами Zabbix, такими как узлы сети и элементы данных.

1. Импортируйте класс `ZabbixAPI` из zabbix_utils.

2. Создайте экземпляр `ZabbixAPI` и укажите URL вашего веб-интерфейса Zabbix.

3. Вызовите метод `login()` у экземпляра `ZabbixAPI` и укажите имя пользователя и пароль. У учетной записи должно быть разрешение на доступ к Zabbix API.

4. Вызовите метод `logout()`, чтобы закрыть сеанс после выполнения операций 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()
```

Создание узла сети в Zabbix

Теперь, когда у вас есть имя узла сети из Zabbix agent и вы подключены к Zabbix API, вы можете создать новый узел сети в Zabbix.

1. Вызовите метод API `host.create()` для экземпляра `ZabbixAPI` и укажите сведения об узле сети:

- `host` - задайте значение переменной `hostname`, содержащей имя узла сети, полученное из Zabbix agent.
- `interfaces` - содержит сведения о подключении для Zabbix agent, работающего на узле сети.
- `groups` - содержит как минимум одну группу узлов сети, к которой должен принадлежать узел сети.

2. Поскольку Zabbix API возвращает ID только что созданного узла сети, сохраните этот ID в переменной `host_id` для последующего использования.

3. Выведите сообщение для подтверждения создания узла сети.

```
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}")
```

Создание нового элемента данных в Zabbix

После создания узла сети вы можете добавить к нему элемент данных.

1. Определите уникальный ключ для создаваемого элемента данных.

2. Вызовите метод `item.create()` для экземпляра `ZabbixAPI` и передайте сведения об элементе данных:

- `hostid` - установите значение переменной `host_id`, содержащей ID созданного вами узла сети.
- `name` - имя элемента данных.
- `key_` - установите значение переменной `item_key`, которую вы только что определили.
- `type` - установите значение 2 (элемент данных trapper), требуется для получения значений, отправляемых из вашего скрипта на следующих шагах.
- `value_type` - установите значение 3 (numeric unsigned), тип данных, которые хранит этот элемент данных.

3. Поскольку Zabbix API возвращает ID вновь созданного элемента данных, сохраните этот ID в переменной `item_id` для последующего использования.

4. Выведите сообщение для подтверждения создания элемента данных.

```
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}")
```

Отправка значения на узел сети

Теперь, когда вы создали узел сети и элемент данных, вы можете отправить на него данные.

1. Импортируйте класс `time` и подождите несколько секунд перед отправкой данных.
Это гарантирует, что Zabbix полностью обработал ваш новый узел сети и элемент данных.

2. Импортируйте класс `Sender` из `zabbix_utils`.
Этот класс работает как Zabbix sender и позволяет вашему скрипту отправлять данные в Zabbix.

3. Создайте экземпляр `Sender`, который подключается к локальному серверу Zabbix по адресу `127.0.0.1` на порту `10051`.

4. Вызовите метод `send_value()` у экземпляра `Sender`, чтобы отправить значение в элемент данных вашего узла сети, и укажите следующие параметры:

- `hostname` - установите значение переменной `hostname`, содержащей имя узла сети, полученное от агента Zabbix.
- `item_key` - установите значение переменной `item_key`, которую вы только что определили ранее в скрипте.
- `1` - значение для отправки.

4. Выведите сообщение, чтобы подтвердить, что значение было отправлено.

```
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}")
```

После успешного выполнения этого кода вы должны увидеть сообщение об успешной отправке, указывающее, что Zabbix получил ваше значение:

```
Sender response: {"processed": 1, "failed": 0, "total": 1, "time": "0.000151", "chunk": 1}
```

Теперь вы также можете проверить ваш веб-интерфейс Zabbix (*Monitoring* > *Latest data*) и увидеть это значение.

Полный скрипт

Ниже приведен полный скрипт, который объединяет все шаги: получение имени узла сети, создание узла сети и элемента данных с использованием Zabbix API, а также отправку данных в Zabbix.

```
import time
from zabbix_utils import Getter, ZabbixAPI, Sender

###                      Zabbix
agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value

###                    Zabbix API
api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

###              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"       '{hostname}'       ID {host_id}")

###                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_key}'       ID {item_id}")

###       API
api.logout()

###      ,     Zabbix
time.sleep(10)

###
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value(hostname, item_key, 1)

print(f"    Sender: {response}")
```

**Использование Zabbix API**

zabbix_utils позволяет использовать Zabbix API для управления объектами Zabbix, включая создание узлов сети, обновление элементов данных, получение событий и многое другое.

Запросы API можно выполнять в синхронном или асинхронном режиме:

- В синхронном режиме ваш скрипт Python отправляет запрос и ожидает ответа, прежде чем продолжить выполнение, что подходит для простых, последовательных и предсказуемых операций.
- В асинхронном режиме скрипт отправляет запросы, не ожидая каждого ответа, что позволяет выполнять другие операции параллельно; это эффективнее для медленных запросов или больших пакетов данных.

Примеры на этой странице ориентированы на синхронный режим, хотя асинхронный режим использует похожие подходы. Дополнительные примеры доступны в репозитории GitHub zabbix_utils.

Импорт

Чтобы использовать zabbix_utils для работы с Zabbix API, импортируйте класс ZabbixAPI в свой скрипт:

```
from zabbix_utils import ZabbixAPI
```

> **Note:**
> Если вы уже используете библиотеку Python из сообщества, обычно можно заменить этот импорт на ZabbixAPI из zabbix_utils.

Вход в систему

Перед выполнением запросов к API необходимо создать экземпляр ZabbixAPI и выполнить вход в Zabbix API.

Выберите метод, который лучше всего соответствует тому, как вы управляете учетными данными. Вы можете использовать имя пользователя и пароль или API tokens.

Во время инициализации

При создании экземпляра ZabbixAPI вы можете сразу указать URL веб-интерфейса Zabbix и свои учетные данные:

```
###                :
api = ZabbixAPI(url="127.0.0.1/zabbix", user="Admin", password="zabbix")

###     API:
api = ZabbixAPI(url="127.0.0.1/zabbix", token="your_api_token")
```

Если вы предпочитаете группировать параметры подключения, вы можете передать их в виде словаря Python и распаковать его с помощью синтаксиса именованных аргументов Python:

```
ZABBIX_AUTH = {
    "url": "127.0.0.1/zabbix",
    "user": "Admin",
```

```
    "password": "zabbix"
}

api = ZabbixAPI(**ZABBIX_AUTH)
```

Использование `login()`

Сначала можно создать экземпляр ZabbixAPI, указав только URL веб-интерфейса Zabbix, а затем использовать метод `login()` со своими учетными данными:

```
###                 :
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

###     API:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(token="your_api_token")
```

Использование переменных среды

Сначала вы можете сохранить свои учетные данные в переменных среды, например, с помощью терминала:

```
###                 :
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"

###    :
export ZABBIX_TOKEN="your_api_token"
```

Затем, если вы запустите свой скрипт из той же сессии терминала, экземпляру ZabbixAPI в скрипте потребуется только URL веб-интерфейса Zabbix:

```
api = ZabbixAPI(url="127.0.0.1/zabbix")
```

Вы также можете сохранить URL в переменной среды:

```
###                 :
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"
export ZABBIX_URL="https://127.0.0.1/zabbix"

###    :
export ZABBIX_TOKEN="your_api_token"
export ZABBIX_URL="https://127.0.0.1/zabbix"
```

Когда URL и учетные данные заданы как переменные среды, экземпляру ZabbixAPI в скрипте не требуется никаких аргументов:

```
api = ZabbixAPI()
```

Выход из системы

Если вы вошли в систему, используя имя пользователя и пароль, вызовите метод `logout()` после завершения операций с API:

```
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:**
> При использовании токенов API вызывать метод `logout()` не требуется.

API-запросы

После входа в систему вы можете выполнять любые API-запросы, вызывая методы, описанные в справочнике методов Zabbix API.

Методы API вызываются в следующем формате:

```
api_instance.zabbix_object.method(parameters)
```

> **Note:**
> Некоторые имена методов или объектов Zabbix API используют слова, которые являются зарезервированными ключевыми словами в Python (например, `import`). Чтобы избежать ошибок Python, добавляйте символ подчеркивания к имени метода или объекта при использовании его в Python (например, `api.configuration.import_`).

Например, чтобы получить список узлов сети с помощью `host.get`:

```
### 1.         ZabbixAPI    zabbix_utils:
from zabbix_utils import ZabbixAPI

### 2.             ZabbixAPI           :
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### 3.            ,               :
hosts = api.host.get(
    filter={
        "host": [
            "Zabbix server",
            "Linux server"
        ]
    }
)

### 4.                                        :
for host in hosts:
    print(host)

### 5.        API,             :
api.logout()
```

Асинхронный режим

Асинхронный режим позволяет вашему скрипту отправлять API-запросы, не дожидаясь завершения каждого из них. Это может сделать ваш скрипт более эффективным, когда ему нужно выполнить много API-запросов или когда некоторые запросы занимают много времени.

При использовании асинхронного режима есть важные отличия по сравнению с синхронным режимом:

- Импортируйте модуль Python `asyncio` (сначала необходимо установить требуемые зависимости).
- Импортируйте `AsyncZabbixAPI` вместо `ZabbixAPI`.
- Пишите код внутри функции `async`.
- Используйте `await` при вызове методов API, включая `login()` и `logout()`.

> **Attention:**
> При создании экземпляра `AsyncZabbixAPI` вы не можете выполнить вход во время инициализации.

Например, чтобы увидеть список узлов сети с помощью `host.get` в асинхронном режиме:

```
### 1. Import asyncio for asynchronous mode, and AsyncZabbixAPI from zabbix_utils:
import asyncio
from zabbix_utils import AsyncZabbixAPI

### 2. Define the main async function where all API operations will run:
async def main():

    # 3. Create an AsyncZabbixAPI instance and log in (must await):
    api = AsyncZabbixAPI(url="127.0.0.1")
```

```
    await api.login(user="User", password="zabbix")

    # 4. Retrieve hosts matching a filter (must await):
    hosts = await api.host.get(
        filter={
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    )

    # 5. Iterate over returned hosts and print each host's details:
    for host in hosts:
        print(host)

    # 6. Log out from the API to close the session (must await):
    await api.logout()

### 7. Run the async main() function using asyncio's event loop:
asyncio.run(main())
```

Примеры

Следующие примеры показывают распространенные задачи Zabbix API с использованием библиотеки zabbix_utils.

Дополнительные примеры доступны в каталоге `examples` репозитория zabbix_utils на GitHub.

**Сбор данных с агента Zabbix**

Обзор

zabbix_utils позволяет собирать данные с агента Zabbix (аналогично Zabbix get).

Данные можно собирать в синхронном или асинхронном режиме:

- В синхронном режиме ваш скрипт Python запрашивает данные и ожидает их получения, прежде чем продолжить выполнение, что подходит для простых, последовательных и предсказуемых операций.
- В асинхронном режиме скрипт запрашивает данные, не ожидая каждого ответа, что позволяет выполнять другие операции параллельно; это более эффективно для медленных запросов или больших пакетов данных.

Примеры на этой странице ориентированы на синхронный режим, хотя асинхронный режим использует похожие подходы. Дополнительные примеры доступны в репозитории GitHub zabbix_utils.

Импорт

Чтобы использовать zabbix_utils для сбора значений элементов данных, импортируйте класс `Getter` в ваш Python-скрипт:

```
from zabbix_utils import Getter
```

Данные запроса

Чтобы запросить значение элемента данных:

1. Создайте экземпляр `Getter`, указав IP-адрес и порт вашего агента Zabbix.
2. Вызовите метод `get()` у экземпляра `Getter`, указав ключ элемента данных, который нужно получить.

Например, чтобы запросить данные для элемента данных `system.uname`:

```
agent = Getter(host='192.0.2.1', port=10050)
response = agent.get('system.uname')
```

Использование IP-адреса, отличного от значения по умолчанию

Если сервер, на котором выполняется ваш скрипт, имеет несколько IP-адресов, вы можете указать `source_ip`, который `Getter` будет использовать при подключении к агенту Zabbix:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    source_ip='10.10.7.1'
)
```

Использование тайм-аута

Вы можете задать `timeout` ответа для `Getter`, чтобы контролировать, как долго ваш скрипт должен ждать ответа от агента Zabbix, прежде чем прекратить ожидание:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    timeout=30
)
```

Использование шифрования

`Getter` не включает встроенную поддержку шифрования, но вы можете добавить ее, создав обертку с использованием сторонних библиотек:

```
def psk_wrapper(sock, tls):
    # ...
    #            TLS PSK
    # ...

agent = Getter(
    host='192.0.2.1',
    port=10050,
    socket_wrapper=psk_wrapper
)
```

Ответ

Ответ от агента Zabbix обрабатывается библиотекой и возвращается как объект `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
```

Асинхронный режим

Асинхронный режим позволяет вашему скрипту собирать значения, не дожидаясь получения каждого из них. Это может сделать ваш скрипт более эффективным, когда ему нужно собрать много значений или когда получение некоторых значений занимает много времени.

При использовании асинхронного режима есть важные отличия по сравнению с синхронным режимом:

- Импортируйте модуль Python `asyncio` (сначала необходимо установить требуемые зависимости).
- Импортируйте `AsyncGetter` вместо `Getter`.
- Пишите код внутри функции `async`.
- Используйте `await` при вызове метода `get()`.

Например, чтобы собрать одно значение с использованием асинхронного режима:

```
### 1.        asyncio                AsyncGetter   zabbix_utils:
import asyncio
from zabbix_utils import AsyncGetter
```

```
### 2.                          ,                              :
async def main():
    agent = AsyncGetter(host='192.0.2.1', port=10050)

    # 3.            system.uname        Zabbix (            await):
    response = await agent.get('system.uname')

    # 4.           ,              Zabbix:
    print(response.value)

### 5.                   main()                    asyncio:
asyncio.run(main())
```

**Отправка данных на сервер Zabbix или прокси**

zabbix_utils позволяет отправлять значения элементов данных на элемент данных типа trapper на сервер Zabbix или прокси (аналогично Zabbix sender).

Вы можете отправить одно значение, несколько значений или даже на несколько кластеров Zabbix.

Данные можно отправлять в синхронном или асинхронном режиме:

- В синхронном режиме ваш скрипт Python отправляет значения и ожидает ответа перед продолжением; это подходит для простых, последовательных и предсказуемых операций.
- В асинхронном режиме скрипт отправляет значения, не ожидая каждого ответа, что позволяет выполнять другие операции параллельно; это эффективнее для медленных запросов или больших пакетов данных.

Примеры на этой странице сосредоточены на синхронном режиме, хотя асинхронный режим использует похожие подходы. Дополнительные примеры доступны в репозитории GitHub zabbix_utils.

Импорт

Чтобы использовать zabbix_utils для отправки значений элементов данных, импортируйте класс `Sender` в свой скрипт:

```
from zabbix_utils import Sender
```

Для отправки нескольких значений вы также можете импортировать класс `ItemValue`:

```
from zabbix_utils import Sender, ItemValue
```

Отправка одного значения

Чтобы отправить значение элемента данных:

1. Создайте экземпляр `Sender`, указав IP-адрес и порт вашего сервера Zabbix или прокси.
2. Вызовите метод `send_value()` у экземпляра `Sender` в следующем формате:

```
sender_instance.send_value('host', 'item.key', 'value', optional_timestamp, optional_nanoseconds)
```

Например, чтобы отправить 1 в элемент данных trapper `service.status` на узел сети `Linux server`:

```
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value('Linux server', 'service.status', 1)
```

Использование IP-адреса, отличного от значения по умолчанию

Если сервер, на котором выполняется ваш скрипт, имеет несколько IP-адресов, вы можете указать `source_ip`, который `Sender` будет использовать при отправке значений на сервер Zabbix или прокси:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    source_ip='10.10.7.1'
)
```

Использование тайм-аута

Вы можете задать `timeout` ответа для `Sender`, чтобы контролировать, как долго ваш скрипт должен ждать ответа от сервера Zabbix или прокси, прежде чем прекратить ожидание:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    timeout=30
)
```

Использование файла конфигурации агента

Вы можете разрешить zabbix_utils считывать параметры Server или ServerActive из локального файла конфигурации Zabbix агента или агента 2. В таких случаях при создании экземпляра Sender не нужно указывать параметры подключения:

```
sender = Sender(
    use_config=True,
    config_path='/etc/zabbix/zabbix_agent2.conf'
)
```

**Attention:**
Если ServerActive содержит один или несколько кластеров Zabbix с несколькими экземплярами сервера, Sender отправляет данные на первый доступный сервер в каждом кластере. Если ServerActive не задан, используется адрес из Server с портом по умолчанию (10051).

Использование шифрования

Sender не включает встроенную поддержку шифрования, но вы можете добавить ее, создав обертку с использованием сторонних библиотек:

```
def psk_wrapper(sock, tls):
    # ...
    #             TLS PSK
    # ...

sender = Sender(
    server='127.0.0.1',
    port=10051,
    socket_wrapper=psk_wrapper
)
```

Ответ для одного значения

Ответ, возвращаемый сервером или прокси Zabbix, обрабатывается библиотекой и возвращается как объект 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
```

Асинхронный режим

Асинхронный режим позволяет вашему Python-скрипту отправлять значения, не дожидаясь ответа от сервера Zabbix или прокси. Это может сделать ваш скрипт более эффективным, когда ему нужно отправлять много значений или когда на отправку некоторых значений уходит много времени.

При использовании асинхронного режима есть несколько важных отличий по сравнению с синхронным режимом:

- Импортируйте модуль Python asyncio (сначала необходимо установить требуемые зависимости).
- Импортируйте AsyncSender вместо Sender.
- Пишите код внутри асинхронной функции async.
- Используйте await при вызове метода send_value().

Например, чтобы отправить одно значение с использованием асинхронного режима:

```
### 1.        asyncio                AsyncSender   zabbix_utils:
import asyncio
from zabbix_utils import AsyncSender

### 2.                        ,                                    (               await):
async def main():
    sender = AsyncSender(server='127.0.0.1', port=10051)
    response = await sender.send_value('Linux server', 'service.status', 1)

    # 3.         ,             Zabbix       :
    print(response)

### 4.                  main()               asyncio:
asyncio.run(main())
```

Отправка нескольких значений

Чтобы отправить несколько значений:

1. Подготовьте массив объектов `ItemValue`, каждый из которых использует тот же формат, что и метод `send_value()`.
2. Создайте экземпляр `Sender`, указав IP-адрес и порт вашего сервера Zabbix или прокси.
3. Вызовите метод `send()` (вместо `send_value()`) для экземпляра `Sender`, указав массив объектов со значениями для отправки.

Например, чтобы отправить пять значений на разные узлы сети:

```
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)
```

Использование пользовательского размера фрагмента

Если вам нужно отправить больше значений, чем элемент данных типа trapper может принять за один запрос, вы можете разделить их на фрагменты.

По умолчанию размер фрагмента составляет 250 значений. Вы можете изменить его, задав параметр `chunk_size` при создании экземпляра `Sender`.

Например, чтобы отправить пять значений в трех фрагментах (2-2-1), задайте параметр `chunk_size` равным 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)
```

Отправка значений в несколько кластеров Zabbix

Чтобы отправить значения в несколько кластеров Zabbix:

1. Подготовьте массив кластеров Zabbix. Если у кластера несколько узлов, значение будет отправлено на первый **доступный** узел каждого кластера.
2. Создайте `Sender`, указав ваш массив кластеров Zabbix.
3. Вызовите метод `send_value()` для экземпляра `Sender`, используя тот же формат, что и метод `send_value()`.

Например, чтобы отправить значение на первый доступный узел в каждом кластере:

```
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)
```

Ответ для нескольких значений

По умолчанию Sender возвращает агрегированный результат отправки значений по всем узлам сети или кластерам:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}
```

Если вам нужна более подробная информация, вы можете просмотреть результаты для каждого кластера и каждого фрагмента с помощью атрибута `response.details`:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}

if response.failed == 0:
    print(f"                  {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"      {resp.processed}   {resp.total}   {node.address}:{node.port}")
            #       1   1   127.0.0.1:10051
            #       1   1   zabbix.example.local:10051
```

**Отладочное логирование**

Чтобы устранять неполадки при работе с Zabbix API, сборе данных с агента Zabbix или отправке данных на сервер Zabbix или прокси, можно включить отладочное логирование.

Библиотека zabbix_utils использует стандартный модуль Python `logging`.

Чтобы видеть подробные отладочные сообщения из библиотеки:

1. Импортируйте модуль Python `logging` в свой скрипт.
2. Настройте систему логирования так, чтобы сообщения выводились в удобочитаемом формате, и установите уровень логирования 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)
```

При включенном отладочном логировании подробная информация о запросах и ответах выводится, например, так:

```
[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
```

## Плагины

**Обзор** Пользовательские загружаемые плагины расширяют функциональность Zabbix agent 2. Они компилируются отдельно, но используют общий пакет с Zabbix agent 2.

Каждый плагин является *Go*-пакетом, который определяет структуру и реализует один или несколько интерфейсов плагина (*Exporter*, *Configurator*, *Runner*).

Перейти к:

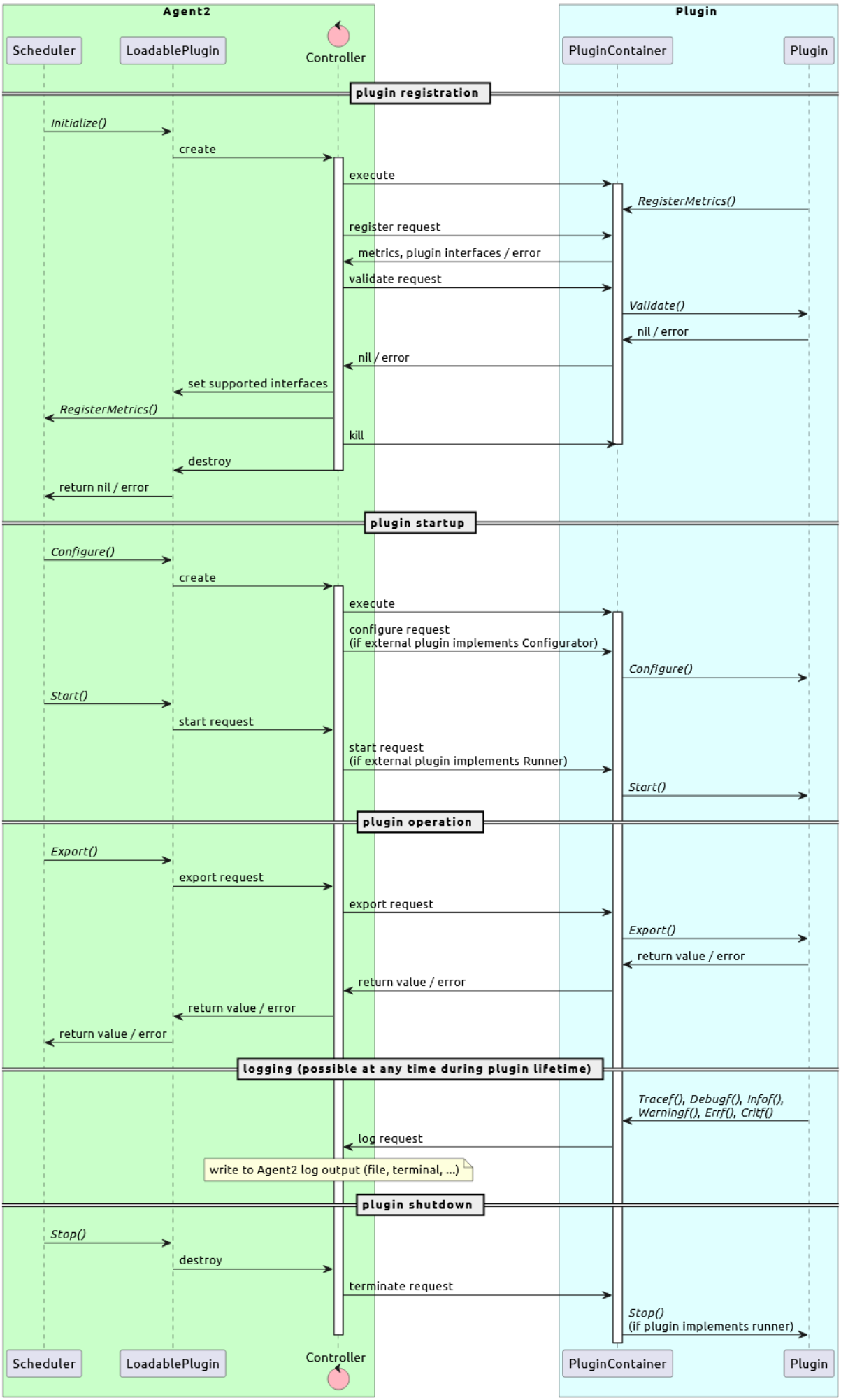
- Написание вашего первого плагина
- Интерфейсы плагинов

См. также:

- Пример плагина для Zabbix agent 2

**Диаграмма подключения** Zabbix agent 2 подключается к плагинам двунаправленно, используя UNIX-сокеты на Linux и именованные каналы (Named Pipes) на Windows.

Диаграмма подключения ниже иллюстрирует процесс связи между Zabbix agent 2 и загружаемым плагином, а также процесс сбора метрик.


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

#### Примеры

В качестве ориентира вы можете использовать несколько пустых примеров, а также существующие загружаемые плагины, предоставляемые Zabbix:

- Пример плагина для агента Zabbix 2
- Плагин MSSQL
- Плагин MongoDB
- Плагин PostgreSQL

См. полный список плагинов, предоставляемых Zabbix.

#### Интерфейсы плагинов

В этом разделе описаны доступные интерфейсы плагинов.

**plugin.Exporter** *Exporter* - это простейший интерфейс, который выполняет опрос и возвращает значение (значения), ничего или ошибку. Он принимает подготовленный ключ элемента, параметры и контекст. Доступ ко всем остальным интерфейсам плагинов является эксклюзивным, и ни один метод не может быть вызван, если плагин уже выполняет задачу. Кроме того, существует ограничение в 100 одновременных вызовов *Export()* на каждый плагин, которое можно уменьшить в соответствии с требованиями каждого плагина.

**plugin.Configurator** Интерфейс *Configurator* предоставляет параметры конфигурации плагина из файлов конфигурации Zabbix агента 2.

**plugin.Runner** Интерфейс *Runner* предоставляет средства для выполнения инициализации при запуске (активации) плагина и деинициализации при остановке (деактивации) плагина. Например, плагин может запускать/останавливать некоторую фоновую *goroutine*, реализуя интерфейс Runner.

#### Создание плагина (руководство)

Это пошаговое руководство по созданию простого загружаемого плагина для Zabbix Agent 2.

Вы также можете использовать репозиторий примеров в качестве шаблона или руководства для создания собственных плагинов.

**Что вы создадите** В этом руководстве показано, как создать новый загружаемый плагин **MyIP**. Плагин будет реализовывать один ключ элемента, **myip**, который возвращает внешний IP-адрес хоста, на котором запущен Zabbix агент 2.

##### Шаг 1: Настройка

1. Плагин — это стандартный модуль Go.

Начните с инициализации файла `go.mod` в каталоге плагина для отслеживания зависимостей плагина:

```
cd path/to/plugins/myip #
go mod init myip
```

2. Установите обязательную зависимость Zabbix Go SDK (`golang.zabbix.com/sdk`):

```
go get golang.zabbix.com/sdk@$LATEST_COMMIT_HASH
```

Замените `$LATEST_COMMIT_HASH` на последний хеш коммита `HEAD` из `golang.zabbix.com/sdk` репозитория в соответствующей ветке релиза. Например:

```
go get golang.zabbix.com/sdk@af85407
```

Обратите внимание, что версионирование `golang.zabbix.com/sdk` в настоящее время не поддерживается, но это может измениться в будущем.

При необходимости дополнительные зависимости можно установить с помощью `go get`.

3. Создайте пустой файл `main.go` для исходного кода плагина:

```
touch main.go
```

Теперь начальная настройка завершена, и плагин готов к разработке.

**Шаг 2: Структура плагина** Модуль `golang.zabbix.com/sdk`, установленный на предыдущем шаге, предоставляет необходимую основу для разработки плагинов и обеспечивает единообразную структуру всех плагинов.

1. Настройте базовый поток выполнения.

Начните с определения основного потока выполнения плагина. Добавьте следующий код в `main.go`:

```
package main

func main() {
err := run()
if err != nil {
panic(err)
}
}

func run() error {
return nil
}
```

Это задает базовый поток выполнения для плагина. Функция run позже будет содержать основную логику плагина.

2. Изучите интерфейсы плагина.

Плагин агента Zabbix 2 должен быть представлен структурой, которая реализует интерфейсы из пакета `golang.zabbix.com/sdk/p`

- *Accessor* - определяет основные методы, которые должны реализовывать все плагины, например задание имени плагина и обработку тайм-аутов ключей элементов данных.
- Один или несколько из следующих функциональных интерфейсов плагина:
- *Exporter* - выполняет опрос и возвращает значение (или значения), ничего или ошибку; часто используется вместе с интерфейсом *Collector*.
- *Collector* - управляет периодическим сбором данных.
- *Runner* - определяет процедуры запуска и завершения работы плагина.
- *Watcher* - позволяет реализовать независимый опрос метрик, обходя внутренний планировщик агента; полезно для мониторинга на основе ловушек или событий.
- *Configurator* - определяет, как плагин читает и применяет свои параметры конфигурации.

Вы можете реализовать эти интерфейсы самостоятельно или использовать стандартные, предоставляемые Zabbix Go SDK, при необходимости изменяя их. В этом руководстве используются стандартные реализации.

3. Создайте структуру плагина.

Теперь импортируйте пакет *plugin* и создайте структуру `myIP`, которая включает структуру `plugin.Base`:

```
import "golang.zabbix.com/sdk/plugin"

type myIP struct {
plugin.Base
}
```

Структура myIP в настоящее время удовлетворяет интерфейсу Accessor. Метод для реализации одного из функциональных интерфейсов плагина, `Exporter`, будет добавлен позже в руководстве.

**Шаг 3: Определите ключи элементов данных** Вашему плагину нужны ключи элементов данных, чтобы собирать данные и передавать их серверу Zabbix или прокси.

1. Импортируйте пакет *errs* для обработки ошибок:

```
import "golang.zabbix.com/sdk/errs"
```

2. Зарегистрируйте ключи элементов данных с помощью функции `plugin.RegisterMetrics()` внутри функции `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
}
```

Чтобы зарегистрировать несколько ключей элементов данных, повторите параметры *metric name* и *description* для каждой метрики. Например:

```
plugin.RegisterMetrics(&impl, "Myip", "metric.one", "Metric one description.", "metric.two", "Metric two d
```

**Шаг 4: Настройка обработчика** Обработчик обеспечивает связь между агентом и плагином.

1. Импортируйте пакет *container*:

```
import "golang.zabbix.com/sdk/plugin/container"
```

2. Внутри функции `run()` добавьте код для создания и настройки обработчика:

```
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
}
```

**Шаг 5: Реализация сбора данных** Сбор данных выполняется через интерфейс Exporter, который описывает метод Export:

```
func Export(
  key string,             //                    .
  params []string,        //      ,                       (`myip[arg1, arg2]`).
  context ContextProvider //                          .
) (any, error)
```

1. Импортируйте необходимые пакеты для HTTP-запросов и чтения ответа:

```
import (
"io"
"net/http"
)
```

2. Реализуйте метод Export для структуры myIP:

```
func (p *myIP) Export(
key string, params []string, context plugin.ContextProvider,
) (any, error) {
//                                                  `key`.
    //                    ,          `key`         .
if key != "myip" {
return nil, errs.Errorf("unknown item key %q", key)
}

//                           2.
p.Infof(
"received request to handle %q key with %d parameters",
key,
len(params),
)

//                  .

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
}
```

**Шаг 6: Сборка и настройка плагина**

1. Чтобы собрать плагин, выполните:

```
go mod tidy
go build
```

Это должно создать исполняемый файл myip в текущем каталоге.

2. Настройте агент Zabbix 2 для использования плагина:

```
echo "Plugins.MyIP.System.Path=$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE" > /etc/zabbix_agent2.d/plugins.d/myip.
```

Замените $PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE на путь к myip, созданному на шаге 5.

Имя плагина в имени параметра конфигурации (*MyIP* в этом руководстве) должно совпадать с именем плагина, определенным в функции *plugin.RegisterMetrics()*.

3. Чтобы протестировать плагин и его элемент данных `myip`, выполните:

```
zabbix_agent2 -c /etc/zabbix_agent2.conf -t myip
```

Вывод должен содержать внешний IP-адрес вашего узла сети и выглядеть примерно так:

```
myip                                          [s|192.0.2.1]
```

На этом вы создали простой загружаемый плагин для агента Zabbix 2. Поздравляем!

```
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{}

//                       `myip`.
err := plugin.RegisterMetrics(
p,
"MyIP",                           //
"myip",                           //
"       IP-          .", //
)
if err != nil {
return errs.Wrap(err, "                    ")
}

//               .
h, err := container.NewHandler("MyIP") //
if err != nil {
return errs.Wrap(err, "                     ")
}

//                                          .
//          p.Logger.Infof, p.Logger.Debugf    .  .
p.Logger = h

//                .
//                                  .
err = h.Execute()
if err != nil {
return errs.Wrap(err, "                       ")
}
```

```
return nil
}

func (p *myIP) Export(
key string, params []string, context plugin.ContextProvider,
) (any, error) {
//                                                    `key`.
//                    ,          `key`         .
if key != "myip" {
return nil, errs.Errorf("unknown item key %q", key)
}

//                            2.
p.Infof(
"                     %q   %d       ",
key,
len(params),
)

//                   .

resp, err := http.Get("https://api.ipify.org")
if err != nil {
return nil, errs.Wrap(err, "            IP-   ")
}

defer resp.Body.Close()

body, err := io.ReadAll(resp.Body)
if err != nil {
return nil, errs.Wrap(err, "                    ")
}

return string(body), nil
}
```

**Полный исходный код**

### Изменения в разработке расширений

На этой странице перечислены все изменения, если таковые имеются, в разработке пользовательских расширений Zabbix.

## Страницы помощи Zabbix

Справочная документация (manpages) по инструментам и службам командной строки Zabbix, содержащая список доступных команд, параметры и примеры использования.

### zabbix_agent2

```
ZABBIX_AGENT2(8)                                          ZABBIX_AGENT2(8)


       zabbix_agent2 - Zabbix agent 2


       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
```

zabbix_agent2 —
.

-c, --config config-file
config-file .

-R, --runtime-control runtime-option
runtime-option.

userparameter_reload

log_level_increase

log_level_decrease

help

metrics

version

-p, --print
.
,
.
.
,
. t,
. , ,
Zabbix
server zabbix_get,
. :

d .

m . ,
,
, ,
.

" ".

s . .

t . , s.

```
       u                .

   -t, --test item-key
                                  .  . --print
                  .

   -v, --verbose

                    .

   -T, --test-config
                                       .

   -h, --help
                                    .

   -V, --version
                                       .


   /usr/local/etc/zabbix_agent2.conf
                              Zabbix agent 2
          (                      ).

.


   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)                                              ZABBIX_AGENTD(8)


   zabbix_agentd -         Zabbix


   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


   zabbix_agentd —                                    .


   -c, --config config-file
                          config-file                .

   -f, --foreground
                   Zabbix          .

   -R, --runtime-control runtime-option
                                         runtime-option.
```

userparameter_reload

log_level_increase[=target]
, , target

log_level_decrease[=target]
, , target

process-type
(active checks, collector, listener)

process-type,N
( , listener,3)

pid , 65535.
target "process-type,N"

-p, --print
.
,
.
.

, .
t,
. , ,
Zabbix zabbix_get
,
. :

d .

m . ,
,
, ,
.

" ".

s . .

t . , s.

u .

-t, --test item-key
. . --print .

-T, --test-config
.

-h, --help
.

-V, --version
.

/usr/local/etc/zabbix_agentd.conf
Zabbix (

).

.

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) ZABBIX_GET(1)

zabbix_get - zabbix_get

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-subject 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

zabbix_get —
Zabbix.

-s, --host host-name-or-IP
IP- .

-p, --port port-number
, .
10050.

-I, --source-address IP-address
IP- .

-t, --timeout seconds
. : 1-600 (
: 30)

-k, --key item-key
, .

-P, --protocol value
, . :

auto JSON- ,

( )

            json                          JSON-

            plaintext
                              ,
                                    (    6.4.x     )

--tls-connect value
                        .     :

            unencrypted
                                   (       )

            psk                           TLS


            cert                          TLS

--tls-ca-file CA-file
                  ,                   CA
                     .

--tls-crl-file CRL-file
                  ,                    .

--tls-agent-cert-issuer cert-issuer
                              .

--tls-agent-cert-subject cert-subject
                             .

--tls-cert-file cert-file
                  ,
              .

--tls-key-file key-file
                  ,               .

--tls-psk-identity PSK-identity
                   PSK.

--tls-psk-file PSK-file
                  ,                         .

--tls-cipher13 cipher-string
                  OpenSSL 1.1.1          TLS 1.3.
                                .               ,
              OpenSSL    1.1.1.

--tls-cipher cipher-string
                  GnuTLS (   TLS 1.2     )
       OpenSSL (       TLS 1.2).
                  .

-h, --help
                        .

-V, --version
                           .


            0,    zabbix_get
    Zabbix.                    ,             1.

```
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
```

.

```
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)                                                  ZABBIX_JS(1)
```

```
zabbix_js -      Zabbix JS
```

```
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
```

```
zabbix_js  —                    ,
                       .
```

```
-s, --script script-file
                                  .
                    '-',                  stdin.
```

```
-p, --param input-param
                         ,             .

               'value'.             -i.
```

```
-i, --input input-file
                ,                ,
            .
                                'value'.
                    '-',                stdin.
                   -p.
```

```
-w, --webdriver url
              URL webdriver.
```

```
-l, --loglevel log-level
                        (    : 0-5).
```

```
-t, --timeout timeout
```

```
                -        .           : 1-600
        (       : 10)

    -h, --help
                                    .

    -V, --version
                                    .


    zabbix_js -s script-file.js -p example

.
    Documentation

    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)                                                 ZABBIX_PROXY(8)


    zabbix_proxy -         Zabbix


    zabbix_proxy [-c config-file]
    zabbix_proxy [-c config-file] -R runtime-option
    zabbix_proxy [-c config-file] -T
    zabbix_proxy -h
    zabbix_proxy -V


    zabbix_proxy —       ,
                  Zabbix.


    -c, --config config-file
                         config-file               .

    -f, --foreground
                    Zabbix          .

    -R, --runtime-control runtime-option
                                       runtime-option.


            config_cache_reload
                                  .       ,
                             .          Zabbix
                     Zabbix                    .
                                    (              -c)
                    PID-   ,                   ,
                        PID-   .

            snmp_cache_reload
                         SNMP.

            housekeeper_execute
```

housekeeper.  ,  housekeeper
.

diaginfo[=section]

section. Section  historycache, preprocessing, locks.

.

prof_enable[=target]
,  ,  target

prof_disable[=target]
,  ,  target

log_level_increase[=target]
,  ,  target
.

log_level_decrease[=target]
,  ,  target
.

process-type
(availability manager,
browser poller, configuration syncer, data sender, discovery manager,
discovery worker, history syncer, housekeeper, http poller,
icmp pinger, internal poller, ipmi manager, ipmi poller,
java poller, odbc poller, poller, agent poller, http agent poller,
snmp poller, preprocessing manager, preprocessing worker,
self-monitoring, snmp trapper, task manager, trapper,
unreachable poller, vmware collector)

process-type,N
(  , poller,3)

pid

process-type
(availability manager,
browser poller, configuration syncer, data sender, discovery manager,
discovery worker, history syncer, housekeeper, http poller,
icmp pinger, internal poller, ipmi manager, ipmi poller,
java poller, odbc poller, poller, agent poller, http agent poller,
snmp poller, preprocessing manager, preprocessing worker,
self-monitoring, snmp trapper, task manager, trapper,
unreachable poller, vmware collector)

process-type,N
(  , history syncer,1)

pid

scope
(rwlock, mutex, processing)  process-type
(  , history syncer,1,processing)

-T, --test-config
.

-h, --help
.

-V, --version
.

/usr/local/etc/zabbix_proxy.conf
Zabbix (
).

.

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) ZABBIX_SENDER(1)

zabbix_sender - Zabbix

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] -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


zabbix_sender -
             Zabbix       .        Zabbix
                Zabbix trapper               .
          ,
      ,            Allowed hosts                .


-c, --config config-file
               config-file. Zabbix sender
                         agentd.         Zabbix sender
                            .
             Hostname, ServerActive, SourceIP, TLSConnect, TLSCAFile,
       TLSCRLFile, TLSServerCertIssuer, TLSServerCertSubject,
       TLSCertFile, TLSKeyFile, TLSPSKIdentity   TLSPSKFile.      ,
                     HostnameItem,             ;

       ( .      -s).                          ,
                               agent ServerActive.
                                           ,
                                      .


-z, --zabbix-server server
                IP-        Zabbix.
                      ,
       IP-       .                 --config
                  ServerActive,
       agentd.

-p, --port port
                    Zabbix trapper,              .
               10051.                 --config
                            ServerActive,
```

agentd.

-I, --source-address IP-address
IP- . --config
SourceIP,
agentd.

-t, --timeout seconds
- . : 1-300
( : 60)

-s, --host host
, (
- Zabbix). IP- DNS-
. --config
Hostname,
agentd.

-k, --key key
, .

-o, --value value
.

-i, --input-file input-file
. -
<input-file>, .
:
<hostname> <key> <value>.
. 3
: <hostname> <key> <value>, "hostname" -
, -
Zabbix, "key" - ,
"value" - . - <hostname>,
agent
--host.

:

"Linux DB3" db.connections 43

- Zabbix. Zabbix sender 250
.
UTF-8.
.
:

- .
- .
- , ,
.
-
.
- .
- (\n)
.

•

.

-T, --with-timestamps

--input-file.

4
: <hostname> <key> <timestamp> <value>.
Unix timestamp.
,
,
.

:

"Linux DB3" db.connections 1429533600 43

. --input-file.

,
“no data”,
;
, .

-N, --with-ns

--with-timestamps.

5
: <hostname> <key> <timestamp> <ns> <value>.

:

"Linux DB3" db.connections 1429533600 7402561 43

. --input-file.

-r, --real-time

.
.

--tls-connect value

. :

unencrypted
( )

psk TLS

cert TLS

--tls-ca-file CA-file

, CA
.

--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-psk-identity PSK-identity
PSK.

--tls-psk-file PSK-file
, .

--tls-cipher13 cipher-string
OpenSSL 1.1.1 TLS 1.3.
.
, OpenSSL 1.1.1.

--tls-cipher cipher-string
GnuTLS ( TLS 1.2 )
OpenSSL ( TLS 1.2).
.

-v, --verbose
, -vv - .

-h, --help
.

-V, --version
.

0,
. ,
, 2.
, 1.

```
zabbix_sender -c  /etc/zabbix/zabbix_agentd.conf  -k  mysql.queries  -o
342.45
```

342.45 mysql.queries
.
Zabbix, agent.

```
zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -s "Monitored Host"  -k
mysql.queries -o 342.45
```

342.45 mysql.queries
Monitored Host, Zabbix,
agent.

```
zabbix_sender -z 192.168.1.113 -i data_values.txt
```

data_values.txt Zabbix
IP 192.168.1.113. .

```
echo "- hw.serial.number  1287872261  SQ4321ASDF"  |  zabbix_sender  -c
/usr/local/etc/zabbix_agentd.conf -T -i -
```

Zabbix,                    agent.
                ,
                    .

echo '"Zabbix server" trapper.item ""' | zabbix_sender -z 192.168.1.113
-p 10000 -i -
                                    Zabbix   IP-
          192.168.1.113      10000            .
                                   .

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
            342.45                  mysql.queries
            Monitored Host         IP 192.168.1.113
              TLS         .

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
            342.45                  mysql.queries
            Monitored Host         IP 192.168.1.113
              TLS                      (PSK).

.
Documentation

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)                                              ZABBIX_SERVER(8)

zabbix_server -         Zabbix

zabbix_server [-c config-file]
zabbix_server [-c config-file] -R runtime-option
zabbix_server [-c config-file] -T
zabbix_server -h
zabbix_server -V

zabbix_server                              Zabbix.

-c, --config config-file
                       config-file               .

-f, --foreground
                Zabbix           .

-R, --runtime-control runtime-option
                                      runtime-option.

config_cache_reload
. , . PID-

snmp_cache_reload
SNMP.

housekeeper_execute
housekeeper. , housekeeper .

trigger_housekeeper_execute
trigger housekeeper ( ). , trigger housekeeper

secrets_reload
Vault. , .

diaginfo[=section]
. historycache, preprocessing, al

prof_enable[=target]
, , target

prof_disable[=target]
, , target

ha_status
. .

ha_remove_node
ha_remove_node[=name|ID] ID, ha_status.

ha_set_failover_delay[=delay]
. .

log_level_increase[=target]
, , target

log_level_decrease[=target]
, , target

process-type
(alerter, alert manager,
alert syncer, 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 poller, http agent poller, icmp pinger, internal poller, ipmi manager, ipmi poller, java poller, lld manager, lld worker, odbc poller, poller, agent poller, preprocessing manager, preprocessing worker, proxy poller, proxy group manager, report manager, report writer, self-monitoring, service manager, snmp poller, snmp trapper, task manager, timer, trapper, trigger housekeeper, unreachable poller, vmware collector)

process-type,N
( , poller,3)
pid

process-type
(alerter, alert manager,
alert syncer, availability manager, browser poller, config-

```
                uration syncer, configuration syncer worker, connector man-
                ager,    connector worker,    discovery manager,    discov-
                ery worker,  escalator,  ha manager,  history poller,  his-
                tory syncer, housekeeper,  http poller,  http agent poller,
                icmp pinger,  internal poller,  ipmi manager,  ipmi poller,
                java poller, lld manager, lld worker, odbc poller,  poller,
                agent poller,  preprocessing manager, preprocessing worker,
                proxy poller,  proxy group manager,   report manager,   re-
                port writer, self-monitoring, service manager, snmp poller,
                snmp trapper,  task manager, timer, trapper, trigger house-
                keeper, unreachable poller, vmware collector)

            process-type,N
                             (     , history syncer,1)

            pid

            scope
                (rwlock, mutex, processing)              process-type
                (     , history syncer,1,processing)

    -T, --test-config
                               .

    -h, --help
                           .

    -V, --version
                               .


    /usr/local/etc/zabbix_server.conf
                                   Zabbix         (
                               ).

 .
    Documentation

    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)                                  ZABBIX_WEB_SERVICE(8)


    zabbix_web_service -   -    Zabbix


    zabbix_web_service [-c config-file]
    zabbix_web_service [-c config-file] -T
    zabbix_web_service -h
    zabbix_web_service -V


    zabbix_web_service —                        -           Zab-
    bix.
```

```
       -c, --config config-file
                              config-file                .

       -T, --test-config
                                    .

       -h, --help
                                .

       -V, --version
                                    .


       /usr/local/etc/zabbix_web_service.conf
                                             -     Zabbix (
                                  ).

 .


       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)
```

## Уведомление об авторских правах

Документация Zabbix НЕ распространяется под лицензией AGPL-3.0. Использование документации Zabbix регулируется следующими условиями:

Вы можете создать печатную копию этой документации исключительно для личного использования. Преобразование в другие форматы разрешено при условии, что фактическое содержимое не изменяется и не редактируется каким-либо образом. Вы не должны публиковать или распространять эту документацию в какой-либо форме или на каких-либо носителях, за исключением случаев, когда вы распространяете документацию способом, аналогичным тому, как Zabbix распространяет её (то есть, в электронном виде для загрузки на веб-сайте Zabbix), или на USB-накопителе или аналогичном носителе, при условии, однако, что документация распространяется вместе с программным обеспечением на том же носителе. Любое другое использование, такое как распространение печатных копий или использование этой документации, полностью или частично, в другой публикации, требует предварительного письменного согласия уполномоченного представителя Zabbix. Zabbix оставляет за собой все права на эту документацию, прямо не предоставленные выше.