

Zabbix documentation in Russian

Эти страницы содержат официальную документацию по Zabbix.

Используйте боковое меню для просмотра страниц документации.

Чтобы была возможность наблюдения за страницами, войдите с именем пользователя и паролем вашей учетной записи с [форума Zabbix](#).

Декларация переводчиков о качестве перевода

- Перевод русской документации не является официальным (впрочем как и веб-интерфейса) и осуществляется переводчиками сообщества, представленными на [этой](#) странице. Переводчики являются опытными пользователями системы мониторинга Zabbix.
- Эта декларация обновлена 01.07.2013

1. Введение

Пожалуйста, воспользуйтесь боковым меню для доступа к содержимому раздела Введение.

1 Структура руководства

Структура

Содержимое данного Руководства о Zabbix 4.4 разделено на разделы и подразделы для обеспечения удобного доступа к представляющим интерес темам.

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

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

Разделы

[Введение](#) дает общую информацию об этой версии программного обеспечения Zabbix. Чтение этого раздела должно вооружить вас некоторыми весомыми причинах в пользу выбора Zabbix.

[Понятия в Zabbix](#) объясняют, используемую в Zabbix, терминологию и рассказывают о компонентах Zabbix в деталях.

Разделы [Установки](#) и [Быстрого старта](#) должны помочь вам в начале работы с Zabbix. [Готовое решение Zabbix](#) - альтернатива для быстрой возможности попробовать Zabbix, чтобы понять, что это такое и как использовать Zabbix.

Раздел [Настройка](#) - один из самых больших и очень важных разделов в этом руководстве. Этот раздел наполнен важными советами о том как настроить Zabbix для мониторинга вашей среды, от настройки узлов сети для получения необходимых данных, до просмотра данных, настройки оповещений и удаленных команд, которые будут выполняться в случае возникновения проблем.

Раздел [Услуги IT](#) объясняет как использовать Zabbix для высокоуровневого обзора вашей среды мониторинга.

[Веб-мониторинг](#) должен помочь вам в понимании мониторинга доступности веб сайтов.

[Мониторинг виртуальных машин](#) содержит рекомендации по настройке мониторинга среды VMware.

Разделы [Обслуживание](#), [Регулярные выражения](#), [Подтверждение событий](#) и [Экспорт/импорт XML](#) рассказывают каким еще образом можно использовать различные аспекты программного обеспечения Zabbix.

[Обнаружение](#) содержит инструкции по настройке автоматического обнаружения устройств сети, активных агентов, файловых систем, сетевых интерфейсов и т.д.

[Распределенный мониторинг](#) описывает возможности использования Zabbix в больших и более сложных средах.

[Шифрование](#) помогает понять возможности шифрования соединений между Zabbix компонентами.

[Веб-интерфейс](#) содержит информацию, относящуюся к использованию веб-интерфейса Zabbix.

Раздел [API](#) содержит подробности о работе с Zabbix API.

Подробный список технической информации включен в [Приложения](#). Здесь вы также можете найти раздел Часто задаваемых вопросов.

2 Что такое Zabbix

Обзор

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

Zabbix - это решение распределенного мониторинга корпоративного класса с открытыми исходными кодами.

Zabbix - это программное обеспечение для мониторинга многочисленных параметров сети, жизнеспособности и целостности серверов. Zabbix использует гибкий механизм оповещений, что позволяет пользователям конфигурировать уведомления основанные на e-mail практически для любого события. Это позволяет быстро реагировать на проблемы с серверами. Zabbix предлагает отличные функции отчетности и визуализации данных основанные на данных истории. Это делает Zabbix идеальным для планирования мощности.

Zabbix поддерживает и поллеры, и трапперы. Все отчеты 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. Правильно настроенный, Zabbix может сыграть 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 GPL General Public License версии 2. Это означает, что его исходный код свободно распространяется и доступен для неограниченного круга лиц.

[Коммерческая поддержка](#) доступна и осуществляется самой компанией Zabbix.

Узнайте больше о [возможностях Zabbix](#).

Пользователи Zabbix

Множество организаций различного размера по всему миру полагаются на Zabbix, как на основную платформу мониторинга.

3 Возможности Zabbix

Обзор

Zabbix - высоко интегрированное решение мониторинга сети, которое предлагает 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

- отправку оповещений можно настроить, используя расписания эскалаций, получателей, типов оповещений
- оповещения можно сделать информативными и полезными при использовании переменных макросов
- автоматические действия, включающие в себя удаленные 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, проверить функционал и время ответа

Широкие возможности визуализации

- возможность создавать пользовательские графики, что позволяет комбинировать множество элементов данных в одном месте
- карты сети
- пользовательские комплексные экраны и слайд-шоу наподобие внешнего вида ПАНЕЛИ
- отчеты
- высокоуровневое (бизнес) представление наблюдаемых ресурсов

Хранение данных истории

- данные записываются в базу данных
- настраиваемая история
- встроенная процедура очистки истории

Простая настройка

- добавление наблюдаемых устройств узлами сети
- как только узлы сети имеются в базе данных, они 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-интерфейс

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

Zabbix API

- Zabbix API обеспечивает программируемый интерфейс к Zabbix для массовых манипуляций, для 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 C, имеют высокую производительность и используют небольшой объем памяти
- легко переносимы

Готовность с сложным средам

- простой удаленный мониторинг с использованием Zabbix прокси

4 Обзор Zabbix

Архитектура

Zabbix состоит из нескольких основных программных компонентов, функции которых изложены ниже.

Сервер

Zabbix сервер является основным компонентом, которому агенты сообщают информацию и статистику о доступности и целостности. Сервер является главным хранилищем, в котором хранятся все данные конфигурации, статистики, а также оперативные данные.

База данных

Как таковая вся информация о конфигурации, а так же данные собранные Zabbix, хранятся в базе данных.

Веб-интерфейс

Для легкого доступа к Zabbix из любого места и с любой платформы, поставляется интерфейс на основе Веб. Интерфейс является частью Zabbix сервера и обычно (но не обязательно) работает на том 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.

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 о 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, использование шаблонов значительно упрощает процесс. Однако, такое построение системы позволяет создавать очень гибкие инсталляции.

2. Определения

Обзор

В этом разделе вы можете ознакомиться со значениями некоторых терминов, которые используются в Zabbix.

Определения

узел сети

- сетевое устройство, мониторинг которого вы хотите производить, имеющее IP/DNS.

группа узлов сети

- логическая группировка узлов сети; она может содержать узлы сети и шаблоны. Узлы сети и шаблоны в группе узлов сети никаким образом не связаны с друг другом. Группы узлов сети используются при назначении прав доступа к узлам сети различным группам пользователей.

элемент данных

- конкретный фрагмент данных, который вы хотите получать от узла сети, метрика

предобработка значений

- трансформация полученного метрикой значения перед сохранением этого значения в базу данных.

триггер

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

Если полученные данные превышают порог, триггер переходит из состояния 'Ok' в состояние 'Проблема'. Если полученные данные ниже порога, триггер остается/возвращается в состояние 'Ok'.

событие

- одиночное возникновение того, что заслуживает внимания, например изменение состояния триггера или обнаружение/авторегистрация агента

тег события

- предопределенный маркер для события. Его можно использовать в корреляции событий, в правах доступа и т.д.

корреляция событий

- гибкий и точный метод корреляции проблем с решением этих проблем.

Например, вы можете задать, что проблема, которую сообщил один триггер, может быть решена другим триггером, который может использовать даже совершенно другой метод сбора данных.

проблема

- триггер, который находится в состоянии "Проблема".

обновление проблемы

- варианты управления проблемами, такими как добавление комментария, подтверждение, изменение важности и закрытие вручную, поставляются Zabbix.

действие

- *предопределенные средства реагирования на события*

Действия состоят из операций (например, отправка оповещений) и условий (когда осуществляется операция)

эскалация

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

способ оповещения

- *способ доставки оповещений; канал доставки.*

оповещение

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

удаленная команда

- *предопределенная команда, которая будет автоматически выполнена на наблюдаемом узле сети при некоторых условиях.*

шаблон

- *набор сущностей (элементы данных, триггеры, графики, комплексные экраны, группы элементов данных, правила низкоуровневого обнаружения, веб-сценарии) готовые к присоединению к одному или нескольким узлам сети.*

Задача шаблонов повысить скорость развертывания задач мониторинга узлов сети; кроме того упростить применение массовых изменений к задачам наблюдения. Шаблоны соединяются напрямую с отдельными узлами сети.

группа элементов данных

- *группировка элементов данных в некую логическую группу.*

веб-сценарий

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

веб-интерфейс

- *веб-интерфейс поставляемый с Zabbix*

ПАНЕЛЬ

- *раздел веб-интерфейса, который отображает сводку и визуализацию важной информации в визуальных блоках называемых виджетами.*

виджет

- *визуальный блок, который отображает информацию определенного вида и источника (сводка, карта, график, часы и т.д.), используется на панели.*

Zabbix API

- *Zabbix API позволяет вам использовать протокол JSON RPC для создания, обновления и получения объектов Zabbix (например, узлов сети, элементов данных, графиков и прочих) или для выполнения любых других пользовательских задач.*

Zabbix сервер

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

Zabbix агент

- процесс разворачиваемый на наблюдаемых целях для активного мониторинга локальных ресурсов и приложений.

Zabbix прокси

- процесс, который может собирать данные от имени Zabbix сервера, перенимая часть нагрузки сервера.

шифрование

- поддержка шифрованной связи между компонентами Zabbix (сервером, прокси, агентом и утилитами zabbix_sender и zabbix_get) с использованием Протокола защиты транспортного уровня (TLS).

обнаружение сети

- автоматическое обнаружение сетевых устройств.

низкоуровневое обнаружение

- автоматическое обнаружение низкоуровневых объектов на конкретном устройстве (например, файловые системы, сетевые интерфейсы и т.д.).

правило низкоуровневого обнаружения

- набор определений для автоматического обнаружения низкоуровневых объектов на устройстве.

прототип элементов данных

- метрика, у которой некоторые параметры представлены в виде переменных, готовых для низкоуровневого обнаружения. После низкоуровневого обнаружения эти переменные автоматически заменяются реальными обнаруженными параметрами и метрика автоматически начинает сбор данных.

прототип триггеров

- триггер, у которого некоторые параметры представлены в виде переменных, готовых для низкоуровневого обнаружения. После низкоуровневого обнаружения эти переменные автоматически заменяются реальными обнаруженными параметрами и триггер автоматически начинает оценку данных.

Прототипы некоторых других Zabbix объектов также используются в низкоуровневом обнаружении

- прототипы графиков, прототипы узлов сети, прототипы групп узлов сети, прототипы групп элементов данных.

авторегистрация агента

- автоматизированный процесс при котором Zabbix агент регистрируется самостоятельно узлом сети и начинает наблюдаться.

3. Процессы Zabbix

3.1 Сервер

Обзор

Zabbix сервер - центральный процесс программного обеспечения Zabbix.

Сервер выполняет опрос и отлов данных, вычисляет триггеры, отправляет оповещения пользователям. Он является центральным компонентом, которому Zabbix агенты и прокси сообщают данные о доступности и целостности систем. Сервер может самостоятельно удаленно проверять сетевые службы (такие как веб-сервера и почтовые сервера), используя простые проверки сервисов.

Сервер является главным хранилищем, в котором хранятся все конфигурационные, статистические и оперативные данные, так же он рассыпает уведомления администраторам в случае возникновения проблем с любой из наблюдаемых систем.

Функционал базового Zabbix сервера разделен на три отдельных компонента; это: Zabbix сервер, веб-интерфейс и хранилище в базе данных.

Все данные о конфигурации Zabbix хранятся в базе данных, с которой взаимодействует и сервер и веб-интерфейс. Например, когда вы создаете новый элемент данных используя веб-интерфейс (или API), запись об этом добавляется в таблицу элементов данных в базе данных. Затем, раз в минуту Zabbix сервер опрашивает таблицу элементов данных для получения списка активных элементов данных, и сохраняет этот список в кэш Zabbix сервера. Именно поэтому любые изменения в веб-интерфейсе Zabbix будут отображены в разделе последних данных с задержкой до двух минут.

Процесс сервера

Если установлен из пакета

Zabbix сервер работает как демон. Для запуска сервера выполните:

```
shell> service zabbix-server start
```

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

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

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

```
shell> service zabbix-server stop  
shell> service zabbix-server restart  
shell> service zabbix-server status
```

Запуск вручную

Если выше приведенные команды не работают, вам необходимо запустить сервер вручную. Найдите путь к бинарному файлу zabbix_server и выполните:

```
shell> zabbix_server
```

Можно использовать следующие параметры командной строки с Zabbix сервером:

-c --config <файл>	путь к файлу конфигурации (по умолчанию /usr/local/etc/zabbix_server.conf)
-R --runtime-control <опция>	выполнение административных функций
-h --help	вывод этого сообщения помощи
-V --version	вывод номера версии

Выполнение административных функций не поддерживается в OpenBSD и NetBSD.

Примеры запуска Zabbix сервера с параметрами командой строки:

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

Управление работой

Опции управления работой:

Опция	Описание	Цель
config_cache_reload	Перезагрузка кэша конфигурации. Игнорируется, если кэш уже загружается в текущий момент времени.	
housekeeper_execute	Запуск процедуры очистки базы данных. Игнорируется, если процедура очистки выполняется в данный момент.	
log_level_increase[=<цель>]	Увеличение уровня журналирования, действует на все процессы, если цель не указана.	pid - Идентификатор процесса (1 до 65535) тип процесса - Все процессы указанного типа (например, poller) тип процесса,N - Тип процесса и номер (например, poller,3)
log_level_decrease[=<цель>]	Уменьшение уровня журналирования, действует на все процессы, если цель не указана.	

Допустимый диапазон PID изменения уровня журналирования одного процесса с 1 до 65535. На системах с большими значениями PID опция <тип процесса,N> может использоваться для изменения уровня журналирования отдельных процессов.

Пример использования административных функций для перезагрузки кэша конфигурации сервера:

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

Пример использования административных функций для вызова выполнения очистки базы данных:

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

Примеры использования административных функций по изменению уровня журналирования:

Увеличение уровня журналирования по всем процессам:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase
```

Увеличение уровня журналирования у второго процесса поллера:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2
```

Увеличение уровня журналирования у процесса с PID 1234:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234
```

Уменьшение уровня журналирования по всем http поллер процессам:

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

Пользователь процесса

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

Если вы попытаетесь запустить сервер от 'root', сервер 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 легкостью получить, например, пароль от базы данных.

Файл конфигурации

Смотрите опции [файла конфигурации](#) для получения подробной информации по настройке Zabbix сервера.

Скрипты запуска

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

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
- Tru64/OSF1

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

Региональные настройки (локаль)

Обратите внимание, что сервер требует UTF-8 локаль, 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.

3.2 Агент

Обзор

Zabbix агенты разворачиваются на наблюдаемых целях для активного мониторинга локальных ресурсов и приложений (статистика жестких дисков, памяти, процессоров и т.д.).

Агент локально собирает оперативную информацию и отправляет данные Zabbix серверу для дальнейшей обработки. В случае проблем (таких как отсутствие свободного места на жестком диске или аварийного завершения процесса сервиса), Zabbix сервер может быстро уведомить администраторов конкретного сервера, сообщившего об ошибке.

Zabbix агенты чрезвычайно эффективны, потому что используют родные системные вызовы для сбора информации статистики.

Пассивные и активные проверки

Zabbix агенты могут выполнять пассивные и активные проверки.

В случае [пассивной проверки](#) агент отвечает на запрос данных. Zabbix сервер (или прокси) запрашивает данные, например, загрузку CPU, и Zabbix агент возвращает результат.

[Активные проверки](#) требуют более сложной обработки. Агент сначала получает список элементов данных для независимой обработки от Zabbix сервера. Далее он будет периодически отправлять новые значения серверу.

Выбор между пассивной и активной проверкой осуществляется выбором соответствующего [типа элемента данных](#). Zabbix агент обрабатывает элементы данных типов 'Zabbix агент' и 'Zabbix агент (активный)'.

Поддерживаемые платформы

Zabbix агент поддерживается на:

- Linux
- IBM AIX
- FreeBSD
- NetBSD
- OpenBSD
- HP-UX
- Mac OS X
- Solaris
- Windows: все версии для настольных компьютеров и серверов начиная с XP

Агент на UNIX системах

Zabbix агент на UNIX-подобных системах запускается на хостах, которые необходимо мониторить.

Установка

Смотрите раздел [установки из пакетов](#) для получения более детальных сведений по установке Zabbix агента из пакетов.

В качестве альтернативы вы можете обратиться к инструкциям по [ручной установке](#), если вы не хотите использовать пакеты.

В целом 32-битный Zabbix агент будет работать на 64-битных системах, но в отдельных случаях возможно возникновение ошибок.

Если установлен пакетом

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

```
shell> service zabbix-agent start
```

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

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

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

```
shell> service zabbix-agent stop  
shell> service zabbix-agent restart  
shell> service zabbix-agent status
```

Запуск вручную

Если выше приведенные команды не работают, вам необходимо запустить агент вручную. Найдите путь к бинарному файлу zabbix_agentd и выполните:

```
shell> zabbix_agentd
```

Агент на Windows системах

Zabbix агент на Windows работает как служба Windows.

Подготовка

Zabbix агент распространяется в виде zip архива. После загрузки архива вам необходимо распаковать его. Выберите любую папку для записи Zabbix агента и файла конфигурации, например:

```
C:\zabbix
```

Скопируйте bin\win64\zabbix_agentd.exe и conf\zabbix_agentd.win.conf файлы в c:\zabbix.

Измените c:\zabbix\zabbix_agentd.win.conf файл в соответствии с вашими требованиями, убедитесь что указан корректный параметр “Hostname”.

Установка

После этого используйте следующую команду для установки Zabbix агента службой Windows:

```
C:\> c:\zabbix\zabbix_agentd.exe -c c:\zabbix\zabbix_agentd.win.conf -i
```

Теперь у вас должна быть возможность настраивать службу “Zabbix agent” так же как и любую другую службу Windows.

Смотрите [более детальные сведения](#) по установке и запуску Zabbix агента на Windows.

Другие опции агента

Имеется возможность запуска нескольких экземпляров агента на одном узле сети. Одиночный экземпляр может быть настроен на использование файла конфигурации по умолчанию или файл конфигурации указанный в командной строке. В случае нескольких экземпляров каждый агент должен иметь свой собственный файл конфигурации (один из них может использовать файл конфигурации по умолчанию).

Следующие параметры командной строки могут быть использованы с Zabbix агентом:

Параметр	Описание
UNIX и Windows агент	
-c --config <файл-конфигурации>	Путь к файлу конфигурации. Вы можете использовать данную опцию, чтобы задать файл конфигурации, размещенному в папке отличной от заданной по умолчанию. В UNIX, путь по умолчанию /usr/local/etc/zabbix_agentd.conf или как задано во время компиляции переменными --sysconfdir или --prefix

Параметр	Описание
UNIX и Windows агент	
-p --print	В Windows, путь по умолчанию c:\zabbix_agentd.conf Вывод известных элементов данных и выход. <i>Обратите внимание:</i> Также для получения результатов пользовательских параметров , вы можете указать файл конфигурации (если он находится вне папки заданной по умолчанию).
-t --test <ключ элемента данных>	Тестирование указанного элемента данных и выход. <i>Обратите внимание:</i> Также для получения результатов пользовательских параметров , вы можете указать файл конфигурации (если он находится вне папки заданной по умолчанию).
-h --help	Вывод справочной информации.
-V --version	Вывод номера версии
Только для UNIX агента	
-R --runtime-control <опция>	Выполнение административных функций. Смотрите управление работой .
Только для Windows агента	
-m --multiple-agents	Использование нескольких экземпляров агента (с -i,-d,-s,-x функциями). Для отделения имени экземпляров служб, каждое имя службы будет в значении Hostvalue из указанного файла конфигурации.
Только для Windows агента (функции)	
-i --install	Установка Zabbix агента службой
-d --uninstall	Удаление службы Zabbix Windows агента
-s --start	Запуск службы Zabbix Windows агента
-x --stop	Остановка службы Zabbix Windows агента

Специальные **примеры** использования параметров командной строки:

- отображение всех встроенных элементов данных с их значениями
- тестирование пользовательского параметра с ключом “mysql.ping” заданном в указанном файле конфигурации
- установка службы “Zabbix агента” в Windows с использованием пути по умолчанию до файла конфигурации c:\zabbix_agentd.conf
- установка службы “Zabbix Agent [Hostname]” в Windows с использованием файла конфигурации zabbix_agentd.conf, размещенного в той же самой папке что и бинарный файл агента и определение уникального имени службы, используя значение Hostname из файла конфигурации

```
shell> zabbix_agentd --print
shell> zabbix_agentd -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
shell> zabbix_agentd.exe -i
shell> zabbix_agentd.exe -i -m -c zabbix_agentd.conf
```

Управление работой

Используя опции административных функций, вы можете изменить уровень журналирования у процессов агента.

Опция	Описание	Цель
log_level_increase[= <цель>]	Увеличение уровня журналирования. Действует на все процессы, если цель не указана.	Цель можно указать с помощью: pid - идентификатора процесса (от 1 до 65535) тип процесса - все процессы указанного типа (например, poller)
log_level_decrease[Уменьшение уровня	

Опция	Описание	Цель
=<цель>]	журналирования. Действует на все процессы, если цель не указана.	тип процесса, N - тип процесса и номер (например, poller,3)
Обратите внимание, что допустимый диапазон PID изменения уровня журналирования одного процесса с 1 до 65535. На системах с большими значениями PID опция <тип процесса,N> может использоваться для изменения уровня журналирования отдельных процессов.		

Примеры:

- увеличение уровня журналирования по всем процессам
- увеличение уровня журналирования у второго процесса listener
- увеличение уровня журналирования у процесса с PID 1234
- уменьшение уровня журналирования по всем процессам active check

```
shell> zabbix_agentd -R log_level_increase
shell> zabbix_agentd -R log_level_increase=listener,2
shell> zabbix_agentd -R log_level_increase=1234
shell> zabbix_agentd -R log_level_decrease="active checks"
```

Выполнение административных функций не поддерживается в OpenBSD, NetBSD и Windows.

Пользователь процесса

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

Если вы попытаетесь запустить агента от 'root', он сразу переключится на пользователя 'zabbix', который должен присутствовать в вашей системе. Единственный способ запустить агента от пользователя 'root' - соответствующим образом отредактировать параметр 'AllowRoot' в файле конфигурации агента.

Файл конфигурации

Для получения подробной информации смотрите параметры файла конфигурации по настройке [zabbix_agentd](#) или [Windows агента](#).

Региональные настройки (локаль)

Обратите внимание что сервер требует UTF-8 локаль чтобы некоторые текстовые элементы данных могли быть интерпретированы корректно. Большинство современных Unix-подобных систем имеют локаль UTF-8 установленной по умолчанию, тем не менее, есть некоторые системы где это необходимо задать вручную.

Код завершения процесса

До версии Zabbix 2.2, Zabbix агент возвращал 0 в случае успешного завершения процесса и код 255 в случае ошибки. Начиная с версии Zabbix 2.2 и выше Zabbix агент возвращает код 0 в случае корректного завершения процесса и код 1 в случае отказа/аварийного завершения процесса.

3.3 Прокси

Обзор

Zabbix прокси - это процесс, способный собирать данные мониторинга с одного или нескольких наблюдаемых устройств и отправлять эту информацию Zabbix серверу, таким образом прокси

работает от имени сервера. Все собранные данные локально буферизуются и затем отправляются Zabbix серверу, которому принадлежит этот прокси.

Развертывание прокси не обязательно, но может быть очень полезно для распределения нагрузки одиночного Zabbix сервера. Если только прокси собирают данные, то обработка этих данных на сервере не так сильно нагружает CPU и I/O диска.

Zabbix прокси - идеальное решение для централизованного мониторинга удаленных объектов, филиалов и сетей, где отсутствуют локальные администраторы.

Для Zabbix прокси требуется отдельная база данных.

Zabbix прокси поддерживает следующие базы данных SQLite, MySQL и PostgreSQL. Вы можете использовать Oracle или IBM DB2 на свой собственный риск, при этом возможны определенные ограничения, например в [значениях](#), возвращаемых правилами низкоуровневого обнаружения.

Смотрите также: [Использование прокси в распределенной среде](#)

Процесс прокси

Если установлен из пакета

Zabbix прокси работает как демон. Для запуска прокси выполните:

```
shell> service zabbix-proxy start
```

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

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

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

```
shell> service zabbix-proxy stop  
shell> service zabbix-proxy restart  
shell> service zabbix-proxy status
```

Запуск вручную

Если выше приведенные команды не работают, вам необходимо запустить прокси вручную. Найдите путь к бинарному файлу zabbix_proxy и выполните:

```
shell> zabbix_proxy
```

Следующие параметры командной строки могут быть использованы с Zabbix прокси:

-c --config <файл>	путь к файлу конфигурации (по умолчанию /etc/zabbix/zabbix_proxy.conf)
-R --runtime-control <опция>	выполнение административных функций
-h --help	вывод этого сообщения помощи
-V --version	вывод номера версии

Выполнение административных функций не поддерживается в OpenBSD и NetBSD.

Примеры запуска Zabbix прокси с параметрами командой строки:

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

Управление работой

Опции управления работой:

Опция	Описание	Цель
config_cache_reload	Перезагрузка кэша конфигурации. Игнорируется, если кэш уже загружается в текущий момент времени.	
housekeeper_execute	Активный Zabbix прокси подключится к Zabbix серверу и запросит данные конфигурации. Запуск процедуры очистки базы данных. Игнорируется, если процедура очистки выполняется в данный момент.	
log_level_increase[=<цель>]	Увеличение уровня журналирования, действует на все процессы, если цель не указана.	pid - Идентификатор процесса (1 до 65535) тип процесса - Все процессы указанного типа (например, poller) тип процесса,N - Тип процесса и номер (например, poller,3)
log_level_decrease[=<цель>]	Уменьшение уровня журналирования, действует на все процессы, если цель не указана.	
Допустимый диапазон PID изменения уровня журналирования одного процесса с 1 до 65535. На системах с большими значениями PID опция <тип процесса,N> может использоваться для изменения уровня журналирования отдельных процессов.		

Пример использования административных функций для перезагрузки кэша конфигурации прокси:

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

Пример использования административных функций для вызова выполнения очистки базы данных:

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

Примеры использования административных функций по изменению уровня журналирования:

Увеличение уровня журналирования по всем процессам:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase
```

Увеличение уровня журналирования у второго процесса поллера:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=poller,2
```

Увеличение уровня журналирования у процесса с PID 1234:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=1234
```

Уменьшение уровня журналирования по всем http поллер процессам:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_decrease="http polller"
```

Пользователь процесса

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

Если вы попытаетесь запустить прокси от 'root', он сразу переключится на пользователя 'zabbix', который должен присутствовать в вашей системе. Единственный способ запустить прокси от пользователя 'root' - соответствующим образом отредактировать параметр 'AllowRoot' в конфигурационном файле прокси.

Файл конфигурации

Для получения подробной информации смотрите опции [файла конфигурации](#) по настройке zabbix_proxy.

Поддерживаемые платформы

Zabbix прокси запускается на тех же [поддерживаемых платформах](#), что и Zabbix сервер.

Региональные настройки (локаль)

Обратите внимание что сервер требует UTF-8 локаль чтобы некоторые текстовые элементы данных могли быть интерпретированы корректно. Большинство современных Unix-подобных систем имеют локаль UTF-8 установленной по умолчанию, тем не менее, есть некоторые системы где это необходимо задать вручную.

3.4 Java gateway

Обзор

Начиная с Zabbix 2.0 появился новый демон Zabbix, называемый “Zabbix Java gateway”, обеспечивающий нативную поддержку мониторинга JMX приложений. Zabbix Java gateway - это демон написанный на языке Java. Когда Zabbix сервер хочет знать значение конкретного JMX счетчика узла сети, он опрашивает Zabbix Java gateway, который используя [API управления JMX](#) опрашивает интересующее удаленное приложение. Приложению не требуется никакого дополнительного программного обеспечения, оно просто должно быть запущено с опцией командной строки `-Dcom.sun.management.jmxremote`.

Java gateway принимает входящие подключения от Zabbix сервера или прокси и может быть использован только как “пассивный прокси”. Но в отличии от Zabbix прокси, Java gateway может использоваться с Zabbix прокси (тогда как один Zabbix прокси не может работать через другой Zabbix прокси). Доступ к каждому Java gateway настраивается непосредственно в файле конфигурации Zabbix сервера или прокси, таким образом только один Java gateway может быть настроен на Zabbix сервере или Zabbix прокси. Если у узла сети есть элементы данных типа **JMX агент** и элементы данных других типов, то только элементы данных **JMX агент** будут запрошены через Java gateway.

Когда элемент данных должен быть обновлен через Java gateway, Zabbix сервер или прокси подключается к Java gateway и запрашивает значение, Java gateway в свою очередь запрашивает это значение и возвращает серверу или прокси. Таким образом, Java gateway никакие значения не кэширует.

У Zabbix сервера и прокси есть специальный тип процессов, которые подключаются к Java gateway, их количество настраивается опцией **StartJavaPollers**. Внутренне, Java gateway запускается несколькими потоками, настраиваемыми опцией **START_POLLERS**. На стороне сервера, если соединение занимает более чем **Timeout** секунд, оно будет завершено, но Java gateway может оставаться занят получением значения JMX счетчика. Чтобы решить эту проблему, Java gateway начиная с Zabbix 2.0.15, Zabbix 2.2.10 и Zabbix 2.4.5 поддерживают опцию **TIMEOUT**, позволяющую указать время ожидания сетевых операций JMX.

Zabbix сервер и прокси будут пытаться максимально объединить запросы к одной цели JMX (зависит от интервалов обновления элементов данных) и отправлять их в Java gateway за одно подключение для лучшей производительности.

Рекомендуется выставить значение **StartJavaPollers** меньшим или равным чем **START_POLLERS**, в противном случае могут возникнуть ситуации, когда потоков Java gateway может не хватить для обслуживания входящих запросов.

Разделы ниже рассказывают о том как получить и запустить Zabbix Java gateway, как настроить Zabbix сервер (или Zabbix прокси) для использования Zabbix Java gateway в мониторинге JMX, и как

настроить элементы данных Zabbix в Zabbix веб-интерфейсе, которые соответствуют конкретным JMX счетчикам.

3.4.2.1 Получение Java gateway

Вы можете установить Java gateway как из исходных кодов, так и из пакетов, которые можно загрузить с [Zabbix веб-сайта](#).

Смотрите инструкции по установке Java gateway:

- [из исходных кодов](#)
- из пакетов для [RHEL/CentOS](#)
- из пакетов для [Debian/Ubuntu](#)

3.4.2.2 Обзор файлов из поставки Java gateway

Независимо от того как вы получили Java gateway, в итоге у вас должен быть набор скриптов, JAR файлом и файлами конфигурации в папке \$PREFIX/sbin/zabbix_java. Назначение этих файлов изложено ниже.

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

Собственно JAR файл Java gateway.

`lib/logback-core-0.9.27.jar`
`lib/logback-classic-0.9.27.jar`
`lib/slf4j-api-1.6.1.jar`
`lib/android-json-4.3_r3.1.jar`

Зависимости Java gateway: [Logback](#), [SLF4J](#), и библиотека [Android JSON](#).

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

Файлы конфигурации для Logback.

`shutdown.sh`
`startup.sh`

Скрипты для удобства запуска и остановки Java gateway.

`settings.sh`

Файл конфигурации, который используется вышеупомянутыми скриптами запуска и остановки.

3.4.2.3 Настройка и запуск Java gateway

По умолчанию, Java gateway слушает порт 10052. Если вы планируете работу Java gateway на другом порту, то вы можете указать его в скрипте `settings.sh`. Смотрите описание [файла конфигурации Java gateway](#) для получения сведений о том как указать эту и другие опции.

Порт 10052 не [зарегистрирован в IANA](#).

Выполнив настройки, вы можете запустить Java gateway, выполнив скрипт запуска:

`$./startup.sh`

Точно так же, если вам более не требуется Java gateway, выполните скрипт завершения работы для остановки Java gateway:

`$./shutdown.sh`

Обратите внимание, что в отличии от сервера и прокси, Java gateway легок и не требует наличия базы данных.

3.4.2.4 Настройка сервера для использования с Java gateway

Когда Java gateway запущен и работает, вы должны указать Zabbix серверу где искать Zabbix Java gateway. Чтобы это сделать, укажите параметры JavaGateway и JavaGatewayPort в [файле конфигурации сервера](#). Если же узел сети на котором работает JMX приложение наблюдается через Zabbix прокси, то параметры соединения указываются в [файле конфигурации прокси](#).

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

По умолчанию, сервер не запускает процессы связанные с мониторингом JMX. Если же вы хотите использовать этот тип мониторинга, то вам нужно указать количество экземпляров Java поллеров. Вы можете это сделать таким же способом как и изменение количества поллеров и трапперов.

```
StartJavaPollers=5
```

Не забудьте перезапустить сервер или прокси после того как закончите изменение настроек.

3.4.2.5 Отладка Java gateway

В случае возникновения каких-либо проблем с Java gateway или в случае, если сообщение об ошибке элемента данных в веб-интерфейсе недостаточно информативно, вы можете обратиться к файлу журнала Java gateway.

По умолчанию, Java gateway записывает журнал в файл /tmp/zabbix_java.log с уровнем журналирования "info". Бывает, что этой информации недостаточно и требуется информация уровня журналирования "debug". Чтобы увеличить уровень журналирования, отредактируйте файл lib/logback.xml и измените атрибут level тега <root> на "debug":

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

Обратите внимание, что в отличии от Zabbix сервера или Zabbix прокси, вам не нужно перезапускать Zabbix Java gateway после изменения файла logback.xml - изменения в logback.xml будут применены автоматически. Когда вы завершите отладку, вы можете вернуть уровень журналирования обратно в "info".

Если вы хотите записывать журнал в другой файл или в совершенно другую среду такую как база данных, настройте файл logback.xml в соответствии с вашими потребностями. Обратитесь к [Руководству по Logback](#) для получения более подробных сведений.

Иногда для отладки полезно запустить Java gateway как консольное приложение, а не как демон. Чтобы это сделать, закомментируйте переменную PID_FILE в settings.sh. Если PID_FILE не указан, скрипт startup.sh запускает Java gateway как консольное приложение, при этом Logback использует файл lib/logback-console.xml, который не только выводит журнал в консоль, но и имеет уровень журналирования "debug".

В заключение, отметим, поскольку Java gateway использует SLF4J для журналирования, вы можете заменить Logback выбранным вами фреймворком, поместим соответствующий JAR файл в каталог lib. Обратитесь к [Руководство по SLF4J](#) для получения более подробных сведений.

3.5 Sender

Обзор

Zabbix sender - это утилита командной строки, которая может быть использована для отправки данных о производительности Zabbix серверу для последующей их обработки.

Обычно эта утилита используется в долго выполняемых пользовательских скриптах для периодической отправки данных о доступности и производительности.

Чтобы отправлять результаты напрямую на Zabbix сервер или прокси, необходимо задать тип [элемента данных траппер](#).

Отправка одного значения

Пример отправки одного значения на Zabbix сервер используя Zabbix sender:

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

где:

- z - хост Zabbix сервера (также можно использовать IP адрес)
- s - техническое имя наблюдаемого узла сети (указанное в веб-интерфейсе Zabbix)
- k - ключ элемента данных
- o - отправляемое значение

Если объекты содержат пробелы, то эти объекты должны быть заключены в двойные кавычки.

Обратитесь к [странице помощи Zabbix sender](#) для получения более подробной информации.

Zabbix sender в Windows можно выполнить тем же самым способом:

```
zabbix_sender.exe [опции]
```

Начиная с версии Zabbix 1.8.4 утилита Zabbix_sender была улучшена в плане отправки данных в реальном времени. Имеется ввиду, что большое количество значений, полученное за короткий промежуток времени, будут накапливаться во временном стеке и затем будет отправлено серверу за одно соединение. Данные, пришедшие через менее, чем 0,2 сек после предыдущего значения накапливаются в одном стеке, но максимальное время их накопления до отправки все же 1 секунда.

Zabbix sender завершится, если в файле конфигурации имеется ошибочный (не соответствующий определению *параметр=значение*) параметр.

3.6 Get

Обзор

Zabbix get - это утилита командной строки, которая подключается к Zabbix агенту и получает от него запрашиваемую информацию.

Утилита обычно используется для диагностики Zabbix агентов.

Выполнение Zabbix get

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

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

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

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

Обратите внимание, что элемент данных содержит пробелы поэтому используются двойные кавычки для выделения ключа элемента данных. Двойные кавычки не являются частью ключа элемента данных; они будут обрезаны командной строкой и не будут переданы Zabbix агенту.

Zabbix get принимает следующие параметры командной строки:

-s --host <имя хоста или IP>	Требуется указать имя хоста или IP адрес хоста
-p --port <номер порта>	Требуется указать номер порта агента, запущенного на узле сети. По умолчанию 10050.
-I --source-address <IP адрес>	Требуется указать IP адрес для исходящего подключения
-k -key <ключ элемента данных>	Требуется указать ключ элемента данных, значение которого мы хотим получить.
-h --help	Вывод этого сообщения помощи
-V --version	Вывод номера версии

Смотрите также [страницу помощи по Zabbix get](#) для получения более подробной информации.

Zabbix get в Windows можно выполнить тем же самым способом:

```
zabbix_get.exe [опции]
```

4. Установка

4.1 Получение Zabbix

Обзор

Имеется четыре способа получения Zabbix:

- Установка из [пакетов](#)
- Загрузка самого нового архива с исходными кодами и [самостоятельная их сборка](#)
- Установка из [контейнеров](#)
- Загрузка [решения виртуализации](#)

Для загрузки самых новых исходных кодов или решения виртуализации, перейдите на [страницу загрузки Zabbix](#), где указаны прямые ссылки на последние версии. Ссылки на более старые версии находятся ниже ссылок на стабильные версии.

4.2 Требования

Аппаратное обеспечение

Память

Zabbix требуется и оперативная память, и физическая память на жестком диске. Отправной точкой могут быть 128 МБ оперативной памяти и 256 МБ свободного места на жестком диске. Впрочем, очевидно, что объем необходимой дисковой памяти зависит от количества наблюдаемых узлов сети и наблюдаемых параметров. Если вы планируете достаточно долго хранить историю наблюдаемых параметров, то потребуется по крайней мере несколько гигабайт для хранения данных истории в базе данных. Каждый процесс демона Zabbix требует несколько подключений к базе данных. Объем памяти требуемый каждым подключением к базе данных зависит от настроек базы данных.

Чем больше оперативной памяти у вас имеется, тем быстрее работает база данных (а следовательно, и Zabbix)!

CPU

Zabbix и особенно база данных может потребовать значительных процессорных ресурсов в зависимости от количества наблюдаемых параметров и выбранной базы данных.

Другое оборудование

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

Примеры конфигураций оборудования

В таблице приводятся несколько вариантов аппаратных конфигураций:

Название	Платформа	CPU/Память	База данных	Наблюдаемые узлы сети
Маленькая	CentOS	Виртуальная машина	MySQL InnoDB	20
Средняя	CentOS	2 ядра CPU / 2ГБ	MySQL InnoDB	500
Большая	RedHat Enterprise Linux	4 ядра CPU / 8ГБ	RAID10 MySQL InnoDB или PostgreSQL	>1000
Очень большая	RedHat Enterprise Linux	8 ядер CPU / 16ГБ	Быстрый RAID10 MySQL InnoDB или PostgreSQL	>10000

Фактически параметры конфигурации зависят от количества активных элементов данных и частоты обновления этих элементов. Настоятельно рекомендуем производить запуск базы данных на отдельном сервере для крупных установок.

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 всех ведущих версиях ОС.

Zabbix протестирован на следующих платформах:

- Linux
- IBM AIX
- FreeBSD
- NetBSD
- OpenBSD
- HP-UX
- Mac OS X
- Solaris
- Windows: все версии для настольных компьютеров и серверов начиная с XP (только Zabbix агент)

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

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

Программное обеспечение

Zabbix построен на современном веб-сервере Apache, ведущих СУБД, и языке сценариев PHP.

Системы управления базами данных

Программа	Версия	Комментарии
MySQL	5.0.3 - 8.0.x	Требуется, если MySQL используется как основная база данных Zabbix'a. Требуется InnoDB engine. MariaDB также работает с Zabbix.
Oracle	10g или более новая	Требуется, если Oracle используется как основная база данных Zabbix'a.
PostgreSQL	8.1 или более новая	Требуется, если PostgreSQL используется как основная база данных Zabbix'a. Предлагаем использовать PostgreSQL по крайней мере версии 8.3, который показывает очень хорошую производительность VACUUM .
IBM DB2	9.7 или более новая	Требуется, если IBM DB2 используется как основная база данных Zabbix'a.
SQLite	3.3.5 или более новая	Поддерживается только на стороне Zabbix прокси. Требуется, если SQLite используется базой данных Zabbix прокси.

Поддержка IBM DB2 является экспериментальной!

Веб-интерфейс

Следующее программное обеспечение потребуется для работы веб-интерфейса Zabbix:

Приложение	Версия	Комментарии
Apache	1.3.12 или	

Приложение	Версия	Комментарии
<i>PHP</i>	более поздняя 5.4.0 или более поздняя	
<i>gd</i>	2.0 или более поздняя	Расширение PHP GD должен поддерживать формат PNG (<i>--with-png-dir</i>), JPEG (<i>--with-jpeg-dir</i>) изображения и FreeType 2 (<i>--with-freetype-dir</i>).
<i>bcmath</i>		php-bcmath (<i>--enable-bcmath</i>)
<i>ctype</i>		php-ctype (<i>--enable-ctype</i>)
<i>libXML</i>	2.6.15 или более поздняя	php-xml или php5-dom, если поставляется как отдельный пакет от поставщика.
<i>xmlreader</i>		php-xmlreader, если поставляется как отдельный пакет от поставщика.
<i>xmlwriter</i>		php-xmlwriter, если поставляется как отдельный пакет от поставщика.
<i>session</i>		php-session, если поставляется как отдельный пакет от поставщика.
<i>sockets</i>		php-net-socket (<i>--enable-sockets</i>). Требуется для поддержки пользовательских скриптов.
<i>mbstring</i>		php-mbstring (<i>--enable-mbstring</i>)
<i>gettext</i>		php-gettext (<i>--with-gettext</i>). Требуется для работы переводов.
<i>ldap</i>		php-ldap. Требуется только, если в веб-интерфейсе используется LDAP аутентификация.
<i>ibm_db2</i>		Требуется, если используется IBM DB2 в качестве базы для Zabbix.
<i>mysqli</i>		Требуется, если используется MySQL в качестве базы для Zabbix.
<i>oci8</i>		Требуется, если используется Oracle в качестве базы для Zabbix.
<i>pgsql</i>		Требуется, если используется PostgreSQL в качестве базы для Zabbix.
Также Zabbix может работать и с предыдущими версиями Apache, MySQL, Oracle, и PostgreSQL. Для шрифтов, кроме DejaVu, который установлен по умолчанию, нужна функция PHP imagerotate . Если функция не установлена, то эти шрифты могут неправильно отображаться при отображении графика. Эта функция доступна только если PHP скомпилирован вместе с GD, которого нет для Debian и для некоторых других дистрибутивов.		

Веб-браузер на стороне клиента

Cookies и Java Script должны быть включены.

Поддерживаются последние версии Google Chrome, Mozilla Firefox, Microsoft Internet Explorer и Opera. Также и другие браузеры (Apple Safari, Konqueror) могут работать с Zabbix.

Реализована одна и та же политика источника для IFrames, что означает, что Zabbix веб-интерфейс нельзя поместить во фреймы на другом домене.

Всё еще, страницы помещенные в Zabbix фреймы будут иметь доступ к веб-интерфейсу Zabbix (посредством JavaScript), если страница, которая помещена во фрейм и веб-интерфейс Zabbix располагаются на одном домене. Такая страница как

<http://secure-zabbix.com/cms/page.html>, если помещена в комплексные экраны на

<http://secure-zabbix.com/zabbix/>, будет иметь полный доступ к Zabbix посредством JS.

Сервер

Обязательные требования нужны всегда. Тогда как опциональные требования требуется только для поддержки определенных функций.

Требование	Статус	Описание
<i>libpcre</i>		PCRE библиотека требуется для поддержки PCRE совместимых регулярных выражений .
<i>libevent</i>	Обязательно	Наименование может отличаться в зависимости от GNU/Linux дистрибутива, например 'libpcre3' или 'libpcre1'. Обратите внимание, что необходима именно PCRE(v8.X), тогда как PCRE2(v10.X) библиотека не используется.
<i>libpthread</i>		Требуется для массового сбора метрик и IPMI мониторинга. Версия 1.4 или более новая.
<i>zlib</i>		Обратите внимание, что для Zabbix прокси это требование опциональное, и нужно только для IPMI мониторинга.
<i>OpenIPMI</i>		Требуется для поддержки IPMI
<i>libssh2</i>		Требуется для поддержки SSH. Версия 1.0 или более новая.
<i>fping</i>		Требуется для элементов данных ICMP пинг .
<i>libcurl</i>	Опционально	Требуется для веб-мониторинга, мониторинга VMware и для SMTP аутентификации. Для SMTP аутентификации, требуется версия 7.20.0 или выше. Также необходима для поддержки Elasticsearch.
<i>libiksemel</i>		Требуется для поддержки Jabber.
<i>libxml2</i>		Требуется для мониторинга VMware.
<i>net-snmp</i>		Требуется для поддержки SNMP.

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 как пакет 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 определенного дистрибутива Linux), ниже приведен список версий библиотек, для которых подтверждена работоспособность Java gateway. Zabbix может также работать с другими версиями этих библиотек.

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

Библиотека	Лицензия	Веб сайт	Комментарии
<i>logback-core-0.9.27.jar</i>	EPL 1.0, LGPL 2.1	http://logback.qos.ch/	Протестировано с версиями 0.9.27, 1.0.13, и 1.1.1.
<i>logback-classic-0.9.27.jar</i>	EPL 1.0, LGPL 2.1	http://logback.qos.ch/	Протестировано с версиями 0.9.27, 1.0.13, и 1.1.1.
<i>slf4j-api-1.6.1.jar</i>	MIT License	http://www.slf4j.org/	Протестировано с версиями 1.6.1, 1.6.6, и 1.7.6.
<i>android-json-4.3_r3.1.jar</i>	Apache License 2.0	https://android.googlesource.com/platform/libcore/+master/json	Протестировано с версиями 2.3.3_r1.1 и 4.3_r3.1. Обратитесь к файлу src/zabbix_java/lib/README для получения инструкций по созданию JAR файла.

Java gateway компилируется и запускается при наличии Java версии 1.6 или выше. Если вы готовите прокомпилированную версию Zabbix gateway для использования ее другими, то рекомендуется использовать для компиляции Java 1.6, в этом случае Zabbix gateway будет работать на всех версиях Java вплоть до самой последней.

Размер базы данных

Данные конфигурации Zabbix требуют фиксированное количество дискового пространства и сильно не увеличиваются.

Размер базы данных Zabbix в основном зависит от следующих переменных, которые определяют объем хранимых данных истории:

- Количество обрабатываемых запросов 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 записей.

- Настройки очистки истории в базе данных

Zabbix хранит значения определенный период времени, обычно несколько недель или месяцев. Каждое новое значение требует определенный объем дискового пространства для данных и индексов.

Таким образом, если требуется сохранение 30 дней истории и каждую секунду в базу данных добавляется 50 новых записей, общее количество значений будет равно примерно $(30*24*3600)* 50 = 129.600.000$ или около 130М значений.

В зависимости от типа базы данных, типа полученных значений (с плавающей точкой, целочисленный, строки, файлы журналов и т.д.) может потребоваться от 40 байт до сотен байт дискового пространства для хранения значения. Обычно одно значение занимает около 90 байт по числовым элементам данных². В нашем случае это означает, что 130М значений потребуют 130М * 90 байт = **10.9ГБ** дискового пространства.

Размер значений текстовых/журнальных элементов данных невозможно предугадать, но вы можете ожидать около 500 байт на значение.

- Настройки очистки динамики изменений в базе данных

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. Предположим, что если требуется хранить динамику изменений в течении 5 лет. Значения 3000 элементов данных потребуют $3000*24*365* 90 = 2.2ГБ$ за год, или **11ГБ** за 5 лет.

- Настройки очистки событий в базе данных

Каждое событие требует около 250 байт дискового пространства¹. Сложно точно оценить количество событий, ежедневно генерируемых Zabbix сервером. В самом худшем случае, мы можем 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 по каждому событию.

Опционально, у событий могут быть теги, каждая запись тега требует приблизительно 100 байт дискового пространства¹. Количество тегов по каждому событию (#теги) зависит от конфигурации. Таким образом каждая запись потребует дополнительно #теги * 100 байт дискового пространства². Это означает, что для того, чтобы хранить события 3 года, потребуется $3*365*24*3600*(250 + 80 + \#теги * 100) = \sim 30\text{ГБ} + \#теги * 100\text{Б}$ дискового пространства.

¹ Больше, когда имеются не-ASCII имена событий, тегов и значения.

² Приблизительные размеры основаны на MySQL и могут отличаться для других баз данных.

Представленная ниже таблица содержит формулы для расчета требуемого пространства жесткого диска для системы мониторинга Zabbix:

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

Общее количество требуемого места на жестком диске рассчитывается:

Конфигурация + История + Динамика изменений + События

После установки Zabbix такое дисковое пространство более НЕ будет использовано сразу. Размер базы данных будет постепенно увеличиваться и остановится по достижении определенного момента, зависящего от настроек очистки базы данных.

Необходимое свободное место на жестком диске, при использовании распределенного мониторинга, рассчитывается аналогичным образом, но оно так же будет зависеть от количества подчиненных нод, связанных с нодой рассчитываемого узла.

Синхронизация времени

Очень важно иметь точную дату и время системы на сервере с запущенным Zabbix. [ntp](#) один из наиболее популярных демонов синхронизации времени хоста с временем на остальных серверах. Настоятельно рекомендуется поддерживать синхронизированное время на всех системах, где работают Zabbix компоненты.

Если время не синхронизировано, Zabbix будет конвертировать штампы времени собранных данных на время Zabbix сервера путем получения штампов времени клиента/сервера после установки соединения для передачи данных и корректировки штампов времени у полученных значений элементов данных при помощи разницы между клиент-сервер временем. Чтобы не усложнять работу

и избежать возможные сложности, задержка в соединении игнорируется. По этой причине задержка в соединении добавляется к штампам времени полученных данных с активных подключений (активный агент, активный прокси, sender) и вычитается из штампов времени полученных данных с пассивных подключений (пассивный прокси). Все остальные проверки выполняются по времени сервера и их штампы времени не корректируются.

Наилучшие практики для безопасной установки Zabbix

Обзор

В этом разделе содержатся рекомендации, которые следует соблюдать для того, чтобы настроить Zabbix безопасным образом.

Практики, описанные здесь, не требуются для работы Zabbix. Они рекомендуются для повышения безопасности системы.

Защищённый пользователь для Zabbix агента

В конфигурации по умолчанию процессы Zabbix сервера и Zabbix агента делят одного 'zabbix' пользователя. Если вы убедитесь, что агент не сможет получить доступ к конфиденциальной информации из конфигурации сервера (например, данные подключения в базе данных), агента необходимо запускать из под другого пользователя:

1. Создайте защищённого пользователя
2. Укажите этого пользователя в [файле конфигурации](#) агента (параметр 'User')
3. Перезапустите агента с правами администратора. Привилегии администратора сбрасываются на указанного пользователя.

Настройка SSL для Zabbix веб-интерфейса

На RHEL/Centos, установите пакет mod_ssl:

```
yum install mod_ssl
```

Создайте папку 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 -out /etc/httpd/ssl/apache-selfsigned.crt
```

Заполните подсказки соответствующим образом. Самая важная строка здесь, которая запрашивает Common Name. Вам необходимо указать имя домена, которое вы хотите связать с вашим сервером. Вместо него вы можете указать публичный IP адрес, если у вас отсутствует имя домена. В этой статье мы будем использовать *example.com*.

```
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 []:
```

Измените конфигурацию 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 сервис Apache, чтобы применить изменения:

```
systemctl restart httpd.service
```

Включение Zabbix в корневом каталоге URL

Добавьте виртуальный хост в конфигурацию Apache и задайте постоянную переадресацию для корневого канала на Zabbix SSL URL. Не забудьте заменить *example.com* на актуальное имя сервера.

```
/etc/httpd/conf/httpd.conf
```

#Добавьте строки

```
<VirtualHost *:*>  
    ServerName example.com  
    Redirect permanent / http://example.com  
</VirtualHost>
```

Перезапустите сервис Apache, чтобы применить изменения:

```
systemctl restart httpd.service
```

Отключение отображения информации о веб-сервере

Рекомендуется отключить все подписи веб-сервера, как часть процесса по улучшению защищенности веб-сервера. По умолчанию веб-сервер раскрывает подпись программного обеспечения:

```
▼ Response Headers      view source  
Cache-Control: no-store, no-cache, must-revalidate  
Connection: Keep-Alive  
Content-Encoding: gzip  
Content-Length: 1160  
Content-Type: text/html; charset=UTF-8  
Keep-Alive: timeout=5, max=100  
Pragma: no-cache  
Server: Apache/2.4.18 (Ubuntu)
```

Эту подпись можно отключить, добавив две строки в файл конфигурации Apache (используется как пример):

```
ServerSignature Off  
ServerTokens Prod
```

Подпись PHP (Заголовок X-Powered-By HTTP) 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.

Дополнительный уровень безопасности можно достичь, используя mod_security (пакет libapache2-mod-security2) с Apache. mod_security позволяет совсем удалить подпись сервера вместо удаления лишь версии из подписи сервера. Подпись можно изменить на любое значение, исправив “SecServerSignature” на любое желаемое значение, после установки mod_security.

Пожалуйста, обратитесь к документации по вашему веб-серверу для того, чтобы узнать как удалять/изменять подписи к программному обеспечению.

Отключение страниц ошибок по умолчанию веб-сервера

Рекомендуется отключить страницы ошибок по умолчанию, чтобы избежать раскрытия информации. По умолчанию веб-сервер использует встроенные страницы ошибок:

Not Found

The requested URL /custom-text was not found on this server.

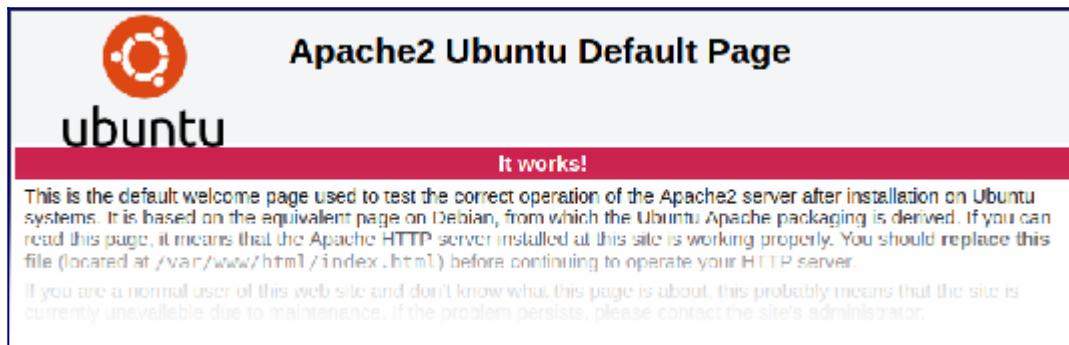
Apache/2.4.18 (Ubuntu) Server at localhost Port 80

Страницы ошибок по умолчанию необходимо заменить/удалить, как часть процесса по улучшению защищенности веб-сервера. Можно использовать директиву “ErrorDocument”, чтобы задать пользовательскую страницу/текст веб-серверу Apache (используется как пример).

Пожалуйста, обратитесь к документации по вашему веб-серверу для того, чтобы узнать как заменять/удалять страницы ошибок по умолчанию.

Удаление тестовой страницы с веб-сервера

Рекомендуется удалить тестовую страницу веб-сервера, чтобы избежать раскрытия информации. По умолчанию, корневой каталог веб-сервера содержит тестовую страницу с именем index.html (Apache2 на Ubuntu используется как пример):



Тестовую страницу необходимо удалить или сделать недоступной, как часть процесса по улучшению защищенности веб-сервера.

4.3 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.

4.3.1 Установка демонов Zabbix

4.3.1.1 Загрузите архив исходных кодов

Перейдите на [страницу загрузки Zabbix](#) и скачайте архив исходных кодов. Как только он будет загружен, распакуйте исходные коды, выполнив:

```
$ tar -zvxf zabbix-4.0.0.tar.gz
```

Укажите корректную версию Zabbix в команде. Она должна совпадать с именем загруженного архива.

4.3.1.2 Создайте аккаунт пользователя

Все процессы демонов Zabbix требуют непrivилегированного пользователя. Если демон Zabbix запущен от аккаунта пользователя без привилегий, то он будет работать под этим пользователем.

Однако, если демон запущен из под аккаунта 'root', демон переключится на аккаунт пользователя 'zabbix', который должен существовать. Для создания такого аккаунта пользователя (принадлежащего к своей группе, "zabbix") на системах на основе RedHat, выполните:

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

На системах на основе Debian, выполните:

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

Zabbix процессам не требуется домашняя директория, поэтому мы не рекомендуем создавать её. Однако, если вы используете некоторый функционал, которому требуется наличие домашней директории (например, хранение учетных данных mysql в \$HOME/.my.cnf), вы вправе создать домашнюю директорию, используя следующие команды. 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
```

Для установки веб-интерфейса Zabbix отдельного аккаунта пользователя не требуется.

Если Zabbix [сервер](#) и [агент](#) работают на одной машине, 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 легкостью получить, например, пароль от базы данных.

Запуск Zabbix из под root, bin или из под любых других аккаунтов со специальными правами является риском для безопасности.

4.3.1.3 Создайте базу данных Zabbix

Для демонов Zabbix [сервера](#) и [прокси](#), а так же для веб-интерфейса Zabbix, требуется база данных. Она не требуется для запуска Zabbix [агента](#).

Для создания схемы базы данных и вставки набора данных предоставляются SQL [скрипты](#). Для базы данных Zabbix proxy требуется только создание схемы данных, в то время как для Zabbix сервера помимо схемы необходимо также установить набор данных.

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

4.3.1.4 Сконфигурируйте исходный код

При конфигурировании исходного кода Zabbix сервера или прокси, вы должны указать используемый тип базы данных. Только один тип базы данных может быть скомпилирован для процессов сервера или прокси единовременно.

Для просмотра всех доступных опций конфигурирования, выполните в папке с извлеченным исходным кодом Zabbix:

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

Для конфигурирования исходных кодов для Zabbix сервера и агента, вы можете выполнить нечто вроде:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp  
--with-libcurl --with-libxml2
```

Опция конфигурации --with-libcurl с cURL 7.20.0 или выше требуется для SMTP аутентификации, поддерживается начиная с Zabbix 3.0.0.

Опции конфигурации --with-libxml2 и --with-libxml2 требуются для мониторинга виртуальных машин, поддерживается начиная с Zabbix 2.2.0.

Начиная с версии Zabbix 3.4, он всегда должен быть скопирован с PCRE библиотекой. Её установка не опциональна. --with-libpcre=[DIR] позволяет указать расположение базовой директории установки, вместо поиска в нескольких стандартных местах расположения libpcre файлов.

Для конфигурирования исходных кодов для Zabbix сервера (с PostgreSQL и т.д.), вы можете выполнить:

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

Для конфигурирования исходных кодов для Zabbix прокси (с SQLite и т.д.), вы можете выполнить:

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

Для конфигурирования исходных кодов для Zabbix агента, вы можете выполнить:

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

Имеется возможность использования флага --enable-static для статической линковки библиотек. Если вы планируете распространять скомпилированные исполняемые файлы на другие сервера, вы должны использовать этот флаг, чтобы приложения могли работать без требуемых библиотек. Флаг --enable-static [не работает под OC Solaris](#).

Использование опции --enable-static не рекомендуется при сборке сервера.

В случае сборки сервера со статической линковкой у вас должны иметься статические версии всех необходимых внешних библиотек.

Утилиты командной строки zabbix_get и zabbix_sender компилируются, если использована опция --enable-agent.

Добавьте опциональный путь к файлу конфигурации

MySQL--with-mysql=/<путь_к_файлу>/mysql_config, чтобы выбрать желаемую библиотеку клиента MySQL, когда имеется необходимость использовать библиотеку, которая не расположена в месте установки по умолчанию.

Бывает полезно там, где установлено несколько версий MySQL или MariaDB установлена вместе с MySQL на одной системе.

Используйте флаг --with-ibm-db2 чтобы указать расположение CLI API.

Используйте флаг --with-oracle чтобы указать расположение OCI API.

Для поддержки шифрования смотрите [Компиляция Zabbix с поддержкой шифрования](#).

4.3.1.5 Соберите и установите всё

Если устанавливаете код взятый из SVN, вы должны сначала выполнить:

```
$ make dbschema
```

```
make install
```

Этот шаг должен быть выполнен пользователем с достаточными правами (как правило 'root', или с помощью `sudo`).

Выполнение `make install` установит исполняемые файлы демонов (`zabbix_server`, `zabbix_agentd`, `zabbix_proxy`) в `/usr/local/sbin` и исполняемые файлы клиентов (`zabbix_get`, `zabbix_sender`) в `/usr/local/bin`.

Для установки в другое место, отличное от `/usr/local`, используйте ключ `--prefix` в предыдущем шаге конфигурирования исходных кодов, например `--prefix=/home/zabbix`. В этом случае исполняемые файлы демонов будут установлены в папку `<предфикс>/sbin`, а утилиты в папку `<предфикс>/bin`. Страницы помощи будут установлены в папку `<предфикс>/share`.

4.3.1.6 Просмотрите и отредактируйте файлы конфигурации

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

Вам нужно сконфигурировать этот файл для каждого хоста 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 он используется).

Остальные параметры подойдут со значениями по умолчанию, если у вас небольшая установка (до десятка наблюдаемых узлов сети). Вы должны изменить параметры по умолчанию, если вы хотите увеличить производительность Zabbix сервера (или прокси). Обратитесь к разделу [Оптимизация производительности](#) для получения подробных сведений.

- 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 он используется).

В случае в SQLite должен быть указан путь к файлу базы данных; пользователь БД и пароль не требуются.

4.3.1.7 Запустите демоны

Запустите `zabbix_server` на стороне сервера.

```
shell> zabbix_server
```

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

Обратитесь к разделу "[Смотрите также](#)" в нижней части этой страницы, чтобы узнать как настроить разделяемую память.

Запустите `zabbix_agentd` на всех наблюдаемых машинах.

```
shell> zabbix_agentd
```

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. Это может произойти в Solaris 8.

Если вы установили Zabbix прокси, запустите zabbix_proxy.

```
shell> zabbix_proxy
```

Установка веб-интерфейса Zabbix

Копирование файлов PHP

Веб-интерфейс Zabbix написан на языке PHP, поэтому чтобы его запустить вам потребуется веб-сервер с поддержкой PHP. Установка производится путем простого копирования PHP файлов в папку HTML веб-сервера.

Общепринятые места размещения папки HTML документов для веб-сервера Apache включают:

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

Рекомендуется использовать подпапку вместо корневой папки HTML. Чтобы создать подпапку и скопировать файлы веб-интерфейса Zabbix, выполните следующие команды, заменив <htdocs> на корректный путь для вашего случая:

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

Если устанавливаете из SVN и планируете использовать любой другой язык, кроме английского, то вы должны генерировать файлы переводов. Для этого выполните:

```
locale/make_mo.sh
```

Требуется утилита `msgfmt` из пакета gettext.

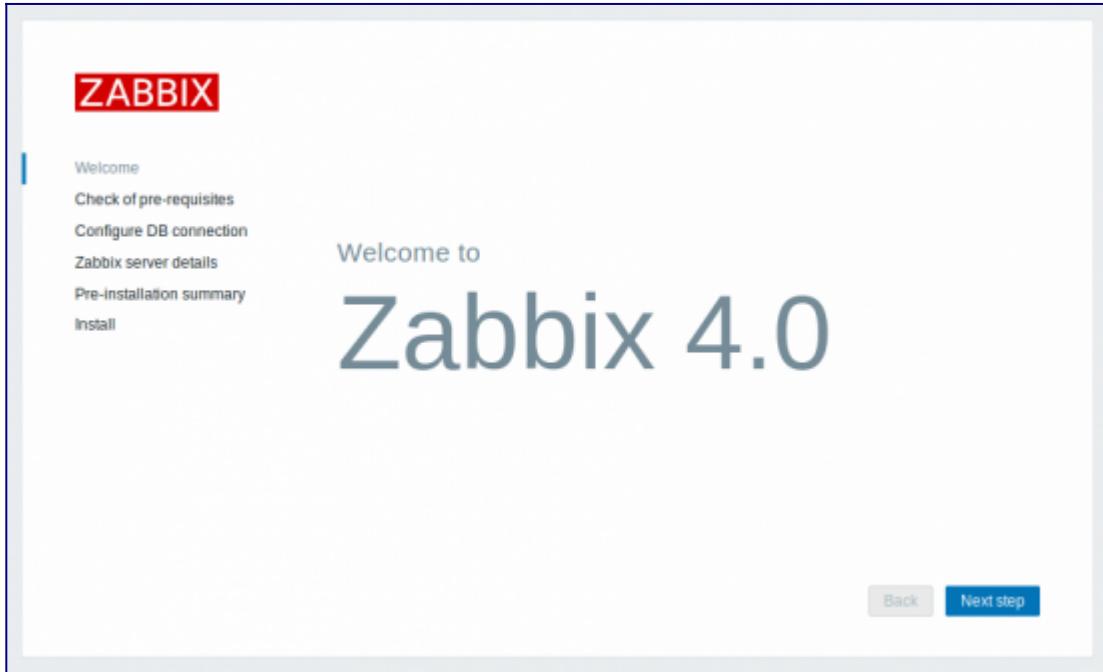
Кроме того, для использования любого другого языка отличного от английского, этот язык должен быть установлен на веб-сервере. Обратитесь к разделу [“Смотрите также”](#) страницы “Профиль пользователя”, чтобы узнать как установить дополнительные языки, если они потребуются.

Установка веб-интерфейса

Шаг 1

Откройте URL Zabbix: `http://<ip_или_имя_сервера>/zabbix`, в вашем браузере.

Вы должны увидеть первую страницу помощника установки веб-интерфейса.



Шаг 2

Убедитесь, что все требования к программному обеспечению выполнены.

This screenshot shows the 'Check of pre-requisites' step. It lists various PHP configuration parameters with their current values and required minimum values. All items are marked as 'OK'. The table includes columns for requirement, current value, required value, and status.

Requirement	Current Value	Required	Status
PHP version	5.4.20	5.4.0	OK
PHP option memory_limit	128M	128M	OK
PHP option post_max_size	32M	16M	OK
PHP option upload_max_filesize	16M	2M	OK
PHP option max_execution_time	600	300	OK
PHP option max_input_time	600	300	OK
PHP time zone	Europe/Riga		OK
PHP databases support	MySQL		OK
PHP bcmath	on		OK

At the bottom right are 'Back' and 'Next step' buttons.

Требование	Минимальное значение	Описание
Версия PHP	5.4.0	
PHP опция <i>memory_limit</i>	128МБ	В php.ini: memory_limit = 128M
PHP опция <i>post_max_size</i>	16МБ	В php.ini: post_max_size = 16M
PHP опция <i>upload_max_filesize</i>	2МБ	В php.ini: upload_max_filesize = 2M
PHP опция <i>max_execution_time</i>	300 секунд (значения 0 и -1 разрешены)	В php.ini: max_execution_time = 300
PHP опция <i>max_input_time</i>	300 секунд (значения 0 и -1 разрешены)	В php.ini: max_input_time = 300
PHP опция	должна быть	В php.ini:

Требование	Минимальное значение	Описание
<i>session.auto_start</i>	отключена	<code>session.auto_start = 0.</code>
<i>Поддержка баз данных</i>	Одна из: MySQL, Oracle, PostgreSQL, IBM DB2	Один из следующих модулей должен быть установлен: <code>mysql, oci8, pgsql, ibm_db2</code>
<i>bcmath</i>		<code>php-bcmath</code>
<i>mbstring</i>		<code>php-mbstring</code>
<i>PHP опция mbstring.func_overload</i>	должна быть отключена	В <code>php.ini</code> : <code>mbstring.func_overload = 0.</code>
<i>PHP опция always_populate_raw_post_data</i>	должна быть равной 1	Требуется только для версий PHP 5.6.0 и новее. В <code>php.ini</code> : <code>always_populate_raw_post_data = -1</code>
<i>sockets</i>		<code>php-net-socket</code> . Требуется для поддержки пользовательских скриптов.
<i>gd</i>	2.0 или выше	<code>php-gd</code> . Расширение PHP GD должно поддерживать PNG изображения (<code>--with-png-dir</code>), JPEG (<code>--with-jpeg-dir</code>) изображения и FreeType 2 (<code>--with-freetype-dir</code>).
<i>libxml</i>	2.6.15	<code>php-xml</code> или <code>php5-dom</code>
<i>xmlwriter</i>		<code>php-xmlwriter</code>
<i>xmlreader</i>		<code>php-xmlreader</code>
<i>ctype</i>		<code>php-ctype</code>
<i>session</i>		<code>php-session</code>
<i>gettext</i>		<code>php-gettext</code> Начиная с Zabbix 2.2.1, расширение PHP gettext более не является обязательным для установки Zabbix. Если <code>gettext</code> не установлен, веб-интерфейс будет работать как обычно, однако, переводы будут недоступны.

В списке могут также присутствовать необязательные требования. Если необязательное требование не удовлетворено, то оно отображается оранжевым цветом и имеет состояние Предупреждение (Warning). Установка может продолжаться, если необязательные требования не удовлетворены.

Если имеется необходимость изменить пользователя или группу пользователей Apache, необходимо проверить права доступа к папке сессий. В противном случае установка Zabbix не сможет продолжиться.

Шаг 3

Укажите информацию для подключения к базе данных. База данных Zabbix должна быть уже создана.

ZABBIX

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Database type	MySQL
Database host	localhost
Database port	0 0 - use default port
Database name	zabbix
User	zabbix
Password	*****

[Back](#) [Next step](#)

Шаг 4

Введите подробные сведения о сервере Zabbix.

ZABBIX

Zabbix server details

Please enter the host name or host IP address and port number of the Zabbix server, as well as the name of the installation (optional).

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
Pre-installation summary
Install

Host	localhost
Port	10051
Name	

[Back](#) [Next step](#)

Шаг 5

Просмотрите результат настроек.

ZABBIX

Pre-installation summary

Please check configuration parameters. If all is correct, press "Next step" button, or "Back" button to change configuration parameters.

Welcome	
Check of pre-requisites	MySQL
Configure DB connection	localhost
Zabbix server details	default
Pre-installation summary	zabbix
Install	zabbix
	Database password *****
Zabbix server	localhost
Zabbix server port	10051
Zabbix server name	

[Back](#) [Next step](#)

Шаг 6

Скачайте файл конфигурации и поместите его в каталог conf/.

ZABBIX

Install

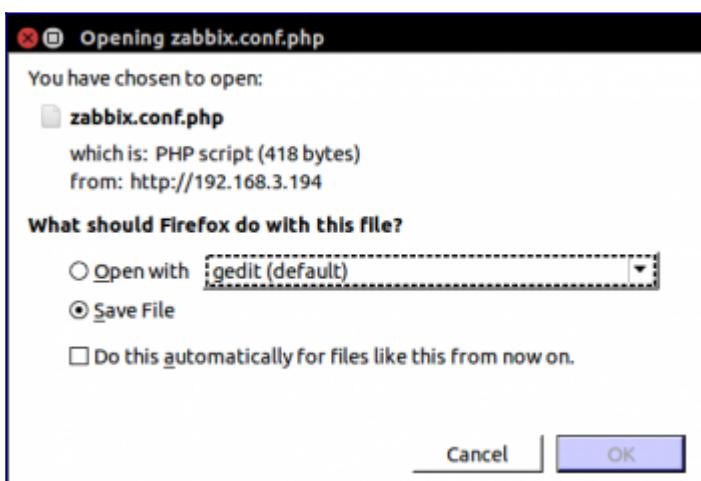
! Details ▾ Cannot create the configuration file.

Unable to create the configuration file.

Alternatively, you can install it manually:

1. [Download the configuration file](#)
2. Save it as "/var/www/html/zabbix/conf/zabbix.conf.php"

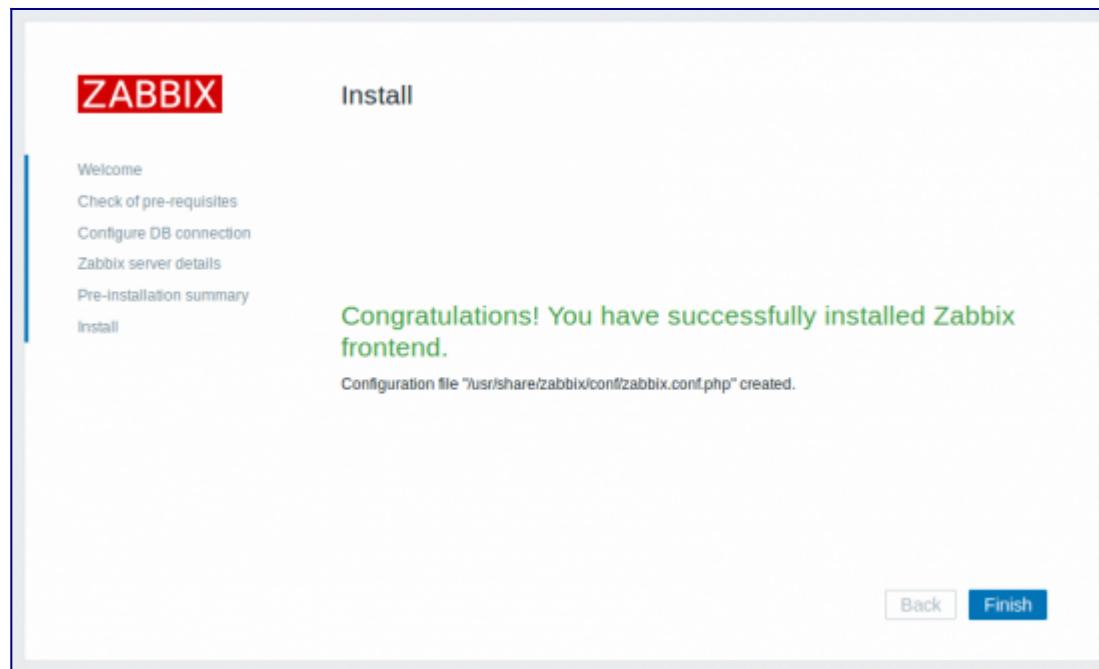
[Back](#) [Finish](#)



В случае, если веб-сервер имеет право на запись в каталог conf/, файл конфигурации будет сохранен автоматически и можно будет сразу же перейти к следующему шагу.

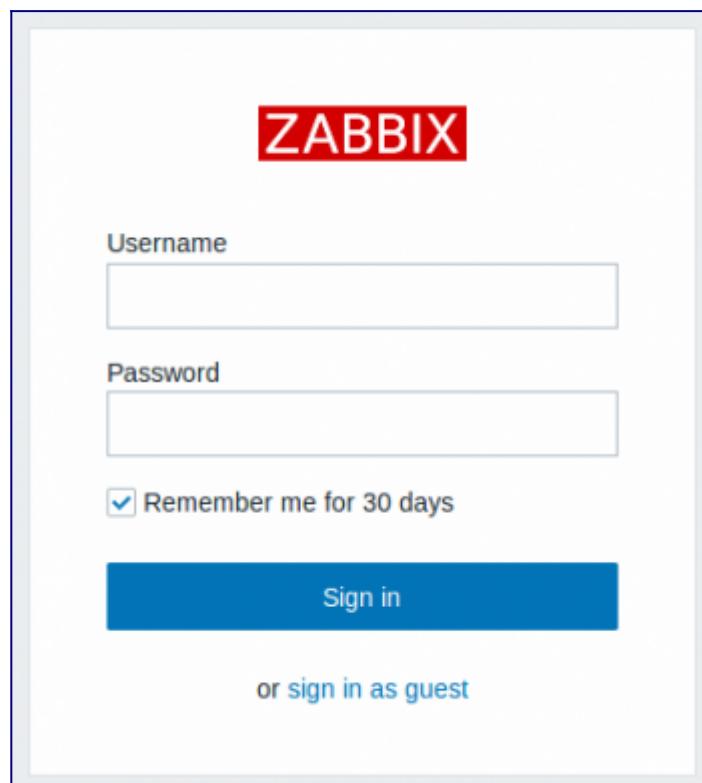
Шаг 7

Завершение установки.



Шаг 8

Веб-интерфейс Zabbix готов! По умолчанию имя пользователя **Admin**, пароль **zabbix**.



Перейти к [началу работы с Zabbix](#).

3 Установка Java gateway

Необходимо установить Java gateway, если вы хотите мониторить JMX приложения. Java gateway легковесный и не требует наличия базы данных.

4. Установка

Чтобы установить его из исходных кодов, сначала [загрузите](#) и извлеките исходные коды из архива.

Чтобы скомпилировать Java gateway, выполните `./configure` скрипт с `--enable-java` опцией. Целесообразно указать опцию `--prefix` для изменения пути установки с `/usr/local` по умолчанию, так как при установке Java gateway будет создано целое дерево директорий, не просто один исполняемый файл.

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

Для компиляции и сборки пакета Java gateway в JAR файл выполните `make`. Обратите внимание, для выполнения этого шага вам потребуется наличие выполняемых файлов `javac` и `jar` в известных системе местах.

```
$ make
```

Теперь у вас имеется `zabbix-java-gateway-$VERSION.jar` файл в `src/zabbix_java/bin`. Если вам удобно запускать Java gateway из `src/zabbix_java` в директории с исходными кодами, тогда вам нужно перейти к инструкциям по настройке и запуску [Java gateway](#). В противном случае убедитесь, что у вас имеются достаточные привилегии и выполните `make install`.

```
$ make install
```

Перейдите на [страницу Java gateway](#) для получения более детальных сведений о том как запустить Java gateway и как настроить Zabbix сервер на работу с ним.

Смотрите также

1. [Как настроить разделяемую память для демонов Zabbix \[en\]](#)

4.4 Установка из пакетов

Из пакетов дистрибутивов

Некоторые популярные дистрибутивы ОС предоставляют пакеты Zabbix. Вы можете использовать эти пакеты для установки Zabbix.

В репозиториях ОС может не быть последней версии Zabbix.

Из официального репозитория Zabbix

Zabbix SIA поставляет официальные RPM и DEB пакеты для:

- [Red Hat Enterprise Linux/CentOS](#)
- [Debian/Ubuntu](#)

Файлы пакетов доступны по адресу [repo.zabbix.com](#). Репозитории yum и apt также доступны на сервере. Пошаговое руководство по установке Zabbix из пакетов представлено в подстраницах этого раздела.

4.4.1 Red Hat Enterprise Linux/CentOS

Обзор

Официальные пакеты Zabbix доступны для RHEL 7, Oracle Linux 7 и CentOS 7. В этой документации мы будем ссылаться на все 3, используя термин RHEL.

Пакеты агента и прокси доступны также и для [RHEL 6](#) и [RHEL 5](#).

Добавление Zabbix репозитория

Установите пакет конфигурации репозитория. Этот пакет содержит файлы конфигурации yum (менеджера пакетов приложений).

RHEL 8:

```
# rpm -Uvh https://repo.zabbix.com/zabbix/4.4/rhel/8/x86_64/zabbix-release-4.4-1.el8.noarch.rpm
```

RHEL 7:

```
# rpm -Uvh https://repo.zabbix.com/zabbix/4.4/rhel/7/x86_64/zabbix-release-4.4-1.el7.noarch.rpm
```

RHEL 6:

```
# rpm -Uvh https://repo.zabbix.com/zabbix/4.4/rhel/6/x86_64/zabbix-release-4.4-1.el6.noarch.rpm
```

RHEL 5:

```
# rpm -Uvh https://repo.zabbix.com/zabbix/4.4/rhel/5/x86_64/zabbix-release-4.4-1.el5.noarch.rpm
```

Предварительные условия установки веб-интерфейса

Веб-интерфейс Zabbix требует дополнительные пакеты, которые отсутствуют в базовой установке. Вам необходимо активировать репозиторий опциональных rpm пакетов в системе, где вы запускаете веб-интерфейс Zabbix:

RHEL 7:

```
# yum-config-manager --enable rhel-7-server-optional-rpms
```

Установка сервера/прокси/веб-интерфейса

Для установки сервера (пакеты доступны для RHEL 7, [устарело для RHEL 6](#)) с MySQL:

```
# yum install zabbix-server-mysql zabbix-web-mysql
```

Для установки Zabbix прокси с MySQL:

```
# yum install zabbix-proxy-mysql
```

Для установки веб-интерфейса Zabbix (пакеты доступны для RHEL 7, [устарело для RHEL 6](#)) с MySQL:

```
# yum install zabbix-web-mysql
```

Замените 'mysql' на 'pgsql' в командах для использования PostgreSQL, или на 'sqlite' для использования SQLite3 (только прокси).

Создание базы данных

Для Zabbix [сервера](#) и [прокси](#) демонов требуется база данных. Она не требуется для работы Zabbix [агента](#).

Если Zabbix сервер и прокси установлены на один хост, их базы данных необходимо создавать с разными именами!

Создайте базу данных, используя имеющиеся [скрипты по созданию базы данных](#) для MySQL или PostgreSQL.

Импорт данных

Теперь импортируйте изначальную схему и данные сервера на MySQL:

```
# zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p zabbix
```

Вам будет предложено ввести пароль от недавно созданной базы данных.

На PostgreSQL:

```
# zcat /usr/share/doc/zabbix-server-pgsql*/create.sql.gz | sudo -u <имя_пользователя> psql zabbix
```

Для прокси импортируйте изначальную схему:

```
# zcat /usr/share/doc/zabbix-proxy-mysql*/schema.sql.gz | mysql -uzabbix -p zabbix
```

Для прокси на PostgreSQL (или SQLite):

```
# zcat /usr/share/doc/zabbix-proxy-pgsql*/schema.sql.gz | sudo -u <имя_пользователя> psql zabbix  
# zcat /usr/share/doc/zabbix-proxy-sqlite3*/schema.sql.gz | sqlite3 zabbix.db
```

Настройка базы данных для Zabbix сервера/прокси

Измените zabbix_server.conf или zabbix_proxy.conf на использование созданной базы данных.

Например:

```
# vi /etc/zabbix/zabbix_server.conf  
DBHost=localhost  
DBName=zabbix  
DBUser=zabbix  
DBPassword=<пароль>
```

В параметре DBPassword используйте пароль от MySQL базы данных Zabbix; пароль пользователя PostgreSQL для PostgreSQL.

Используйте DBHost= для PostgreSQL. Вы возможно, захотите оставить настройку по умолчанию DBHost=localhost (или IP адрес), но в этом случае PostgreSQL будет использовать сетевой сокет при подключении к Zabbix. Смотрите раздел **Настройка SELinux** ниже для получения более подробных сведений.

Запуск процесса Zabbix сервера

Самое время запустить процесс Zabbix сервера:

```
# systemctl start zabbix-server
```

и добавить его в автозагрузку:

RHEL 7 и более новые версии:

```
# systemctl enable zabbix-server
```

RHEL до версии 7:

```
# chkconfig --level 12345 zabbix-server on
```

Замените 'zabbix-server' на 'zabbix-proxy', если вы установили Zabbix прокси.

Настройка Zabbix веб-интерфейса

Для RHEL 7 и поздних версий файл конфигурации Apache для Zabbix веб-интерфейса располагается в /etc/httpd/conf.d/zabbix.conf.

Если вы используете RHEL 6, пожалуйста, ознакомьтесь с разделом об [использовании Zabbix веб-интерфейса на RHEL 6](#), как настроить веб-интерфейс.

Некоторые настройки PHP уже выполнены. Однако, необходимо раскомментировать “date.timezone” настройку и [указать корректный для вас часовой пояс](#).

```
php_value max_execution_time 300
php_value memory_limit 128M
php_value post_max_size 16M
php_value upload_max_filesize 2M
php_value max_input_time 300
php_value max_input_vars 10000
php_value always_populate_raw_post_data -1
# php_value date.timezone Europe/Riga
```

Теперь вы готовы приступить к выполнению [шагов по установке веб-интерфейса](#), которые позволят вам получить доступ к недавно установленному Zabbix.

Официальный репозиторий Zabbix также поставляет пакеты fping, iksemel, libssh2. Эти пакеты располагаются в [non-supported](#) папке.

Настройка SELinux

Если состояние SELinux в принудительном режиме, вам необходимо выполнить следующие команды, чтобы разрешить соединения между Zabbix веб-интерфейсом и сервером:

RHEL 7 и более новые версии:

```
# setsebool -P httpd_can_connect_zabbix on
Если база данных доступна через сеть (включая 'localhost' в случае PostgreSQL) вам
также потребуется разрешить соединение между Zabbix веб-интерфейсом и базой данных:
# setsebool -P httpd_can_network_connect_db on
```

RHEL до версии 7:

```
# setsebool -P httpd_can_network_connect on
# setsebool -P zabbix_can_network on
```

После завершения настройки веб-интерфейса и SELinux вам необходимо перезапустить веб-сервер Apache:

```
# service httpd restart
```

Веб-интерфейс и сервер Zabbix на RHEL 6

Веб-интерфейс Zabbix на RHEL 6 не поддерживается из-за версии PHP. Начиная с версии Zabbix 3.0, требуется PHP версии 5.4.0 и выше, тогда как в RHEL 6 самая последняя версия 5.3.3.

В большинстве случаев Zabbix сервер и веб-интерфейс устанавливаются на одной машине. При обновлении с 2.2 на 3.0 Zabbix сервер выполняет обновление базы данных и веб-интерфейс перестает работать. В этом случае нет возможности откатить изменения в базе данных поэтому пользователи должны обновить PHP используя сторонние пакеты. Это причина по которой Zabbix сервер также считается устаревшим на RHEL 6.

Если вам всё же необходимо использовать Zabbix веб-интерфейс на RHEL 6 и вы уже обновили версию PHP используя сторонние пакеты, тогда вам необходимо сначала активировать репозиторий `zabbix-deprecated`:

- откройте файл `/etc/yum.repos.d/zabbix.repo`
- найдите раздел `[zabbix-deprecated]`
- измените `enabled=1`
- сохраните файл

Вам придётся сделать еще немного ручной настройки. Потому что мы не можем определить версию Apache необходимую для вашего PHP, что делает невозможным для нас предоставить необходимую конфигурацию Apache для веб-интерфейса Zabbix. Мы поставляем 2 конфигурационных файла Apache в пакете Zabbix веб-интерфейса один для Apache 2.2, другой для 2.4, какой вам необходимо использовать с Apache выбирайте сами:

- `httpd22-example.conf`
- `httpd24-example.conf`

Для получения полного пути к файлам выполните:

```
$ rpm -ql zabbix-web | grep example.conf
```

Установка агента

Чтобы установить агента, выполните

```
# yum install zabbix-agent
```

Чтобы запустить агента, выполните:

```
# service zabbix-agent start
```

Установка Java gateway

Необходимо установить [Java gateway](#), если вы хотите мониторить JMX приложения. Java gateway легковесный и не требует наличия базы данных.

Как только требуемый [репозиторий](#) будет добавлен, вы сможете установить Zabbix Java gateway выполнив команду:

```
# yum install zabbix-java-gateway
```

Параметры конфигурации Zabbix Java gateway можно настроить в следующем файле:

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

Для получения более подробных сведенийсмотрите [параметры](#) конфигурации Zabbix Java gateway.

Чтобы запустить Zabbix Java gateway:

```
# service zabbix-java-gateway restart
```

Обратите внимание, что Zabbix сервер [нужно настроить](#) для работы с Java gateway.

Чтобы Zabbix Java gateway автоматически запускался при загрузке системы:

RHEL 7 и более новые версии:

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

RHEL до версии 7:

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

Файл журнала Zabbix Java gateway:

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

Если вы хотите увеличить уровень журналирования, измените следующий файл:

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

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

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

4.4.2 Debian/Ubuntu

Обзор

Официальные пакеты Zabbix доступны для:

- Debian 9 (Stretch)
- Debian 8 (Jessie)
- Ubuntu 18.04 (Bionic Beaver) LTS
- Ubuntu 16.04 (Xenial Xerus) LTS
- Ubuntu 14.04 (Trusty Tahr) LTS

Добавление Zabbix репозитория

Установите пакет конфигурации репозитория. Этот пакет содержит файлы конфигурации apt (менеджера пакетов приложений).

Для Debian 9, выполните следующие команды:

Обратите внимание! Для Debian 8, в командах замените 'stretch' на 'jessie'.

```
# wget https://repo.zabbix.com/zabbix/4.2/debian/pool/main/z/zabbix-release/zabbix-
release_4.2-1+stretch_all.deb
# dpkg -i zabbix-release_4.2-1+stretch_all.deb
# apt update
```

Для Ubuntu **Ubuntu 18.04 (bionic)**, выполните следующие команды:

```
# wget https://repo.zabbix.com/zabbix/4.2/ubuntu/pool/main/z/zabbix-release/zabbix-
release_4.2-1+bionic_all.deb
# dpkg -i zabbix-release_4.2-1+bionic_all.deb
# apt update
```

- Для Ubuntu 16.04 LTS измените в командах 'bionic' на 'xenial'.
- Для Ubuntu 14.04 LTS измените в командах 'bionic' на 'trusty'.

Установка сервера/прокси/веб-интерфейса

Для установки Zabbix сервера с поддержкой MySQL:

```
# apt install zabbix-server-mysql
```

Для установки Zabbix прокси с поддержкой MySQL:

```
# apt install zabbix-proxy-mysql
```

Для установки Zabbix веб-интерфейса:

```
# apt install zabbix-frontend-php
```

Замените 'mysql' в командах на 'pgsql' при использовании PostgreSQL, или на 'sqlite' при использовании SQLite3 (только прокси).

Создание базы данных

Для Zabbix [сервера](#) и [прокси](#) демонов требуется база данных. Она не требуется для работы Zabbix [агента](#).

Если Zabbix сервер и прокси установлены на один хост, их базы данных необходимо создавать с разными именами!

Создайте базу данных, используя инструкции для [MySQL](#) или [PostgreSQL](#).

Импорт данных

Теперь импортируйте изначальную схему и данные сервера на MySQL:

```
# zcat /usr/share/doc/zabbix-server-mysql/create.sql.gz | mysql -uzabbix -p zabbix
```

Вам будет предложено ввести пароль от недавно созданной базы данных.

На PostgreSQL:

```
# zcat /usr/share/doc/zabbix-server-pgsql/create.sql.gz | psql -U <username> zabbix
```

Для прокси импортируйте изначальную схему:

```
# zcat /usr/share/doc/zabbix-proxy-mysql/create.sql.gz | mysql -uzabbix -p zabbix
```

Для прокси на PostgreSQL (или SQLite):

```
# zcat /usr/share/doc/zabbix-proxy-pgsql/create.sql.gz | psql -U <username> zabbix
# zcat /usr/share/doc/zabbix-proxy-sqlite/create.sql.gz | sqlite3 zabbix.db
```

Настройка базы данных для Zabbix сервера/прокси

Измените zabbix_server.conf или zabbix_proxy.conf для использования созданной базы данных.

Например:

```
# vi /etc/zabbix/zabbix_server.conf
DBHost=localhost
DBName=zabbix
DBUser=zabbix
DBPassword=<пароль>
```

В параметре DBPassword используйте пароль от MySQL базы данных Zabbix; пароль пользователя PostgreSQL для PostgreSQL.

Используйте DBHost= для PostgreSQL. Вы возможно, захотите оставить настройку по умолчанию DBHost=localhost (или IP адрес), но в этом случае PostgreSQL будет использовать сетевой сокет при подключении к Zabbix. Обратитесь к [соответствующему разделу](#) по RHEL/CentOS для получения детальных инструкций.

Запуск процесса Zabbix сервера

Самое время запустить процесс Zabbix сервера и добавить его в автозагрузку:

```
# service zabbix-server start  
# update-rc.d zabbix-server enable
```

Замените 'zabbix-server' на 'zabbix-proxy' для запуска процесса Zabbix прокси.

Настройка SELinux

Обратитесь к [соответствующему разделу](#) по RHEL/CentOS.

После завершения настройки веб-интерфейса и SELinux вам необходимо перезапустить веб-сервер Apache:

```
# service apache2 restart
```

Настройка веб-интерфейса

Файл конфигурации Apache для Zabbix веб-интерфейса располагается в /etc/apache2/conf-enabled/zabbix.conf. Некоторые настройки PHP уже выполнены. Однако, необходимо раскомментировать "date.timezone" настройку и [указать корректный для вас часовой пояс](#).

```
php_value max_execution_time 300  
php_value memory_limit 128M  
php_value post_max_size 16M  
php_value upload_max_filesize 2M  
php_value max_input_time 300  
php_value max_input_vars 10000  
php_value always_populate_raw_post_data -1  
# php_value date.timezone Europe/Riga
```

Теперь вы готовы приступить к выполнению [шагов по установке веб-интерфейса](#), которые позволят вам получить доступ к недавно установленному Zabbix.

Установка агента

Для установки агента выполните

```
# apt install zabbix-agent
```

Для запуска агента выполните:

```
# service zabbix-agent start
```

Установка Java gateway

Необходимо установить [Java gateway](#), если вы хотите мониторить JMX приложения. Java gateway легковесный и не требует наличия базы данных.

Как только требуемый [репозиторий](#) будет добавлен, вы сможете установить Zabbix Java gateway выполнив команду:

```
# apt install zabbix-java-gateway
```

Параметры конфигурации Zabbix Java gateway можно настроить в следующем файле:

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

Для получения более подробных сведенийсмотрите [параметры](#) конфигурации Zabbix Java gateway.

Чтобы запустить Zabbix Java gateway:

```
# service zabbix-java-gateway restart
```

Обратите внимание, что Zabbix сервер [нужно настроить](#) для работы с Java gateway.

Чтобы Zabbix Java gateway автоматически запускался при загрузке системы:

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

Файл журнала Zabbix Java gateway:

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

Если вы хотите увеличить уровень журналирования, измените следующий файл:

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

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

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

4.5 Установка из контейнеров

Docker

Zabbix предоставляет [Docker](#) образы для каждого компонента Zabbix как переносимые и самодостаточные контейнеры для ускорения развертывания и процедуры обновления.

Zabbix компоненты поставляются с поддержкой MySQL и PostgreSQL баз данных, поддержкой Apache2 и Nginx веб-серверов. Эти образы разделены на несколько различных образом.

Базовые образы Docker

Zabbix компоненты поставляются на основе Ubuntu, Alpine Linux и CentOS образов:

Образ Версия

[alpine](#) 3.4

[ubuntu](#) trusty

[centos](#) latest

Все образы сконфигурированы таким образом, что будут пересозданы в случае, если базовые образы обновятся.

Образы Zabbix агента, Zabbix прокси, Zabbix сервера и Zabbix готового решения основаны на Alpine Linux версии 3.4, все остальные образы основаны на Alpine Linux 3.7.

Исходные файлы Docker

Каждый может следить за изменениями Docker файлов в [официальном репозитории](#) Zabbix на [github.com](#). Вы можете сделать ответвления от проекта или создавать свои собственные образы на основе официальных Docker файлов.

Структура

Все компоненты Zabbix доступны в следующих Docker репозиториях:

- Готовое решение Zabbix с поддержкой MySQL базы данных и Nginx веб-сервером - [zabbix/zabbix-appliance](#)
- Zabbix агент - [zabbix/zabbix-agent](#)
- Zabbix сервер
 - Zabbix сервер с поддержкой MySQL базы данных - [zabbix/zabbix-server-mysql](#)
 - Zabbix сервер с поддержкой PostgreSQL базы данных - [zabbix/zabbix-server-pgsql](#)
- Zabbix веб-интерфейс
 - Zabbix веб-интерфейс на основе Apache2 веб-сервера с поддержкой MySQL базы данных - [zabbix/zabbix-web-apache-mysql](#)
 - Zabbix веб-интерфейс на основе Apache2 веб-сервера с поддержкой PostgreSQL базы данных - [zabbix/zabbix-web-apache-pgsql](#)
 - Zabbix веб-интерфейс на основе Nginx веб-сервера с поддержкой MySQL базы данных - [zabbix/zabbix-web-nginx-mysql](#)
 - Zabbix веб-интерфейс на основе Nginx веб-сервера с поддержкой PostgreSQL базы данных - [zabbix/zabbix-web-nginx-pgsql](#)
- Zabbix прокси
 - Zabbix прокси с поддержкой SQLite3 базы данных - [zabbix/zabbix-proxy-sqlite3](#)
 - Zabbix прокси с поддержкой MySQL базы данных - [zabbix/zabbix-proxy-mysql](#)
- Zabbix Java Gateway - [zabbix/zabbix-java-gateway](#)

Дополнительно имеется поддержка SNMP трапов. Она поставляется как отдельный репозиторий ([zabbix/zabbix-snmptraps](#)) основанный только на Ubuntu Trusty. Его можно связать с Zabbix сервером или Zabbix прокси.

Версии

Каждый репозиторий Zabbix компонент содержит следующие тэги:

- **latest** - последняя стабильная версия Zabbix компоненты на основе образа Alpine Linux
- **alpine-latest** - последняя стабильная версия Zabbix компоненты на основе образа Alpine Linux
- **ubuntu-latest** - последняя стабильная версия Zabbix компоненты на основе образа Ubuntu
- **alpine-4.0-latest** - последняя минорная версия Zabbix 4.0 компоненты на основе образа Alpine Linux
- **ubuntu-4.0-latest** - последняя минорная версия Zabbix 4.0 компоненты на основе образа Ubuntu
- **alpine-4.0.*** - различные минорные версии Zabbix 4.0 компоненты на основе образа Alpine Linux, где * - минорная версия Zabbix компоненты
- **ubuntu-4.0.*** - различные минорные версии Zabbix 4.0 компоненты на основе образа Ubuntu, где * - минорная версия Zabbix компоненты

Использование

Переменные окружения

Все образы Zabbix компонент предусматривают наличие переменных окружения для управления конфигурацией. Эти переменные окружения перечислены в репозиториях каждого компонента. Переменные окружения являются опциями из файлов конфигурации Zabbix, но имеют немного другое наименование. Например, `ZBX_LOGSLOWQUERIES` идентичен `LogSlowQueries` из Zabbix сервера и Zabbix прокси файлов конфигурации.

Некоторые опции конфигурации не разрешены для изменения. Например, `PIDFile` и `LogType`.

Некоторые компоненты имеют специфичные переменные окружения, которые отсутствуют в официальных файлах конфигурации Zabbix:

Переменная	Компоненты	Описание
DB_SERVER_HOST	Сервер Прокси Веб-интерфейс	Переменная является IP адресом или DNS именем MySQL или PostgreSQL сервера. По умолчанию, значение mysql-server или postgres-server для MySQL или PostgreSQL соответственно
DB_SERVER_PORT	Сервер Прокси Веб-интерфейс	Переменная является портом MySQL или PostgreSQL сервера. По умолчанию, значение '3306' или '5432' соответственно.
MYSQL_USER	Сервер Proxy Веб-интерфейс	Имя пользователя к базе данных MySQL. По умолчанию, значение 'zabbix'.
MYSQL_PASSWORD	Сервер Прокси Веб-интерфейс	Пароль к базе данных MySQL. По умолчанию, значение 'zabbix'.
MYSQL_DATABASE	Сервер Прокси Веб-интерфейс	Имя базы данных Zabbix. По умолчанию, значение 'zabbix' для Zabbix сервера и 'zabbix_proxy' для Zabbix прокси.
POSTGRES_USER	Сервер Веб-интерфейс	Имя пользователя к базе данных PostgreSQL. По умолчанию, значение 'zabbix'.
POSTGRES_PASSWORD	Сервер Веб-интерфейс	Пароль к базе данных PostgreSQL. По умолчанию, значение 'zabbix'.
POSTGRES_DB	Сервер Веб-интерфейс	Имя базы данных Zabbix. По умолчанию, значение 'zabbix' для Zabbix сервера и 'zabbix_proxy' для Zabbix прокси.
TZ	Веб-интерфейс	Часовой пояс в PHP формате. Полный список поддерживаемых часовых поясов доступен на php.net . По умолчанию, значение 'Europe/Riga'.
ZBX_SERVER_NAME	Веб-интерфейс	Видимое имя Zabbix инсталляции в верхнем правом углу веб-интерфейса. По умолчанию, значение 'Zabbix Docker'
ZBX_JAVAGATEWAY_ENABLE	Сервер Прокси	Включение взаимодействия с Zabbix Java gateway для сбора проверок связанных с Java. По умолчанию, значение "false"
ZBX_ENABLE_SNMP_TRAPS	Сервер Прокси	Включение функции SNMP трапов. Эта переменная требует zabbix-snmptraps экземпляр и разделяемый том /var/lib/zabbix/snmptraps с Zabbix сервером или Zabbix прокси.

Тома

Образы позволяют использовать некоторые точки мониторинга. Такие точки мониторинга различны и зависят от типа Zabbix компонента:

Том	Описание
Zabbix агент	
/etc/zabbix/zabbix_agentd.d	Том позволяет включать *.conf файлы и расширять Zabbix агент используя функционал UserParameter
/var/lib/zabbix/modules	Том позволяет загружать дополнительные модули и расширять Zabbix агент используя функционал LoadModule
/var/lib/zabbix/enc	Том используется для хранения файлов связанных с TLS функционалом. Эти имена файлов указываются при помощи переменных окружения ZBX_TLSCAFILE, ZBX_TLSCRLFILE, ZBX_TLSKEY_FILE и ZBX_TLSPSKFILE

Zabbix сервер

/usr/lib/zabbix/alertscripts	Том используется для пользовательских скриптов оповещения. Том является параметром <code>AlertScriptsPath</code> в zabbix_server.conf
/usr/lib/zabbix/externalscripts	Том используется для внешних проверок . Том является параметром <code>ExternalScripts</code> в zabbix_server.conf
/var/lib/zabbix/modules	Том позволяет загружать дополнительные модули и расширять Zabbix сервер используя функционал LoadModule
/var/lib/zabbix/enc	Том используется для хранения файлов связанных с TLS функционалом. Эти имена файлов указываются при помощи переменных окружения <code>ZBX_TLSCAFILE</code> , <code>ZBX_TLSCRLFILE</code> , <code>ZBX_TLSKEY_FILE</code> и <code>ZBX_TLSPSKFILE</code>
/var/lib/zabbix/ssl/certs	Том используется для размещения файлов клиентских SSL сертификатов для аутентификации клиентов. Этот том является аналогом параметра <code>SSLCertLocation</code> в zabbix_server.conf
/var/lib/zabbix/ssl/keys	Том используется для размещения файлов приватных SSL ключей для аутентификации клиентов. Этот том является аналогом параметра <code>SSLKeyLocation</code> в zabbix_server.conf
/var/lib/zabbix/ssl/ssl_ca	Том используется для размещения файлов центра сертификации (CA) для верификации SSL сертификатов сервера. Этот том является аналогом параметра <code>SSLCALocation</code> в zabbix_server.conf
/var/lib/zabbix/snmptraps	Том используется как размещение <code>snmptraps.log</code> файла. Этот том может быть использован совместо с <code>zabbix-snmptraps</code> контейнером и унаследован из <code>volumes_from</code> опции Docker при создании нового экземпляра Zabbix сервера. Функцию обработки SNMP трапов можно включить использовав совместный том и переключив <code>ZBX_ENABLE_SNMP_TRAPS</code> переменную окружения в 'true'
/var/lib/zabbix/mibs	Том позволяет добавлять новые MIB файлы. Функция не поддерживает подпапки, все MIB файлы должны быть помещены в <code>/var/lib/zabbix/mibs</code>

Zabbix прокси

/usr/lib/zabbix/externalscripts	Том используется для внешних проверок . Том является параметром <code>ExternalScripts</code> в zabbix_proxy.conf
/var/lib/zabbix/modules	Том позволяет загружать дополнительные модули и расширять Zabbix прокси используя функционал LoadModule
/var/lib/zabbix/enc	Том используется для хранения файлов связанных с TLS функционалом. Эти имена файлов указываются при помощи переменных окружения <code>ZBX_TLSCAFILE</code> , <code>ZBX_TLSCRLFILE</code> , <code>ZBX_TLSKEY_FILE</code> и <code>ZBX_TLSPSKFILE</code>
/var/lib/zabbix/ssl/certs	Том используется для размещения файлов клиентских SSL сертификатов для аутентификации клиентов. Этот том является аналогом параметра <code>SSLCertLocation</code> в zabbix_proxy.conf
/var/lib/zabbix/ssl/keys	Том используется для размещения файлов клиентских SSL сертификатов для аутентификации клиентов. Этот том является аналогом параметра <code>SSLCertLocation</code> в zabbix_proxy.conf
/var/lib/zabbix/ssl/ssl_ca	Том используется для размещения файлов центра сертификации (CA) для верификации SSL сертификатов сервера. Этот том является аналогом параметра <code>SSLCALocation</code> в zabbix_proxy.conf
/var/lib/zabbix/snmptraps	Том используется как размещение <code>snmptraps.log</code> файла. Этот том может быть использован совместо с <code>zabbix-snmptraps</code> контейнером и унаследован из <code>volumes_from</code> опции Docker при создании нового экземпляра Zabbix прокси. Функцию обработки SNMP трапов можно включить использовав совместный том и переключив <code>ZBX_ENABLE_SNMP_TRAPS</code> переменную окружения в 'true'
/var/lib/zabbix/mibs	Том позволяет добавлять новые MIB файлы. Функция не поддерживает подпапки, все MIB файлы должны быть помещены в <code>/var/lib/zabbix/mibs</code>

Zabbix веб-интерфейс на основе Apache2 веб-сервера

Том позволяет активировать HTTPS для Zabbix веб-интерфейса. Этот том должен содержать два файла `ssl.crt` и `ssl.key`, подготовленные для Apache2 SSL соединений

Zabbix веб-интерфейс на основе Nginx веб-сервера

Том позволяет активировать HTTPS для Zabbix веб-интерфейса. Этот том должен содержать три файла `ssl.crt`, `ssl.key` и `dhparam.pem` подготовленные для Nginx SSL соединений

Zabbix snmptraps

`/var/lib/zabbix/snmptraps` Том содержит файл журнала с именем `snmptraps.log` с полученными SNMP трапами
`/var/lib/zabbix/mibs` Том позволяет добавлять новые MIB файлы. Функция не поддерживает подпапки, все MIB файлы должны быть помещены в `/var/lib/zabbix/mibs`

Для получения дополнительной информации обратитесь к официальными репозиториям Zabbix в Docker Hub.

Примеры использования

Пример 1

Этот пример демонстрирует как запустить готовое решение Zabbix со встроенными MySQL базой данных, Zabbix сервером, Zabbix веб-интерфейсов на основе Nginx веб-сервера и Zabbix Java gateway.

```
# docker run --name zabbix-appliance -t \
  -p 10051:10051 \
  -p 80:80 \
  -d zabbix/zabbix-appliance:latest
```

Экземпляр готового решения Zabbix раскрывает 10051/TCP порт (Zabbix траппера) и 80/TCP порт (HTTP) на хост машину.

Пример 2

Этот пример демонстрирует как запустить Zabbix сервер с поддержкой MySQL базы данных, Zabbix веб-интерфейсом на основе Nginx веб-сервера и Zabbix Java gateway.

1. Запустите пустой экземпляр MySQL сервера

```
# 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" \
  -d mysql:5.7 \
  --character-set-server=utf8 --collation-server=utf8_bin
```

2. Запустите экземпляр Zabbix Java gateway

```
# docker run --name zabbix-java-gateway -t \
  -d zabbix/zabbix-java-gateway:latest
```

3. Запустите экземпляр Zabbix сервера и соедините этот экземпляр с недавно созданным экземпляром MySQL сервера

```
# 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 MYSQL_ROOT_PASSWORD="root_pwd" \
  -e ZBX_JAVAGATEWAY="zabbix-java-gateway" \
  --link mysql-server:mysql \
  --link zabbix-java-gateway:zabbix-java-gateway \
```

4. Установка

```
-p 10051:10051 \
-d zabbix/zabbix-server-mysql:latest
```

Экземпляр Zabbix сервера раскрывает 10051/TCP порт (Zabbix траппера) на хост машину.

4. Запустите Zabbix веб-интерфейс и соедините этот экземпляр с недавно созданными экземплярами MySQL сервера и Zabbix сервера

```
# docker run --name zabbix-web-nginx-mysql -t \
-e DB_SERVER_HOST="mysql-server" \
-e MYSQL_DATABASE="zabbix" \
-e MYSQL_USER="zabbix" \
-e MYSQL_PASSWORD="zabbix_pwd" \
-e MYSQL_ROOT_PASSWORD="root_pwd" \
--link mysql-server:mysql \
--link zabbix-server-mysql:zabbix-server \
-p 80:80 \
-d zabbix/zabbix-web-nginx-mysql:latest
```

Экземпляр Zabbix веб-интерфейс раскрывает 80/TCP порт (HTTP) на хост машину.

Пример 3

Этот пример демонстрирует как запустить Zabbix сервер с поддержкой PostgreSQL базы данных, Zabbix веб-интерфейсом на основе Nginx веб-сервера и с функцией приёма SNMP трапов.

1. Запустите пустой экземпляр PostgreSQL сервера

```
# docker run --name postgres-server -t \
-e POSTGRES_USER="zabbix" \
-e POSTGRES_PASSWORD="zabbix" \
-e POSTGRES_DB="zabbix_pwd" \
-d postgres:latest
```

2. Запустите экземпляр Zabbix snmptraps

```
# 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 \
-p 162:162/udp \
-d zabbix/zabbix-snmptraps:latest
```

Экземпляр Zabbix snmptraps раскрывает 162/UDP порт (SNMP трапы) на хост машину.

3. Запустите экземпляр Zabbix сервера и соедините этот экземпляр с недавно созданным экземпляром PostgreSQL сервера

```
# docker run --name zabbix-server-pgsql -t \
-e DB_SERVER_HOST="postgres-server" \
-e POSTGRES_USER="zabbix" \
-e POSTGRES_PASSWORD="zabbix" \
-e POSTGRES_DB="zabbix_pwd" \
-e ZBX_ENABLE_SNMP_TRAPS="true" \
--link postgres-server:postgres \
-p 10051:10051 \
--volumes-from zabbix-snmptraps \
-d zabbix/zabbix-server-pgsql:latest
```

Экземпляр Zabbix сервера раскрывает 10051/TCP порт (Zabbix траппера) на хост машину.

4. Запустите Zabbix веб-интерфейс и соедините этот экземпляр с недавно созданными экземплярами PostgreSQL сервера и Zabbix сервера

```
# docker run --name zabbix-web-nginx-pgsql -t \
-e DB_SERVER_HOST="postgres-server" \
-e POSTGRES_USER="zabbix" \
```

```

-e POSTGRES_PASSWORD="zabbix" \
-e POSTGRES_DB="zabbix_pwd" \
--link postgres-server:postgres \
--link zabbix-server-pgsql:zabbix-server \
-p 443:443 \
-v /etc/ssl/nginx:/etc/ssl/nginx:ro \
-d zabbix/zabbix-web-nginx-pgsql:latest

```

Экземпляр Zabbix веб-интерфейса раскрывает 443/TCP порт (HTTPS) на хост машину. Папка `/etc/ssl/nginx` должна содержать сертификат с требуемым именем.

Docker Compose

Zabbix также поставляет файлы наборов для определения и запуска нескольких контейнеров Zabbix компонент в Docker. Такие файлов наборов доступны в официальном репозитории Zabbix docker на [github.com: https://github.com/zabbix/zabbix-docker](https://github.com/zabbix/zabbix-docker). Они добавлены как примеры, эти файлы перегружены. Например, они содержат прокси с поддержкой MySQL и SQLite3.

Имеется несколько различных версий файлов наборов:

Имя файла	Описание
<code>docker-compose_v3_alpine_mysql_localhost.yaml</code>	Файл набора запускает последнюю версию компонент Zabbix 4.0 на Alpine Linux с поддержкой MySQL базы данных.
<code>docker-compose_v3_alpine_mysql_local.yaml</code>	Файл набора собирает локально последнюю версию Zabbix 4.0 и запускает компоненты Zabbix на Alpine Linux с поддержкой MySQL базы данных.
<code>docker-compose_v3_alpine_pgsql_localhost.yaml</code>	Файл набора запускает последнюю версию компонент Zabbix 4.0 на Alpine Linux с поддержкой PostgreSQL базы данных.
<code>docker-compose_v3_alpine_pgsql_local.yaml</code>	Файл набора собирает локально последнюю версию Zabbix 4.0 и запускает компоненты Zabbix на Alpine Linux с поддержкой PostgreSQL базы данных.
<code>docker-compose_v3_centos_mysql_localhost.yaml</code>	Файл набора запускает последнюю версию компонент Zabbix 4.0 на CentOS 7 с поддержкой MySQL базы данных.
<code>docker-compose_v3_centos_mysql_local.yaml</code>	Файл набора собирает локально последнюю версию Zabbix 4.0 и запускает компоненты Zabbix на CentOS 7 с поддержкой MySQL базы данных.
<code>docker-compose_v3_centos_pgsql_localhost.yaml</code>	Файл набора запускает последнюю версию компонент Zabbix 4.0 на CentOS 7 с поддержкой PostgreSQL базы данных.
<code>docker-compose_v3_centos_pgsql_local.yaml</code>	Файл набора собирает локально последнюю версию Zabbix 4.0 и запускает компоненты Zabbix на CentOS 7 с поддержкой PostgreSQL базы данных.
<code>docker-compose_v3_ubuntu_mysql_localhost.yaml</code>	Файл набора запускает последнюю версию компонент Zabbix 4.0 на Ubuntu 14.04 с поддержкой MySQL базы данных.
<code>docker-compose_v3_ubuntu_mysql_local.yaml</code>	Файл набора собирает локально последнюю версию Zabbix 4.0 и запускает компоненты Zabbix на Ubuntu 14.04 с поддержкой MySQL базы данных.
<code>docker-compose_v3_ubuntu_pgsql_localhost.yaml</code>	Файл набора запускает последнюю версию компонент Zabbix 4.0 на Ubuntu 14.04 с поддержкой PostgreSQL базы данных.
<code>docker-compose_v3_ubuntu_pgsql_local.yaml</code>	Файл набора собирает локально последнюю версию Zabbix 4.0 и запускает компоненты Zabbix на Ubuntu 14.04 с поддержкой PostgreSQL базы данных.

Доступные файлы наборов Docker поддерживают как версию 2, так и версию 3 Docker Compose.

Хранилище

Файлы наборов сконфигурированы для поддержки локального хранилища на хост машине. Docker Compose создаст директорию `zbx_env` в папке с файлом набора, когда вы запустите компоненты Zabbix с использованием файла набора. Директория будет содержать ту же структуру, что и описано выше в разделе [Тома](#) и директорию для хранения базы данных.

Также в файлах наборов имеются тома в режиме только чтения для `/etc/localtime` и `/etc/timezone` файлов.

Файлы окружения

В той же директории с файлами наборов на github.com вы можете найти файлы с переменными окружения по умолчанию для каждой компоненты из файла наборов. Эти файлы окружения называются примерно так `.env_<тип компоненты>`.

Примеры

Пример 1

```
# docker-compose -f ./docker-compose_v3_alpine_mysql_latest.yaml up -d
```

Эта команда загрузит последние образы Zabbix 4.0 для каждой Zabbix компоненты и затем запустит их в `detach` режиме.

Не забудьте загрузить `.env_<тип компонента>` файлы с github.com официального репозитория Zabbix вместе с файлами наборов.

Пример 2

```
# docker-compose -f ./docker-compose_v3_ubuntu_mysql_local.yaml up -d
```

Эта команда загрузит базовый образ Ubuntu 14.04, затем соберет Zabbix 4.0 компоненты локально и запустит их в `detach` режиме.

4.6 Процедура обновления

Обзор

Этот раздел описывает информацию об обновлении для Zabbix 4.0:

- с использованием пакетов:
 - для [Red Hat Enterprise Linux/CentOS](#)
 - для [Debian/Ubuntu](#)
- с использованием [исходного кода](#)

Обновление напрямую до Zabbix 4.0 возможно с Zabbix **3.4.x, 3.2.x, 3.0.x, 2.4.x, 2.2.x и 2.0.x**. При обновлении с более ранних версий обратитесь к документации Zabbix по 2.0 и более ранним версиям.

Обновление с использованием пакетов

Обзор

В этом разделе описаны шаги, которые требуется выполнить для успешного [обновления](#) с использованием официальных RPM и DEB пакетов, которые поставляются Zabbix для:

- [Red Hat Enterprise Linux/CentOS](#)
- [Debian/Ubuntu](#)

1 Red Hat Enterprise Linux/CentOS

Обзор

Этот раздел описывает требуемые шаги для успешного [обновления](#) с Zabbix 3.4.x до Zabbix 4.0.x с использованием официальных пакетов Zabbix для Red Hat Enterprise Linux/CentOS.

В то время как обновление Zabbix агентов не является обязательным шагом (но рекомендуемым), Zabbix сервер и прокси должны быть [одной мажорной версии](#). Поэтому, при наличии сервер-прокси инсталляции, Zabbix сервер и все прокси должны быть остановлены и обновлены.

Чтобы минимизировать время простоя и потери данных в процессе обновления, рекомендуется остановить и обновить Zabbix сервер и один за другим затем остановить, обновить и запустить Zabbix прокси. Когда все прокси будут обновлены, запустите Zabbix сервер. Пока Zabbix сервер будет остановлен, запущенные прокси будут продолжать сбор и хранение данных и отправят эти данные на Zabbix сервер, когда сервер запустится и заработает. Любые оповещения о проблемах в процессе простоя Zabbix сервера будут сгенерированы только после того как обновленный сервер запустится.

Обратите внимание, что при наличии прокси с SQLite базой данных, данные истории этих прокси будут потеряны до обновления, так как обновление файла базы данных SQLite не поддерживается и этот файл необходимо удалить вручную. Когда прокси запускается в первый раз и будет отсутствовать файл базы данных SQLite, прокси создаст его автоматически.

В зависимости от размера базы данных обновление базы данных до версии 4.0 может занять продолжительное время.

Перед выполнением обновления убедитесь что прочитали все соответствующие **заметки по обновлению!**

Доступны следующие заметки по обновлению:

Обновление с	Прочтайте заметки по обновлению полностью	Важные заметки/изменения между версиями
3.4.x	Для 4.0	<p>Библиотеки 'libpthread' и 'zlib' теперь обязательны; Поддержка протокола в виде простого текста убрана и заголовок обязательен; Zabbix агенты версий Pre-1.4 более не поддерживаются; Параметр Server в конфигурации пассивного прокси теперь обязательен</p> <p>Поддержка SQLite в виде основной базы данных убрана для Zabbix сервера/веб-интерфейса; Поддерживается Perl совместимые регулярные выражения (PCRE) вместо POSIX расширенных; Библиотеки 'libpcre' and 'libevent' обязательны для Zabbix сервера;</p> <p>Добавлены проверки кода выхода для пользовательских параметров, удаленных команд и элементов данных system.run[] без 'nowait' флага, а также для выполняемых скриптов Zabbix сервером; Zabbix Java gateway необходимо обновить для поддержки новых функций</p>
3.2.x	Также для 3.4	Обновление базы данных может быть медленным, в
3.0.x	Также для 3.2	

зависимости от размеров таблиц истории

Вы возможно захотите также проверить [требования](#) для 4.0.

Возможно удобно запустить две параллельные SSH сессии на время обновления, выполняя шаги обновления в одной сессии и наблюдая за файлами журналов сервера/прокси в другой. Например, при выполнении `tail -f zabbix_server.log` или `tail -f zabbix_proxy.log` во второй SSH сессии будут отображаться последние записи из файла журнала и возможные ошибки в режиме реального времени. Такой подход может быть критичным на продуктивных серверах.

Процесс обновления

1 Остановите Zabbix процессы

Остановите Zabbix сервер, чтобы быть уверенными, что в базу данных не будет происходить запись новых данных.

```
# systemctl stop zabbix-server
```

При обновлении прокси, остановите также и его.

```
# systemctl stop zabbix-proxy
```

Известно, что имеется возможность запустить обновленный сервер с ещё не обновленными прокси, которые будут отправлять данные на новый сервер (хотя прокси и не смогут обновлять свою конфигурацию). Однако, такой подход не рекомендуется и не поддерживается Zabbix, его выбор исключительно на ваш страх и риск.

2 Сделайте архивную копию существующей базы данных Zabbix

Этот шаг очень важен. Убедитесь, что у вас есть архивная копия вашей базы данных. Это поможет, если процедура обновления закончится неудачно (отсутствие свободного места на диске, выключение питания, любая неожиданная 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.

Файлы конфигурации:

```
# mkdir /opt/zabbix-backup/
# cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
# cp /etc/httpd/conf.d/zabbix.conf /opt/zabbix-backup/
```

Файлы PHP и бинарные файлы Zabbix:

```
# cp -R /usr/share/zabbix/ /opt/zabbix-backup/
# cp -R /usr/share/doc/zabbix-* /opt/zabbix-backup/
```

4 Обновите пакет конфигурации репозитория

Чтобы продолжить обновление, необходимо обновить пакет текущий репозитория.

```
# rpm -Uvh https://repo.zabbix.com/zabbix/4.0/rhel/7/x86_64/zabbix-release-4.0-1.el7.centos.noarch.rpm
```

5 Обновите компоненты Zabbix

Для обновления компонентов Zabbix вы можете выполнить что-то вроде:

```
# yum upgrade zabbix-server-mysql zabbix-web-mysql zabbix-agent
```

Если используете PostgreSQL, в команде замените `mysql` на `pgsql`. Если обновляете прокси, в команде замените `server` на `proxy`.

6 Просмотрите параметры конфигурации компонент

Для получения более подробных сведений смотрите заметки по обновлению на предмет [обязательных изменений](#).

7 Запустите процессы Zabbix

Запустите обновленные компоненты Zabbix.

```
# systemctl start zabbix-server  
# systemctl start zabbix-proxy  
# systemctl start zabbix-agent
```

8 Очистите cookies и кэш в веб-браузере

После обновления вам, возможно, потребуется очистить cookies веб-браузера и кэш веб-браузера, чтобы Zabbix веб-интерфейс работал правильно.

Обновление между минорными версиями

Имеется возможность обновления между минорными версиями 4.0.x (например, с 4.0.1 на 4.0.3). Процедура минорного обновления Zabbix очень проста.

Для выполнения минорного обновления Zabbix необходимо выполнить:

```
$ sudo yum upgrade 'zabbix-*'
```

Для выполнения обновления минорной версии Zabbix сервера выполните:

```
$ sudo yum upgrade 'zabbix-server-*'
```

Для выполнения обновления минорной версии Zabbix агента выполните:

```
$ sudo yum upgrade 'zabbix-agent-*'
```

Обратите внимание, что в этих командах вы можете также использовать 'update' вместо 'upgrade'. В то время как 'upgrade' удалит устаревшие пакеты, 'update' сохранит их.

2 Debian/Ubuntu

Overview

Этот раздел описывает требуемые шаги для успешного [обновления](#) с Zabbix 3.4.x до Zabbix 4.0.x с использованием официальных пакетов Zabbix для Debian/Ubuntu.

В то время как обновление Zabbix агентов не является обязательным шагом (но рекомендуемым), Zabbix сервер и прокси должны быть [одной мажорной версии](#). Поэтому, при наличии сервер-прокси инсталляции, Zabbix сервер и все прокси должны быть остановлены и обновлены.

Чтобы минимизировать времяостоя и потери данных в процессе обновления, рекомендуется остановить и обновить Zabbix сервер и один за другим затем остановить, обновить и запустить Zabbix прокси. Когда все прокси будут обновлены, запустите Zabbix сервер. Пока Zabbix сервер будет остановлен, запущенные прокси будут продолжать сбор и хранение данных и отправят эти данные на Zabbix сервер, когда сервер запустится и заработает. Любые оповещения о проблемах в процессеостоя Zabbix сервера будут сгенерированы только после того как обновленный сервер запустится.

Обратите внимание, что при наличии прокси с SQLite базой данных, данные истории этих прокси будут потеряны до обновления, так как обновление файла базы данных SQLite не поддерживается и этот файл необходимо удалить вручную. Когда прокси запускается в первый раз и будет отсутствовать файл базы данных SQLite, прокси создаст его автоматически.

В зависимости от размера базы данных обновление базы данных до версии 4.0 может занять продолжительное время.

Перед выполнением обновления убедитесь что прочитали все соответствующие **заметки по обновлению!**

Доступны следующие заметки по обновлению:

Обновление с	Прочтайте заметки по обновлению полностью	Важные заметки/изменения между версиями
3.4.x	Для 4.0	Библиотеки 'libpthread' и 'zlib' теперь обязательны; Поддержка протокола в виде простого текста убрана и заголовок обязательен; Zabbix агенты версий Pre-1.4 более не поддерживаются; Параметр Server в конфигурации пассивного прокси теперь обязательен
3.2.x	Также для 3.4	Поддержка SQLite в виде основной базы данных убрана для Zabbix сервера/веб-интерфейса; Поддерживаются Perl совместимые регулярные выражения (PCRE) вместо POSIX расширенных; Библиотеки 'libpcre' and 'libevent' обязательны для Zabbix сервера; Добавлены проверки кода выхода для пользовательских параметров, удаленных команд и элементов данных system.run[] без 'nowait' флага, а также для выполняемых скриптов Zabbix сервером; Zabbix Java gateway необходимо обновить для поддержки новых функций
3.0.x	Также для 3.2	Обновление базы данных может быть медленным, в зависимости от размеров таблиц истории

Вы возможно захотите также проверить [требования](#) для 4.0.

Возможно удобно запустить две параллельные SSH сессии на время обновления, выполняя шаги обновления в одной сессии и наблюдая за файлами журналов сервера/прокси в другой. Например, при выполнении `tail -f zabbix_server.log` или `tail -f zabbix_proxy.log` во второй SSH сессии будут отображаться последние записи из файла журнала и возможные ошибки в режиме реального времени. Такой подход может быть критичным на продуктивных серверах.

Процесс обновления

1 Остановите Zabbix процессы

Остановите Zabbix сервер, чтобы быть уверенными, что в базу данных не будет происходить запись новых данных.

```
# service zabbix-server stop
```

При обновлении прокси, остановите также и его.

```
# service zabbix-proxy stop
```

Известно, что имеется возможность запустить обновленный сервер с ещё не обновленными прокси, которые будут отправлять данные на новый сервер (хотя прокси и не смогут обновлять свою конфигурацию). Однако, такой подход не рекомендуется и не поддерживается Zabbix, его выбор исключительно на ваш страх и риск.

2 Сделайте архивную копию существующей базы данных Zabbix

Этот шаг очень важен. Убедитесь, что у вас есть архивная копия вашей базы данных. Это поможет, если процедура обновления закончится неудачно (отсутствие свободного места на диске, выключение питания, любая неожиданная 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.

Файлы конфигурации:

```
# mkdir /opt/zabbix-backup/
# cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
# cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-backup/
```

Файлы PHP и бинарные файлы Zabbix:

```
# cp -R /usr/share/zabbix/ /opt/zabbix-backup/
# cp -R /usr/share/doc/zabbix-* /opt/zabbix-backup/
```

4 Обновите пакет конфигурации репозитория

Чтобы продолжить обновление, необходимо удалить пакет текущий репозитория.

```
# rm -Rf /etc/apt/sources.list.d/zabbix.list
```

Затем установить пакет нового репозитория.

Для **Debian 9** выполните:

```
# wget https://repo.zabbix.com/zabbix/3.5/debian/pool/main/z/zabbix-release/zabbix-
release_3.5-1+stretch_all.deb
# dpkg -i zabbix-release_3.5-1+stretch_all.deb
```

Для **Debian 8** выполните:

```
# wget https://repo.zabbix.com/zabbix/3.5/debian/pool/main/z/zabbix-release/zabbix-
release_3.5-1+jessie_all.deb
# dpkg -i zabbix-release_3.5-1+jessie_all.deb
```

Для **Debian 7** выполните:

```
# wget https://repo.zabbix.com/zabbix/3.5/debian/pool/main/z/zabbix-release/zabbix-
release_3.5-1+wheezy_all.deb
# dpkg -i zabbix-release_3.5-1+wheezy_all.deb
```

Для **Ubuntu 18.04** выполните:

```
# wget https://repo.zabbix.com/zabbix/3.5/ubuntu/pool/main/z/zabbix-release/zabbix-
release_3.5-1+bionic_all.deb
# dpkg -i zabbix-release_3.5-1+bionic_all.deb
```

Для **Ubuntu 16.04** выполните:

```
# wget https://repo.zabbix.com/zabbix/3.5/ubuntu/pool/main/z/zabbix-release/zabbix-
release_3.5-1+xenial_all.deb
# dpkg -i zabbix-release_3.5-1+xenial_all.deb
```

Для Ubuntu 14.04 выполните:

```
# wget https://repo.zabbix.com/zabbix/3.5/ubuntu/pool/main/z/zabbix-release/zabbix-release_3.5-1+trusty_all.deb  
# dpkg -i zabbix-release_3.5-1+trusty_all.deb
```

Обновите информацию о репозитории.

```
# apt-get update
```

5 Обновите компоненты Zabbix

Для обновления компонентов Zabbix вы можете выполнить что-то вроде:

```
# apt-get install --only-upgrade zabbix-server-mysql zabbix-frontend-php zabbix-agent
```

Если используете PostgreSQL, в команде замените `mysql` на `pgsql`. Если обновляете прокси, в команде замените `server` на `proxy`.

6 Просмотрите параметры конфигурации компонент

Для получения более подробных сведений смотрите заметки по обновлению на предмет [обязательных изменений](#).

Для ознакомления с новыми необязательными параметрами, пожалуйста, смотрите раздел [Что нового](#).

7 Запустите процессы Zabbix

Запустите обновленные компоненты Zabbix.

```
# service zabbix-server start  
# service zabbix-proxy start  
# service zabbix-agent start
```

8 Очистите cookies и кэш в веб-браузере

После обновления вам, возможно, потребуется очистить cookies веб-браузера и кэш веб-браузера, чтобы Zabbix веб-интерфейс работал правильно.

Обновление между минорными версиями

Имеется возможность обновления минорные версии 4.0.x (например, с 4.0.1 на 4.0.3). Процедура очень проста.

Для минорного обновления Zabbix, пожалуйста, выполните:

```
$ sudo apt install --only-upgrade 'zabbix.*'
```

Для обновления минорной версии Zabbix сервера, пожалуйста, выполните:

```
$ sudo apt install --only-upgrade 'zabbix-server.*'
```

Для обновления минорной версии Zabbix агента, пожалуйста, выполните:

```
$ sudo apt install --only-upgrade 'zabbix-agent.*'
```

Обновление из исходных кодов

Обзор

Этот раздел описывает требуемые шаги для успешного [обновления](#) с Zabbix 3.4.x до Zabbix 4.0.x с использованием официальных исходных кодов Zabbix.

В то время как обновление Zabbix агентов не является обязательным шагом (но рекомендуемым), Zabbix сервер и прокси должны быть [одной мажорной версии](#). Поэтому, при наличии сервер-прокси инсталляции, Zabbix сервер и все прокси должны быть остановлены и обновлены.

Чтобы минимизировать время простоя и потери данных в процессе обновления, рекомендуется остановить и обновить Zabbix сервер и один за другим затем остановить, обновить и запустить Zabbix прокси. Когда все прокси будут обновлены, запустите Zabbix сервер. Пока Zabbix сервер будет остановлен, запущенные прокси будут продолжать сбор и хранение данных и отправят эти данные на Zabbix сервер, когда сервер запустится и заработает. Любые оповещения о проблемах в процессе простоя Zabbix сервера будут сгенерированы только после того как обновленный сервер запустится.

Известно, что имеется возможность запустить обновленный сервер с ещё не обновленными прокси, которые будут отправлять данные на новый сервер (хотя прокси и не смогут обновлять свою конфигурацию). Однако, такой подход не рекомендуется и не поддерживается Zabbix, его выбор исключительно на ваш страх и риск.

Обратите внимание, что при наличии прокси с SQLite базой данных, данные истории этих прокси будут потеряны до обновления, так как обновление файла базы данных SQLite не поддерживается и этот файл необходимо удалить вручную. Когда прокси запускается в первый раз и будет отсутствовать файл базы данных SQLite, прокси создаст его автоматически.

В зависимости от размера базы данных обновление базы данных до версии 4.0 может занять продолжительное время.

Перед выполнением обновления убедитесь что прочитали все соответствующие **заметки по обновлению!**

Доступны следующие заметки по обновлению:

Обновление с	Прочтайте заметки по обновлению полностью	Важные заметки/изменения между версиями
3.4.x	Для 4.0	Библиотеки 'libpthread' и 'zlib' теперь обязательны; Поддержка протокола в виде простого текста убрана и заголовок обязательен; Zabbix агенты версий Pre-1.4 более не поддерживаются; Параметр Server в конфигурации пассивного прокси теперь обязательен
3.2.x	Также для 3.4	Поддержка SQLite в виде основной базы данных убрана для Zabbix сервера/веб-интерфейса; Поддерживается Perl совместимые регулярные выражения (PCRE) вместо POSIX расширенных; Библиотеки 'libpcre' and 'libevent' обязательны для Zabbix сервера; Добавлены проверки кода выхода для пользовательских параметров, удаленных команд и элементов данных system.run[] без 'nowait' флага, а также для выполняемых скриптов Zabbix сервером; Zabbix Java gateway необходимо обновить для поддержки новых функций
3.0.x	Также для 3.2	Обновление базы данных может быть медленным, в зависимости от размеров таблиц истории
2.4.x	Также для 3.0	Минимально требуемая версия PHP повышена с 5.3.0 до 5.4.0 Необходимо указывать параметрLogFile агента
2.2.x	Также для 2.4	Удалён распределенный мониторинг на основе нод

Обновление с	Прочтите заметки по обновлению полностью	Важные заметки/изменения между версиями
2.0.x	Также для 2.2	<p>Минимально требуемая версия PHP повышенна с 5.1.6 до 5.3.0; Для правильной работы сервера требуется регистрозависимая база данных MySQL; Расширение 'mysqli' PHP требуется вместо 'mysql'</p> <p>Вы возможно захотите также проверить требования для 4.0.</p>

Возможно удобно запустить две параллельные SSH сессии на время обновления, выполняя шаги обновления в одной сессии и наблюдая за файлами журналов сервера/прокси в другой. Например, при выполнении `tail -f zabbix_server.log` или `tail -f zabbix_proxy.log` во второй SSH сессии будут отображаться последние записи из файла журнала и возможные ошибки в режиме реального времени. Такой подход может быть критичным на продуктивных серверах.

Процесс обновления сервера

1 Остановите Zabbix сервер

Остановите Zabbix сервер, чтобы быть уверенными, что в базу данных не будет происходить запись новых данных.

2 Сделайте архивную копию существующей базы данных Zabbix

Этот шаг очень важен. Убедитесь, что у вас есть архивная копия вашей базы данных. Это поможет, если процедура обновления закончится неудачно (отсутствие свободного места на диске, выключение питания, любая неожиданная проблема).

3 Сделайте копию файлов конфигурации, PHP файлов и приложений Zabbix

Сделайте резервные копии приложений Zabbix, файлов конфигурации и папки с файлами PHP.

4 Установите новые исполняемые файлы сервера

Используйте эти [инструкции](#) для компиляции Zabbix сервера из исходных кодов.

5 Проверьте параметры конфигурации сервера

Смотрите заметки по обновлению для получения детальной информации об [обязательных изменениях](#).

Для ознакомления с новыми необязательными параметрами, пожалуйста,смотрите раздел [Что нового](#).

6 Запустите новые бинарные файлы Zabbix

Запустите сервер. Проверьте файлы журналов, чтобы удостовериться, что сервер запустился успешно.

Процесс Zabbix сервера после запуска автоматически обновит схему базы данных. При запуске Zabbix сервер сообщает текущую (обязательную и опциональную) и требуемую версии базы данных. Если текущая обязательная версия старше чем требуемая версия, 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”. Если какой-либо из патчей обновления будет ошибочным, 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 начала запуска процесса 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 Установите новый веб-интерфейс Zabbix

Минимально требуемая версия PHP 5.4.0. Обновите, если требуется, и следуйте [инструкции по установке](#).

8 Очистите cookies и кэш в веб-браузере

После обновления вам, возможно, потребуется очистить cookies веб-браузера и кэш веб-браузера, чтобы Zabbix веб-интерфейс работал правильно.

Процесс обновления прокси

1 Остановите Zabbix прокси

Остановите Zabbix прокси.

2 Сделайте копию файлов конфигурации и бинарных файлов Zabbix прокси

Сделайте резервные копии бинарных файла Zabbix прокси и файлов конфигурации.

3 Установите новые исполняемые файлы прокси

Используйте эти [инструкции](#) для компиляции Zabbix прокси из исходных кодов.

4 Проверьте параметры конфигурации прокси

В этой версии обязательные изменения в [параметрах](#) прокси отсутствуют. Для ознакомления с новыми необязательными параметрами, пожалуйста,смотрите раздел [Что нового](#).

5 Запустите новый Zabbix прокси

Запустите новый Zabbix прокси. Проверьте файлы журналов, чтобы увидеть, что прокси запустился успешно.

Процесс Zabbix прокси после запуска автоматически обновит схему базы данных. Обновление базы данных происходит аналогично тому, как при запуске [Zabbix сервера](#).

Процесс обновления агента

Обновление агентов не является обязательным процессом. Вы можете обновить агенты, если это требуется для доступа к новому функционалу.

1 Остановите Zabbix агент

Остановите Zabbix агент.

2 Сделайте копию файлов конфигурации и бинарных файлов Zabbix агента

Сделайте резервные копии бинарного файла Zabbix агента и файла конфигурации.

3 Установите новые исполняемые файлы агента

Используйте эти [инструкции](#) для компиляции Zabbix агента из исходных кодов.

Дополнительно, вы можете загрузить уже скомпилированные Zabbix агенты со [страницы загрузки Zabbix](#).

4. Установка

4 Проверьте параметры конфигурации агента

В этой версии обязательные изменения в [параметрах](#) агента отсутствуют.

5 Запустите новый Zabbix агент

Запустите новый Zabbix агент. Проверьте файлы журналов, чтобы увидеть, что агент запустился успешно.

Обновление между минорными версиями

При обновлении между минорными версиями 4.0.x (например, с 4.0.1 на 4.0.3) необходимо выполнить те же действия для сервера/прокси/агента, как и при обновлении между мажорными версиями. Единственное различие заключается в том, что при обновлении между минорными версиями никаких изменений в базе данных не производится.

4.7 Известные проблемы

Глобальная корреляция событий

События могут быть некорректно скоррелированы, если промежуток времени между первым и вторым событиями очень мал, к примеру пол секунды или меньше.

IPMI проверки

IPMI проверки не будут работать со стандартным пакетом библиотеки OpenIPMI на Debian до 9 (stretch) и Ubuntu до 16.04 (xenial). Чтобы исправить проблему, пересоберите OpenIPMI библиотеку с включенным OpenSSL, как обсуждалось в [ZBX-6139](#) [en].

SSH проверки

Некоторые Linux дистрибутивы такие как Debian, Ubuntu не поддерживают шифрованные приватные ключи (с ключевой фразой) при установке библиотеки libssh2 из пакетов.

Для получения более подробных сведенийсмотрите [ZBX-4850](#).

ODBC проверки

Zabbix сервер или прокси, которые используют MySQL в качестве своей базы данных могут или не могут корректно работать с библиотекой MySQL ODBC в связи с [общей проблемой](#). Пожалуйста, просмотрите [ZBX-7665](#) для получения более подробной информации и возможных способах решения проблемы.

XML данные запрашиваемые с Microsoft SQL сервера могут быть обрезаны до 2033 по причине [проблемы](#) в Microsoft.

HTTPS проверки

Веб-сценарии и HTTP агент использующие https протокол, Zabbix агент проверки `net.tcp.service[https...]` и `net.tcp.service.perf[https...]` могут завершиться неудачей, если целевой сервер настроен на запрет протокола TLS v1.0 или ниже. Пожалуйста, просмотрите [ZBX-9879](#) для получения более подробной информации и возможных способах решения проблемы.

Веб-мониторинг и HTTP агент

Zabbix сервер имеет утечку памяти на CentOS 6, CentOS 7 и, возможно, на других похожих дистрибутивах Linux по причине [ошибки в библиотеке](#) [en] при включенной опции "Проверка SSL узла" в веб-сценариях или HTTP агенте. Пожалуйста, просмотрите [ZBX-10486](#) [en] для получения более подробной информации и о доступных методах обхода.

Простые проверки

Имеется проблема в **fping** утилите в версиях до v3.10 выпуска 2.1.2, которая приводит в дубликатам ответных echo пакетов. Такое поведение может вызвать неожиданные результаты в **icmpping**, **icmppingloss**, **icmppingsec** элементах данных. Рекомендуется использовать последнюю версию **fping**. Пожалуйста, смотрите [ZBX-11726 \[en\]](#) для получения более детальных сведений.

SNMP проверки

Если используется OpenBSD операционная система, проблема использования памяти после освобождения памяти в Net-SNMP библиотеке вплоть до 5.7.3 версии может привести к остановке Zabbix сервера, если SourceIP параметр указан в файле конфигурации Zabbix сервера. Как вариант решения, пожалуйста, не задавайте SourceIP параметр. Эта проблема также применима и к Linux, но она не приводит к остановке работы Zabbix сервера. К пакету net-snmp применен локальный патч на OpenBSD и будет выпущен с OpenBSD 6.3.

Проблема совместимости с PHP 7.0

Было замечено, что с PHP 7.0 импорт шаблона с триггерами от веб-мониторинга может провалиться по причине некорректно добавленных двойных кавычек к элементам данных веб-мониторинга в выражениях триггеров. Проблема исчезает с обновлением PHP до 7.1.

Графики

Результатом изменения на Летнее время (DST) является нарушение отображения подписей к оси X (дублирование данных, пропущенная дата и так далее).

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 (смотрите [ZBX-10884](#) для получения более подробной информации).

Медленные запросы MySQL

Zabbix сервер генерирует медленные SELECT запросы в случае несуществующих значений у элементов данных. Это обусловлено известной проблемой в MySQL версий 5.6/5.7. Решением такой проблемы является отключение параметра `index_condition_pushdown` оптимизатора в MySQL. Расширенное обсуждение смотрите здесь [ZBX-10652](#).

API

Параметр `output` работает некорректно с методом `history.get`.

API login

Может быть создано большое количество открытых сессий пользователю при использовании пользовательских скриптов с методом `user.login` без последующего `user.logout`.

Проблема с IPv6 адресами в SNMPv3 трапах

По причине проблемы в net-snmp, IPv6 адреса могут некорректно отображаться при использовании SNMPv3 в SNMP трапах. Для получения более детальных сведений и возможных путях обхода смотрите [ZBX-14541 \[en\]](#).

4.8 Изменения в шаблонах

На этой странице перечислены все изменения шаблонов, которые поставляются с Zabbix. Предлагается изменить эти шаблоны на существующих инсталляциях - в зависимости от изменений, это можно сделать либо импортированием последней версии, либо выполнением изменений вручную.

5. Быстрый старт

5.1 Вход и настройка пользователя

Обзор

В этом разделе вы узнаете как войти в Zabbix и настроить пользователя системы.

Вход

The screenshot shows the Zabbix login interface. It features a red 'ZABBIX' logo at the top. Below it are two input fields for 'Username' and 'Password'. A 'Remember me for 30 days' checkbox is checked. A large blue 'Sign in' button is centered below the inputs. At the bottom, there is a link to 'sign in as guest'.

Это экран “Приветствия” в Zabbix. Введите имя пользователя **Admin** с паролем **zabbix** для входа под [Супер-Администратором Zabbix](#).

Когда вы войдете, вы увидите 'Подключен как Admin' в нижнем правом углу страницы. Доступ к меню *Настройка* и *Администрирование* будет предоставлен.

Защита от атак перебором

В случае пяти последовательных неудачных попыток входа, Zabbix интерфейс будет принудительно делать паузу в 30 секунд для предотвращения атак методом перебора пароля и атак по словарю.

IP адрес, с которого были осуществлены неудачные попытки, будет отображен после успешного входа в систему.

Добавление пользователя

Для просмотра информации о пользователях перейдите в *Администрирование* → *Пользователи*.

Users							User group	All	Create user	
	ALIAS	NAME	SURNAME	USER TYPE	GROUPS	IS ONLINE?	LOGIN	FRONTEND ACCESS	DEBUG MODE	STATUS
<input type="checkbox"/>	Admin	Zabbix	Administrator	Zabbix Super Admin	Zabbix administrators	Yes (2015-08-05 17:25:44)	Ok	System default	Disabled	Enabled
<input type="checkbox"/>	guest			Zabbix User	Guests	Yes (2015-08-05 17:16:38)	Ok	System default	Disabled	Enabled

Изначально в Zabbix только два предустановленных пользователя.

- Пользователь 'Admin' это суперпользователь Zabbix, который имеет все привилегии.
- Пользователь 'Guest' это специальный пользователь по умолчанию. Если пользователь не вошел в систему, тогда он получит доступ с привилегиями пользователя "guest". По умолчанию, "guest" не имеет разрешений на объекты Zabbix.

Для добавления нового пользователя нажмите *Создать пользователя*.

В диалоге создания пользователя убедитесь, что пользователь принадлежит хотя бы одной [группе пользователей](#), например 'Zabbix администраторы'.

Users

User Media Permissions

* Alias

Name

Surname

* Groups

* Password

* Password (once again)

Language

Theme

Auto-login

Auto-logout (min 90 seconds)

* Refresh (in seconds)

* Rows per page

URL (after login)

По умолчанию у новых пользователей нет добавленных способов оповещения (методов отправки уведомлений). Для их создания перейдите на закладку 'Способы оповещения' и нажмите *Добавить*.

Media

Type

* Send to

* When active

Use if severity Not classified
 Information
 Warning
 Average
 High
 Disaster

Enabled

В этом всплывающем окне введите для пользователя его e-mail адрес.

Вы можете указать период времени, когда этот способ будет активен (смотрите страницу [Спецификаций периодов времени](#) для получения сведений о формате), по умолчанию способ активен всегда. Вы также можете настроить [важность триггера](#), для которого способ оповещения будет использоваться, но сейчас оставьте все включенным.

Нажмите *Добавить*, затем *Сохранить* в свойствах пользователя. Новый пользователь появится в списке пользователей.

Users									User group	All	<input type="button" value="Create user"/>
<input type="checkbox"/>	ALIAS	NAME	SURNAME	USER TYPE	GROUPS	IS ONLINE?	LOGIN	FRONTEND ACCESS	DEBUG MODE	STATUS	
<input type="checkbox"/>	Admin	Zabbix	Administrator	Zabbix Super Admin	Zabbix administrators	Yes (2015-11-05 07:26:26)	Ok	System default	Disabled	Enabled	
<input type="checkbox"/>	guest			Zabbix User	Guests	Yes (2015-11-05 07:25:22)	Ok	System default	Disabled	Enabled	
<input type="checkbox"/>	user	New	User	Zabbix User	Zabbix administrators	No	Ok	System default	Disabled	Enabled	

Добавление прав доступа

По умолчанию, новый пользователь не имеет прав. Для предоставления ему прав нажмите на имя группы в колонке *Группы* (в данном случае - 'Zabbix administrators'). В этом диалоге свойств группы перейдите на закладку *Права*.

User groups

The screenshot shows a 'User group' tab selected in the top navigation bar. A modal dialog is open, titled 'Permissions'. It contains a table with one row: 'Host group *' under 'Permissions' and 'None' under 'Permissions'. Below the table is a search bar with placeholder text 'type here to search' and a 'Select' button highlighted with a green arrow. At the bottom of the dialog are buttons for 'Read-write', 'Read', 'Deny', and 'None'.

Этот пользователь должен иметь доступ только на чтение группы *Linux servers*, поэтому нажмите *Выбрать* сразу после поля выбора имени группы пользователей.

The screenshot shows a 'Host groups' tab selected in the top navigation bar. A modal dialog is open, listing host groups: 'Name', 'Discovered hosts', 'Hypervisors', 'Linux servers' (which has a checked checkbox), 'Templates', 'Templates/Applications', 'Virtual machines', and 'Zabbix servers'. The 'Linux servers' item is highlighted with a yellow background. At the bottom right of the dialog is a 'Select' button.

В этом всплывающем окне отметьте 'Linux servers' и затем нажмите *Выбрать*. *Linux servers* должны отображаться в поле выбора. Нажмите на кнопку 'Чтение', чтобы задать уровень прав доступа и затем *Добавить* для добавления группы в список прав доступа. В диалоге свойств группы пользователей, нажмите *Обновить*.

В Zabbix, права доступа к узлам сети назначаются через [группы пользователей](#), а не индивидуально пользователям.

Готово! Теперь вы можете попробовать войти под созданным пользователем.

5.2 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 *Настройка → Узлы сети*. Уже существует один предустановленный узел сети, называемый 'Zabbix server', но мы хотим узнать как добавить еще один узел сети.

Для добавления нового узла сети нажмите *Создать*. Это действие покажет нам диалог настройки узла сети.

Hosts

Host Templates IPMI Macros Host inventory Encryption

* Host name

Visible name

* Groups

* At least one interface must exist.

Agent interfaces

	IP address	DNS name	Connect to
...	127.0.0.1		<input type="button" value="IP"/> <input type="button" value="DNS"/>

SNMP interfaces

JMX interfaces

IPMI interfaces

Description

Monitored by proxy

Enabled

Все обязательные поля ввода отмечены красной звёздочкой.

Как минимум введите:

Имя узла сети

- Введите имя узла сети. Допускаются число-буквенные символы, пробелы, точки, тире и подчёркивания.

Группы

- Выберите одну или несколько существующих групп, нажав на кнопку *Выбор* или введите имя несуществующей группы, чтобы создать новую.



Все права доступа назначаются на группы узлов сети, не индивидуально узлам сети. Поэтому узел сети должен принадлежать хотя бы одной группе.

IP адрес

- Введите IP адрес узла сети. Обратите внимание, Zabbix агенту нужно указать IP адрес Zabbix сервера в файле конфигурации в параметре 'Server'.

Остальные опции подойдут нам на данный момент по умолчанию.

Когда закончите, нажмите на *Добавить*. Ваш новый узел сети должен быть виден в списке узлов сети.



Если иконка *ZBX* в колонке *Доступность* красная, то это значит что имеется какая то ошибка со связью - поместите курсор мышки над этой иконкой и вы увидите всплывающую подсказку о ошибке. Если иконка серая, значит состояние пока не обновилось. Проверьте запущен ли Zabbix сервер и попробуйте обновить страницу немного позже.

5.3 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 какие данные собираются с узла сети.

Добавление элемента данных

Все элементы данных группируются вокруг узлов сети. Именно поэтому для настройки примера элемента данных мы перейдем в *Настройки* → *Узлы сети* и найдем 'Новый узел сети', который мы создали.

Ссылка *Элементы данных* в строке с 'Новым узлом сети' должна отображать количество равное '0'. Нажмите на эту ссылку, и затем нажмите на *Создать элемент данных*. Это покажет нам диалог создания элемента данных.

Item Preprocessing

* Name	CPU Load																
Type	Zabbix agent																
* Key	system.cpu.load																
* Host interface	127.0.0.1 : 10050																
Type of information	Numeric (float)																
Units																	
* Update interval	30s																
Custom intervals	<table border="1"> <thead> <tr> <th>Type</th> <th>Interval</th> <th>Period</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>Flexible</td> <td>Scheduling</td> <td>50s</td> <td>1-7.00:00-24:00</td> </tr> <tr> <td colspan="4">Remove</td> </tr> <tr> <td colspan="4">Add</td> </tr> </tbody> </table>	Type	Interval	Period	Action	Flexible	Scheduling	50s	1-7.00:00-24:00	Remove				Add			
Type	Interval	Period	Action														
Flexible	Scheduling	50s	1-7.00:00-24:00														
Remove																	
Add																	
* History storage period	90d																
* Trend storage period	365d																
Show value	As is show value mappings																
New application	<input type="text"/>																
Applications	-None-																
Populates host inventory field	-None-																
Description	<input type="text"/>																
Enabled	<input checked="" type="checkbox"/>																
Add Cancel																	

Все обязательные поля ввода отмечены красной звёздочкой.

Введите необходимую информацию, для нашего примера элемента данных:

Имя

- Введите значение *Загрузка CPU*. Это будет видимым именем элемента данных в списках и в других местах.

Ключ

- Вручную введите значение *system.cpu.load*. Это формальное имя элемента данных, которое идентифицирует тип информации, которая будет собираться. Этот ключ является лишь одним из [предопределенных ключей](#), которые поставляются с Zabbix агентом.

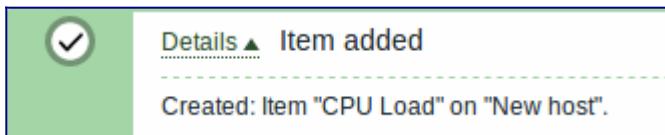
Тип информации

- Здесь выберите *Числовой (с плавающей точкой)*. Этот атрибут определяет формат ожидаемых данных.

Вы также можете захотеть уменьшить количество дней хранения [истории элемента данных](#), до 7 или 14. Это хорошая практика для уменьшения базы данных при хранении большого количества значений истории.

[Остальные опции](#) подойдут нам на данный момент по умолчанию.

Когда закончите, нажмите на *Добавить*. Новый элемент данных должен появится в списке элементов данных. Нажмите на *Детали* выше списка, чтобы просмотреть, что именно было сделано.



Просмотр данных

С элементом данных всё понятно, вам может быть интересно есть ли на самом деле сбор данных. Для этого перейдите в *Мониторинг* → *Последние данные*, выберите 'Новый узел сети' в фильтре и нажмите на *Применить*.

Затем нажмите на + напротив - **другое** - и проверьте есть ли там ваш элемент данных и отображаются ли данные.

HOST	NAME	LAST CHECK	LAST VALUE	CHANGE	Graph
New host	- other - (1 item)				
	CPU Load	2015-08-08 16:00:49	2.24	-0.26	Graph

С учетом сказанного, получение первых данных может занять до 60 секунд. Это время, по умолчанию, как часто сервер читает изменения конфигурации и забирает новые элементы данных для обработки.

Если вы не видите значения в колонке 'Изменен', то возможно к настоящему времени было получено только одно значение. Подождите 30 секунд пока придет другое значение.

Если вы не видите информации о элементе данных как в снимке экрана, убедитесь что:

- вы указали поля 'Ключ' и 'Тип информации' в элементе данных так же как на снимке экрана
- агент и сервер запущены
- состояние узла сети 'Активирован' и его иконка доступности зеленая
- выберите узел сети в списке узлов сети, элемент данных активен

Графики

При наличии работающего элемента данных какое-то время, возможно настало время для того чтобы увидеть что-то визуальное. [Простые графики](#) доступны для любого наблюдаемого числового элемента данных без каких либо дополнительных настроек. Эти графики создаются в режиме реального времени.

Для просмотра графика, перейдите к *Мониторинг* → *Последние данные* и нажмите на ссылку 'График' у этого элемента данных.



5.4 Новый триггер

Обзор

В этом разделе вы узнаете как настроить триггер.

Элементы данных только собирают данные. Для автоматической оценки приходящих данных нам нужно задать триггеры. Триггер содержит выражение, которое определяет порог являющийся приемлемым уровнем для данных.

Если этот уровень будет превышать пришедшие данные, триггер будет “загораться” или перейдет в состояние 'Проблема' - давая понять, что что-то произошло и может потребовать внимания. Если уровень станет снова приемлемым, то триггер вернется в состояние 'OK'.

Добавление триггера

Для настройки триггера для нашего элемента данных перейдите в *Настройка → Узлы сети*, найдите 'Новый узел сети' и далее нажмите на *Триггеры* и затем на *Создать триггер*. Нам будет отображен диалог добавления триггера.

Trigger Dependencies

* Name	CPU load too high on "New host" for 3 minutes				
Severity	Not classified	Information	Warning	Average	High
* Expression	{New host:system.cpu.load.avg(3m)}>2				
Expression constructor					
OK event generation	Expression	Recovery expression	None		
PROBLEM event generation mode	Single	Multiple			
OK event closes	All problems	All problems if tag values match			
Tags	<input type="text" value="tag"/> <input type="text" value="value"/> Add				
Allow manual close	<input type="checkbox"/>				
URL					
Description					
Enabled	<input checked="" type="checkbox"/>				
Add Cancel					

Все обязательные поля ввода отмечены красной звёздочкой.

Введите здесь необходимую информацию для нашего триггера:

Имя

- Введите значение *Загрузка CPU слишком высокая у 'New host' в течении 3 минут.* Это будет имя триггера, которое будет отображаться в списках и в других местах.

Выражение

- Введите: {New host:system.cpu.load.avg(3m)}>2

Это выражение триггера. Убедитесь, что выражение введено верно, вплоть до последнего символа. Здесь ключ элемента данных (system.cpu.load) используется для ссылки на элемент данных. По простому данное выражение говорит, что порог проблемы превышается, когда значение средней

загрузки CPU в течении 3 минут превышает 2. Вы можете узнать больше о [синтаксисе выражений триггеров](#).

Когда завершите, нажмите на *Сохранить*. Этот новый триггер должен появиться в списке триггеров.

Просмотр состояния триггера

После добавления триггера, вы возможно заинтересуетесь узнать его состояние.

Если загрузка CPU превысит порог, который вы указали в триггере, проблема отобразится в *Мониторинг → Проблемы*.

Time	Severity	Recovery time	Status	Info	Host	Problem	Duration
09:48:39	Not classified		PROBLEM		New host	CPU load too high on New host for 3 minutes	1m 23s

Мигание указывает на недавнее изменение состояние триггера, которое имело место за последние 30 минут.

5.5 Получение оповещения о проблеме

Обзор

В этом разделе вы узнаете как настроить в Zabbix уведомления из диалога оповещений.

При наличии элементов данных, которые собирают данные, и триггеров, переходящих в состояние “Проблема” при проблемных ситуациях, было бы полезно иметь какой-нибудь механизм оповещений, который будет уведомлять нас о важных событиях, тогда когда мы не можем смотреть напрямую веб-интерфейс Zabbix.

Это то, что делают оповещения. E-mail является наиболее популярным способом для отправки уведомлений о проблемах, мы узнаем, как настроить уведомления через e-mail.

Настройки E-mail

Изначально в Zabbix имеется несколько предустановленных [способов доставки](#) оповещений. E-mail один из них.

Для конфигурирования настроек e-mail, перейдите в *Администрирование → Способы оповещений* и нажмите на *Email* в списке предустановленных способов оповещений.

Media types					Create media type
	NAME	TYPE	STATUS	USED IN ACTIONS	DETAILS
<input type="checkbox"/>	Email	Email	Enabled		SMTP server: "mail.company.com", SMTP helo: "company.com", SMTP email: "zabbix@company.com"
<input type="checkbox"/>	Jabber	Jabber	Enabled		Jabber identifier: "jabber@company.com"
<input type="checkbox"/>	SMS	SMS	Enabled		GSM modem: "ldevitlyS0"

Отобразится диалог определения настроек e-mail.

Media types

Media type Options

* Name	Email
Type	Email
* SMTP server	mail.zabbix.com
SMTP server port	25
* SMTP helo	zabbix.com
* SMTP email	zabbix@zabbix.com
Connection security	None STARTTLS SSL/TLS
Authentication	None Username and password
Enabled	<input checked="" type="checkbox"/>
<button>Update</button> <button>Clone</button> <button>Delete</button> <button>Cancel</button>	

Все обязательные поля ввода отмечены красной звёздочкой.

Установите значения сервера SMTP, SMTP helo и SMTP e-mail в зависимости от вашей среды.

'SMTP email' будет использоваться как адрес 'От' при отправке оповещений от Zabbix.

Нажмите *Обновить*, когда будет готово.

Теперь у вас есть настроенный 'Email' как рабочий способ оповещения. Способ оповещения должен быть связан с пользователями с указанными адресами доставки (как мы это делали, когда [настраивали нового пользователя](#)), в противном случае он не будет использоваться.

Новое действие

Доставка оповещений является одной из задач [действий](#), которые делаются в Zabbix. Поэтому, для настройки оповещений перейдите в *Настройка → Действия* и нажмите на *Создать действие*.

Actions

Action Operations Recovery operations Update operations

* Name

Conditions

Label	Name	Action
New condition	Trigger name	like

Add

Enabled

* At least one operation, recovery operation or update operation must exist.

Все обязательные поля ввода отмечены красной звёздочкой.

В этом диалоге введите имя действия.

В более простом случае, если мы не будем добавлять более конкретных [условий](#), действием будет обработан любой триггер изменивший состояние с 'Ок' на 'Проблема'.

Мы еще должны определить какое действие нужно делать - это настраивается на вкладке *Операции*. Нажмите во вкладке Операции на *Новое*, появится диалог новой операции.

Actions

Action Operations Recovery operations Update operations

* Default operation step duration 1h

Default subject Problem: {EVENT.NAME}

Default message Problem started at {EVENT.TIME} on {EVENT.DATE}
Problem name: {TRIGGER.NAME}
Host: {HOSTNAME}
Severity: {EVENT.SEVERITY}
Original problem ID: {EVENT.ID}
{TRIGGER.URL}

Pause operations for suppressed problems

Operations

Steps	Details	Start in	Duration	Action
1	Send message to users: user (New user) via Email	Immediately	Default	Edit Remove

Operation details

Steps 1 - 1 (0 - infinitely)

Step duration 0 (0 - use action default)

Operation type Send message

* At least one user or user group must be selected.

Send to User groups

User group	Action
Add	

Send to Users

User	Action
user (New user)	Remove
Add	

Send only to Email

Default message

Conditions

Label	Name	Action
New		

Update Cancel

* At least one operation, recovery operation or update operation must exist.

Add Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Здесь, нажмите на *Добавить* в блоке *Отправлять пользователям* и выберите пользователя ('user'), которого мы добавили. Выберите 'Email' значением в *Отправлять только через*. Когда закончите с этим, нажмите на *Добавить*.

Макросы (или переменные) {TRIGGER.STATUS} и {TRIGGER.NAME}, которые видно в полях *Тема по умолчанию* и *Сообщение по умолчанию*, будут заменены актуальными значениями состояния триггера и имени триггера.

Это всё что требуется для настройки простого действия, поэтому нажмите на *Сохранить* в диалоге действия.

Получение оповещения

Теперь, с настроенной отправкой оповещений вам на самом деле может быть интересно получить одно оповещение. Чтобы помочь с этим, мы попробовать увеличить нагрузку на наш узел сети - так, чтобы наш триггер "зажегся" и мы получили бы оповещение о проблеме.

Откройте консоль на вашем узле сети и выполните:

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

Вы можете запустить один или несколько [этих процессов](#).

Теперь перейдите в *Мониторинг* → *Последние данные* и посмотрите как увеличиваются значения 'Загрузка CPU'. Вспомните, для нашего триггера "поджигание" наступает при значении 'Загрузка CPU' больше '2' в течении 3 минут. Как только это произойдет:

- в *Мониторинг* → *Проблемы* вы должны увидеть триггер с мигающим состоянием 'Проблема'
- вы должны получить оповещение о проблеме на ваш e-mail

Если оповещения не работают:

- Проверьте еще раз, что настройки e-mail и действия были сконфигурированы верно
- Убедитесь, что созданный вами пользователь имеет по крайней мере права на чтение узла сети который генерирует событие, как отмечается в шаге [Добавление пользователя](#). Пользователь, входящий в группу 'Network administrators' должен иметь, по крайней мере, доступ на чтение группы узлов сети 'Linux servers', к которой принадлежит наш узел сети.
- Кроме того, вы можете проверить журнал действий перейдя в *Отчеты* → *Журнал действий*.

5.6 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, они могут выглядеть как множество шагов, если потребуется, скажем, добавление их к тысячи 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 соединен с узлом сети, он унаследует все сущности шаблона. Таким образом, обычно предопределенная куча проверок может быть применена очень быстро.

Добавление шаблона

Для начала работы с шаблонами мы первым делом должны его создать. Чтобы это сделать, перейдите в *Настройка* → *Шаблоны* и нажмите на *Создать шаблон*. Так мы увидим диалог настройки шаблона.

The screenshot shows a 'Templates' configuration dialog. At the top, there are three tabs: 'Template' (selected), 'Linked templates', and 'Macros'. Below the tabs, there are several input fields:

- 'Template name': A required field marked with a red asterisk (*). It contains the value 'New template'.
- 'Visible name': An optional field containing an empty string.
- 'Groups': A dropdown menu currently set to 'Templates' (with a delete icon) and a 'Select' button.
- 'Description': A large text area for notes, which is currently empty.

At the bottom of the dialog are two buttons: 'Add' (in blue) and 'Cancel' (in grey).

Все обязательные поля ввода отмечены красной звёздочкой.

Введите здесь необходимые параметры:

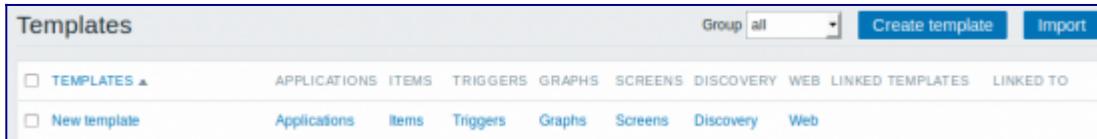
Имя шаблона

- Введите имя шаблона. Допускаются число-буквенные символы, пробелы и подчёркивания.

Группы

- Выберите одну или несколько групп, нажав на кнопку *Выбор*. Шаблон должен принадлежать группе.

Когда завершите, нажмите на *Добавить*. Ваш новый шаблон должен появиться в списке шаблонов.



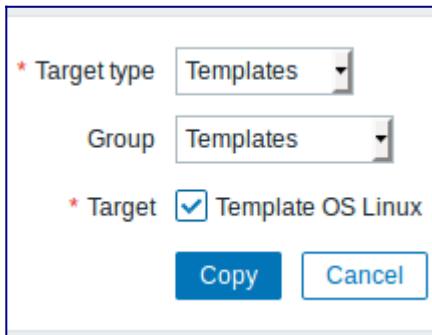
Как вы можете видеть, шаблон есть, но он ничего не содержит в себе - нет элементов данных, триггеров и других сущностей.

Добавление элемента данных в шаблон

Для добавления элемента данных в шаблон, перейдите в список элементов данных 'Новый узел сети' узла сети. В *Настройки* → Узлы сети нажмите на *Элементы данных* рядом с 'Новый узел сети'.

Затем:

- выделите галочкой элемент данных 'Загрузка CPU' из списка
- выберите *Копировать* ниже списка
- выберите шаблон и скопируйте элемент данных туда



Все обязательные поля ввода отмечены красной звёздочкой.

- нажмите на *Копировать*

Если вы сейчас перейдете в *Настройка* → Шаблоны, вы должны увидеть 'Новый шаблон' с одним элементом данных.

Сейчас мы остановимся на одном элементе данных, но вы так же можете добавлять любые другие элементы данных, триггеры и другие сущности в шаблон, пока он не будет заполнен набором объектов для конкретных целей (мониторинг ОС, мониторинг одного приложения).

Соединение шаблона с узлом сети

Когда шаблон будет готов, остается только добавить его к узлу сети. Для этого перейдите в *Настройки* → Узлы сети, нажмите на 'Новый узел сети', откроется диалог его настроек, теперь перейдите на закладку **Шаблоны**.

Здесь, нажмите на *Добавить*, отметьте шаблон который мы создали ('Новый шаблон') и нажмите на *Выбрать*. Шаблон должен появиться в диалоге.

The screenshot shows the 'Hosts' page in Zabbix with the 'Templates' tab selected. A table lists a single template named 'New template' with an 'Unlink' action. Below the table is a search input and an 'Add' button. At the bottom are five buttons: 'Update', 'Clone', 'Full clone', 'Delete', and 'Cancel'.

Нажмите *Обновить* в диалоге для сохранения изменений. Теперь шаблон добавлен к узлу сети со всеми своими объектами, которые он имеет.

Как вы уже догадались, этот способ может быть применен и к любым другим узлам сети. Любые изменения элементов данных, триггеров и остальных объектов на уровне шаблона, изменения будут распространяться и на узлы сети с которыми этот шаблон соединен.

Соединение предопределенных шаблонов с узлами сети

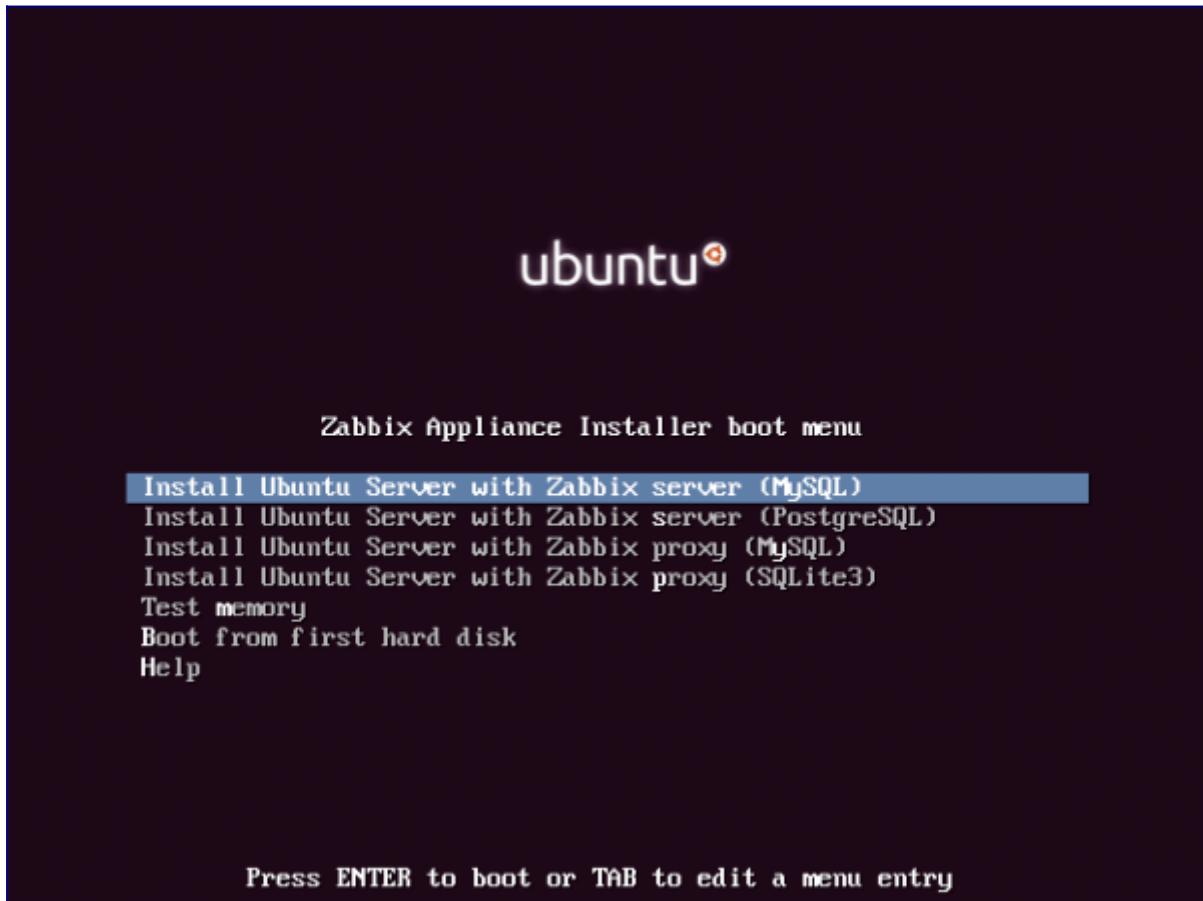
Как вы могли заметить, Zabbix поставляется с набором предопределенных шаблонов для различных ОС, устройств и приложений. Чтобы начать наблюдение как можно быстрее, вы можете присоединить соответствующий шаблон к узлу сети, но нужно учесть, что эти шаблоны нужно доработать для вашей среды. Некоторые проверки могут не понадобится и интервалы проверок могут быть слишком частые.

Доступна более подробная информация о [шаблонах](#).

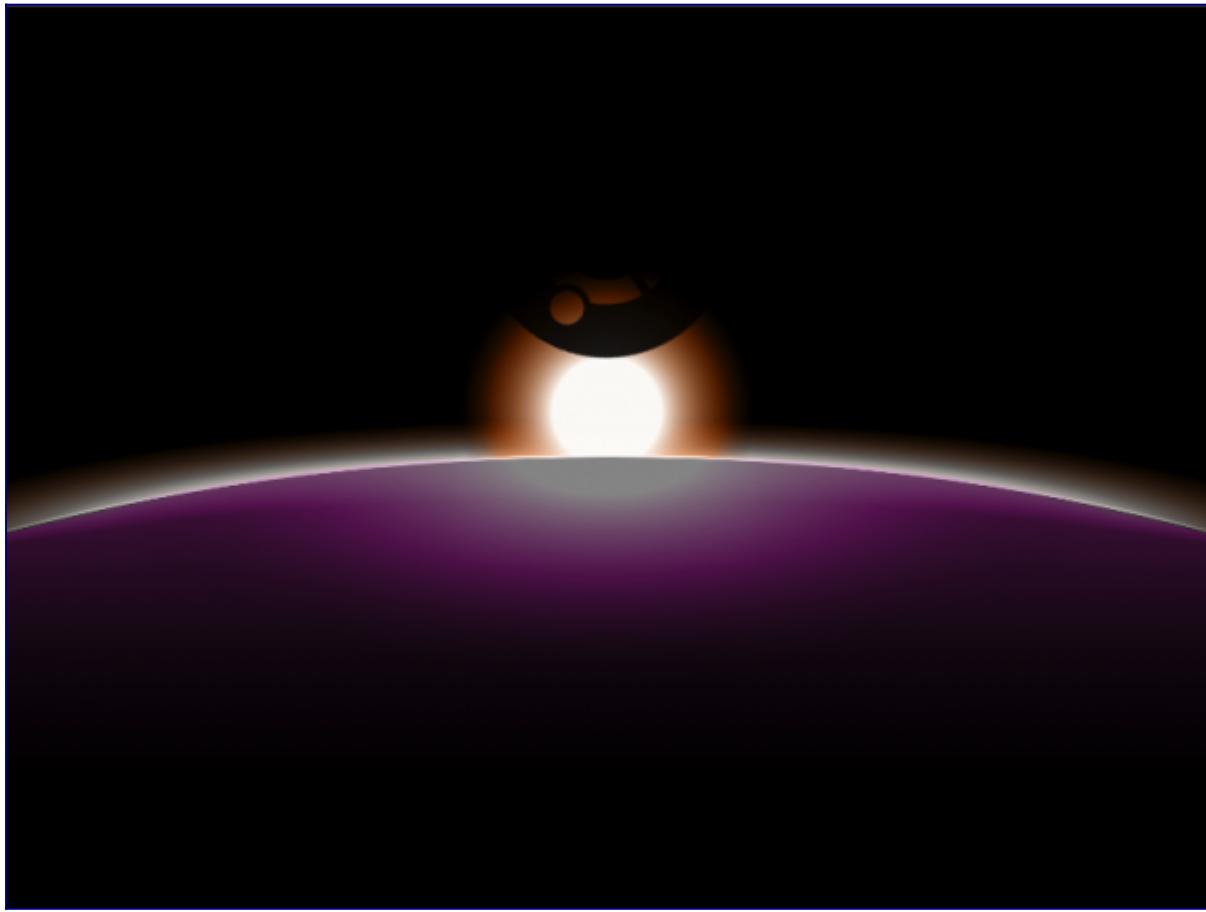
6. Готовое решение Zabbix

В качестве альтернативы ручной настройке или повторному использованию существующего сервера для Zabbix, пользователи могут [загрузить](#) готовое решение Zabbix или установочный образ CD готового решения Zabbix. Установочный образ CD готового решения Zabbix можно использовать для быстрого развертывания Zabbix сервера (MySQL), Zabbix сервера (PostgreSQL), Zabbix прокси (MySQL) и Zabbix прокси (SQLite 3).

Виртуальные машины готового решения Zabbix собраны с Zabbix сервером с поддержкой MySQL. Они собраны при помощи установочного образа CD готового решения Zabbix.



Загрузочное меню установки Zabbix CD/DVD



Загрузка готового решения Zabbix

Готовое решение Zabbix и установочный CD основаны на следующих версиях Ubuntu:

Версия готового решения Zabbix Версия Ubuntu

4.0.0 16.04.3

Готовое решение Zabbix доступно в следующих форматах:

- vmdk (VMware/Virtualbox)
 - OVF (Open Virtualisation Format)
 - KVM
 - HDD/flash image, USB stick
 - Live CD/DVD
 - Xen guest
 - Microsoft VHD (Azure)
 - Microsoft VHD (Hyper-V)

Для запуска, загрузите готовое решение и перейдите в вашем браузере по IP, который получен по DHCP: http://<ip_хоста>/zabbix

Готовое решение имеет настроенный Zabbix сервер, работающий с базой данных MySQL, также доступен и веб-интерфейс.

Готовое решение создано при помощи стандартной возможности Ubuntu/Debian, называемой Preseed файлами.

6.1 Изменения в настройках Ubuntu

Имеются несколько изменения, которые применены к базовой конфигурации Ubuntu.

6.1.1 Репозитории

Добавлен официальный [репозиторий](#) Zabbix в /etc/apt/sources.list:

```
## Zabbix repository
deb https://repo.zabbix.com/zabbix/4.0/ubuntu xenial main
deb-src https://repo.zabbix.com/zabbix/4.0/ubuntu xenial main
```

6.1.2 Межсетевой экран

Готовое решение использует брандмауэр iptables с предустановленными правилами:

- Открыт порт SSH (22 TCP);
- Открыты порты Zabbix агента (10050 TCP) и Zabbix траппера (10051 TCP);
- Открыты порты HTTP (80 TCP) и HTTPS (443 TCP);
- Открыт порт SNMP трапов (162 UDP);
- Открыты исходящие подключения на порт DNS (53 UDP) к 8.8.8.8 и 8.8.4.4;
- Открыты исходящие подключения на порт NTP (123 UDP);
- ICMP пакеты ограничены 5 пакетами в секунду;
- Все остальные входящие подключения отбрасываются.

6.1.3 Дополнительные пакеты

Добавлены различные простые утилиты, которые могут сделать работу с Zabbix и мониторингом в целом легче:

- iptables-persistent
- mc
- htop
- snmptrapfmt
- snmp-mibs-downloader

Некоторые из этих пакетов используются Zabbix, некоторые из них установлены, чтобы помочь пользователям настраивать/управлять настройками готового решения.

6.1.4 Использование статического IP адреса

По умолчанию готовое решение получает IP адрес по DHCP. Для того, чтобы указать статический IP адрес выполните:

- Войдите под пользователем root;
- Откройте файл `/etc/network/interfaces` в вашем любимом редакторе;
- `iface eth0 inet dhcp → iface eth0 inet static`
- Измените следующие строки после `iface eth0 inet static`:
 - `address <IP адрес готового решения>`
 - `netmask <маска подсети>`
 - `gateway <ваш адрес сетевого шлюза>`
- Выполните команды `sudo ifdown eth0 && sudo ifup eth0`.

Для получения более подробной информации о других возможных опциях обратитесь к официальной [документации](#) по Ubuntu.

Для настройки DNS, добавьте записи серверов имен в `/etc/resolv.conf`, указывая по одному серверу имен в одной строке: **nameserver 192.168.1.2**.

6.1.5 Изменение часового пояса

По умолчанию готовое решение использует UTC часовой пояс для системного времени. Для изменения часового пояса, скопируйте соответствующий файл из `/usr/share/zoneinfo` в `/etc/localtime`, например:

```
cp /usr/share/zoneinfo/Europe/Riga /etc/localtime
```

6.1.6 Изменения локали

Готовое решение содержит несколько изменений локалей:

- Содержит следующие языки: *en_US.UTF-8, ru_RU.UTF-8, ja_JP.UTF-8, cs_CZ.UTF-8, ko_KR.UTF-8, it_IT.UTF-8, pt_BR.UTF-8, sk_SK.UTF-8, uk_UA.UTF-8, fr_FR.UTF-8, pl.UTF-8*;
- Локаль по умолчанию *en_US.UTF-8*.

Эти изменения требуются для поддержки мультиязычного веб-интерфейса Zabbix.

6.1.7 Другие изменения

- Сеть настроена на получение IP адреса по DHCP;
- Утилита **fping** имеет права 4710 и ей назначен владелец группы **zabbix** - **suid** и только разрешено использование группе **zabbix**;
- ntpd настроен на синхронизацию с пулами публичных серверов: *ntp.ubuntu.com*;
- Используется LVM том с ext4 файловой системой.
- Добавлено “*UseDNS no*” в файл конфигурации SSH сервера */etc/ssh/sshd_config* для того, чтобы избежать долгих ожиданий при подключении к SSH;
- Демон snmpd отключен в файле конфигурации */etc/default/snmpd*.

6.2 Конфигурация Zabbix

Инсталляция готового решения Zabbix имеет следующие пароли и другие изменения конфигурации:

6.2.1 Учетные данные (имя пользователя:пароль)

Система:

- *appliance:zabbix*

Используйте “*sudo su*” команду вместе с паролем от “*appliance*” пользователя для получения привилегированных root прав.

База данных:

- *root:<случайный>*
- *zabbix:<случайный>*

Пароли к базе данных генерируются случайным образом в процессе инсталляции.

Пароль root записан в */root/.my.cnf* файле, таким образом не требуется вводить пароль, находясь под аккаунтом “root”.

Веб-интерфейс Zabbix:

- *Admin:zabbix*

Для изменения пароля пользователя базы данных, потребуются изменения в следующих местах:

- MySQL;
- */etc/zabbix/zabbix_server.conf*;
- */etc/zabbix/web/zabbix.conf.php*.

6.2.2 Размещения файлов

- Файлы конфигурации расположены в **/etc/zabbix**.
- Файлы журналов (логи) Zabbix сервера, прокси и агента расположены в **/var/log/zabbix**.
- Веб-интерфейс Zabbix расположен в **/usr/share/zabbix**.
- Домашняя папка пользователя **zabbix** - **/var/lib/zabbix**.

6.2.3 Изменения в настройках Zabbix

- Имя сервера для веб-интерфейса Zabbix изменено на “Zabbix Appliance”;
- Часовой пояс веб-интерфейса изменен на Europe/Riga - родина Zabbix (этот параметр можно изменить в файле `/etc/apache2/conf-available/zabbix.conf`);

6.2.4 Сохранение конфигурации

Если вы загрузили готовое решение с Live CD или по какой-то причине не можете иметь постоянного накопителя, вы можете создать архивную копию базы данных, включая все настройки и все собранные данные.

Для создания архивной копии, выполните:

```
mysqldump zabbix | bzip2 -9 > dbdump.bz2
```

Теперь вы можете переместить файл **dbdump.bz2** на другую машину.

Для восстановления из архивной копии, переместите копию в готовое решение и выполните:

```
bzcat dbdump.bz2 | mysql zabbix
```

Убедитесь, что Zabbix сервер не запущен при выполнении восстановления.

6.3 Доступ к веб-интерфейсу

По умолчанию доступ к веб-интерфейсу разрешен отовсюду.

Доступ к веб-интерфейсу может быть получен с `http://<хост>/zabbix`.

Эти параметры могут быть изменены в `/etc/apache2/conf-available/zabbix.conf`. Вы должны перезапустить веб-сервер после изменения этого файла. Чтобы это сделать зайдите на сервер по SSH под root и выполните:

```
service apache2 restart
```

6.4 Межсетевой экран

По умолчанию, открыты только те порты, которые перечислены в изменениях. Чтобы открыть дополнительные порты, просто измените файлы `"/etc/iptables/rules.v4"` или `"/etc/iptables/rules.v6"` и перезагрузите правила межсетевого экрана:

```
service iptables-persistent reload
```

6.5 Возможности мониторинга

Zabbix сервер скомпилирован с поддержкой следующих функций:

- SNMP
- IPMI
- Веб-мониторинг
- VMware мониторинг
- Оповещения по Jabber
- Оповещения по EZ Texting
- ODBC
- SSH2
- IPv6
- SNMP тралы
- Zabbix Java Gateway

6.6 SNMP трапы

Готовое решение Zabbix использует `snmptrapfmt` для обработки SNMP трапов. Утилита настроена на получение всех трапов отовсюду.

Аутентификация не требуется. Если вы желаете включить аутентификацию, вам необходимо изменить файл `/etc/snmp/snmptrapd.conf` и указать требуемые настройки аутентификации.

Все трапы записываются в `/var/log/zabbix/snmptrapfmt.log` файл. Этот файл ротируется при помощи `logrotate` до достижения размера файла в 2ГБ.

6.7 Обновление

Пакеты готового решения можно обновлять. Чтобы это сделать, выполните:

```
sudo apt install --only-upgrade 'zabbix.*'
```

6.8 Именование, скрипты запуска и другие скрипты

Представлены соответствующие скрипты запуска. Для контроля Zabbix сервера используйте следующие команды:

```
service zabbix-server status
```

Замените **server** на **agent** для демона Zabbix агента или на **proxy** для демона Zabbix прокси.

6.8.1 Увеличение доступного дискового пространства

Создайте резервную копию всех данных до выполнения любого из шагов.

Доступного дискового пространства в готовом решении может быть недостаточно. В этом случае имеется возможность расширить используемый диск. Для этого сначала расширьте блочное устройство в вашей среде виртуализации, затем выполните эти шаги.

Запустите `fdisk` для изменения размера раздела. Выполните под `root`:

```
fdisk /dev/sda
```

Эта команда запустит `fdisk` для диска `sda`. Затем, переключитесь на использование секторов и введите:

```
u
```

Не отключайте совместимость с DOS нажатием **c**. При отключении этой совместимости раздел будет поврежден.

Затем удалите существующий раздел и создайте новый с нужным размером. В большинстве случаев вы будете выбирать доступный максимум, что расширит файловую систему до размера, который вы сделали доступным для виртуального диска. Для этого введите следующую последовательность в `fdisk`:

```
d  
n  
p  
1  
(принимаем по умолчанию 63)  
(принимаем по умолчанию max)
```

Если вы хотите оставить пространство для дополнительных разделов (`swap` и т.д.), вы можете ввести другое значение для `last sector` (последний сектор). По завершении сохраните изменения введя:

```
w
```

После создания раздела (с нового диска или расширения существующего) создайте физический том:

```
pvcreate /dev/sdb1
```

Имя раздела /dev/sdb1 используется как пример; в вашем случае имя диска и номер раздела могут быть другими. Вы можете проверить номер раздела, выполнив `fdisk -l /dev/sdb` команду.

Проверьте только что созданный физический том:

```
pvdisplay /dev/sdb1
```

Проверьте доступные физические тома. Должно быть 2 тома zabbix-vg и только что созданный:

```
pvs
```

Расширьте существующую группу томов при помощи только что созданного физического тома:

```
vgextend zabbix-vg /dev/sdb1
```

Проверьте группу томов “zabbix-vg”:

```
vgdisplay
```

Теперь расширьте ваш логический том, заняв все свободное пространство:

```
lvextend -l +100%FREE /dev/mapper/zabbix--vg-root
```

Измените размер вашего корневого тома (можно выполнить на живой системе):

```
resize2fs /dev/mapper/zabbix--vg-root
```

Перезагрузите виртуальную машину (так как раздел, который мы изменили, в настоящее время используется). Теперь файловая система должна быть увеличена до размера раздела. Проверьте “/dev/mapper/zabbix–vg-root” том:

```
df -h
```

6.9 Заметки о специфичных форматах

6.9.1 Xen

Конвертация образа для XenServer

Для использования Xen образов в Citrix Xenserver вам необходимо сконвертировать образ диска. Чтобы это сделать:

- Создайте виртуальный диск, который будет по крайней мере такого же размера как и образ
- Узнайте UUID этого диска

```
xe vdi-list params=all
```

- Если имеется много дисков, их можно отфильтровать по параметру имени *name-label* (имя-метка), как назначенные при создании виртуального диска
- Импортируйте образ

```
xe vdi-import filename="image.raw" uuid=<UUID>"
```

Инструкции с *Brian Radford blog*.

6.9.2 VMWare

Образы в формате *vmdk* используются напрямую в продуктах VMWare Player, Server и Workstation. Для использования в ESX, ESXi и vSphere образы должны быть сконвертированы с использованием [конвертер VMWare](#).

6.9.3 Образ HDD/flash (raw)

```
dd if=./zabbix_appliance_4.0.0_x86_64.raw of=/dev/sdc bs=4k conv=fdatasync
```

Замените */dev/sdc* на ваше Flash/HDD устройство диска.

6.10 Известные проблемы

7. Настройка

7.1 Узлы сети и группы узлов сети

Что такое "узел сети"?

Типичные узлы сети в Zabbix - это устройства, которые необходимо мониторить (сервера, рабочие станции, коммутаторы и т.д.).

Создание узлов сети является одной из первоочередных задач мониторинга в Zabbix. Например, если вы хотите мониторить некоторые параметры на сервере “x”, сначала вы должны создать узел сети называемый, скажем, “Сервер X” и только затем вы можете смотреть в сторону добавления элементов данных для мониторинга сервера.

Узлы сети объединяются в группы узлов сети.

Перейти к [созданию и настройке узла сети](#).

7.1.1 Настройка узла сети

Обзор

Для настройки узла сети в веб-интерфейсе Zabbix, сделайте следующее:

- Перейдите в: *Настройка* → *Узлы сети*
- Нажмите на *Создать узел сети* в верхнем правом углу (или на имя узла сети для изменения существующего узла сети)
- В диалоге введите параметры для узла сети

Вы также можете использовать кнопки *Клонировать* и *Полное клонирование* в диалоге существующего узла для создания нового узла сети. Нажатие на *Клонировать* сохранит все параметры узла сети и все соединения с шаблонами (с сохранением всех объектов из этих шаблонов). *Полное клонирование* дополнительно сохранит все напрямую добавленные объекты (элементы данных, триггеры, графики и группы элементов данных).

Обратите внимание: При клонировании узла сети, он наследует все сущности из шаблонов, как они настроены изначально в шаблонах. Любые сделанные изменения в этих сущностях на уровне узла сети (такие как интервал опроса элементов данных, измененное регулярное выражение или добавленные прототипы к правилу низкоуровневого обнаружения) не будут скопированы в новый узел сети; они останутся так как есть в шаблоне.

Настройка

Вкладка **Узел сети** содержит общие атрибуты узла сети

Hosts

Host Templates IPMI Macros Host inventory Encryption

* Host name: Zabbix server_1
Visible name: Zabbix server
* Groups: Linux servers, Zabbix servers, Discovered hosts
Select

* At least one interface must exist.

Agent interfaces: IP address: 192.168.3.220, DNS name: , Connect to: IP, Port: 10050
Add

SNMP interfaces: IP: 127.0.0.1, DNS: , Connect to: IP, Port: 161
Use bulk requests
Add

JMX interfaces
Add

IPMI interfaces
Add

Description: Added on 2018-02-07

Monitored by proxy: (no proxy)
Enabled:

Add Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Описание

Введите уникальное имя узла сети. Разрешены буквенно-цифровые символы, пробелы, точки, тире и подчёркивания.

Имя узла сети

Обратите внимание: При работающем Zabbix агенте на настраиваемом вами узле сети, параметр *Hostname* из [файла конфигурации](#) агента должен иметь такое же значение, как и введенное здесь имя узла сети. Имя из этого параметра необходимо для обработки [активных проверок](#).

Видимое имя

Если вы укажите это имя, именно оно будет видимо в списках, картах и прочем. Этот атрибут имеет поддержку UTF-8.

Группы

Выберите группы узлов сети к которым будет принадлежать узел сети. Узел сети должен принадлежать по крайней мере одной группе узлов сети. Новую группу можно создать и присоединить к узлу сети, добавив несуществующее имя группы. Для узлов сети поддерживаются несколько типов интерфейсов: *агент*, *SNMP*, *JMX* и *IPMI*.

Интерфейсы

Чтобы добавить новый интерфейс, нажмите на *Добавить* в блоке *Интерфейсы* и введите информацию *IP/DNS*, *Подключаться через* и *Порт*.

Обратите внимание: Интерфейсы, которые используются какими-нибудь элементами данных, не могут быть удалены и у них ссылка *Удалить* отображается серым.

Опция *Использовать массовые запросы* у SNMP интерфейсов позволяет включать/отключать [массовую обработку](#) SNMP запросов по каждому интерфейсу отдельно.

IP адрес

IP адрес узла сети (опционально).

DNS имя

DNS имя узла сети (опционально).

Параметр	Описание
Подключаться через	Нажатие на соответствующую кнопку подскажет Zabbix серверу что использовать для получения данных от агентов: IP - Подключение к узлу сети, используя IP адресу (рекомендуется) DNS - Подключение к узлу сети, используя DNS имя
Порт	Номер TCP/UDP порта. Значения по умолчанию: 10050 для Zabbix агента, 161 для SNMP агента, 12345 для JMX и 623 для IPMI.
По умолчанию	Отметьте, чтобы интерфейс считался по умолчанию.
Наблюдение через прокси	Узел сети может наблюдаться через Zabbix сервер или через один из Zabbix прокси: (без прокси) - узел сети наблюдается Zabbix сервером Имя прокси - узел сети наблюдается через Zabbix прокси “Имя прокси”
Активировано	Отметьте, чтобы сделать узел сети активным, готовым к мониторингу. Если не отмечено, узел сети неактивен, таким образом не наблюдается.

Вкладка **Шаблоны** позволяет вам присоединить [шаблоны](#) к узлу сети. Все объекты (элементы данных, триггеры, графики и группы элементов данных) будут унаследованы из шаблона.

Для присоединения нового шаблона, начните вводить имя шаблона в поле *Соединить с новыми шаблонами* до тех пор пока не появится список совпадающих шаблонов. Прокрутите список и выберете нужное. Когда все шаблоны будут выбраны, нажмите на *Добавить*.

Для отсоединения шаблона используйте одну из двух опций в блоке *Присоединенные шаблоны*:

- *Отсоединить* - отсоединение шаблона, с сохранением всех его элементов данных, триггеров и графиков
- *Отсоединить и очистить* - отсоединение шаблона и удаление всех его элементов данных, триггеров и графиков

Имена перечисленных шаблонов являются ссылками, которые ведут на диалог настройки шаблона.

Вкладка **IPMI** содержит атрибуты управления по IPMI.

Параметр	Описание
Метод аутентификации	Выбор алгоритма аутентификации.
Уровень привилегий	Выбор уровня привилегий.
Имя пользователя	Имя пользователя для аутентификации.
Пароль	Пароль для аутентификации.

Вкладка **Макросы** позволяет вам задать [пользовательские макросы](#) на уровне узла сети. Вы также здесь можете просмотреть макросы уровня шаблона и глобальные макросы, если выберите опцию *Макросы узла сети и унаследованные*. Это то место, где отображаются все определенные пользовательские макросы для этого узла сети со своими раскрытыми значениями, а также информация о том откуда эти макросы.

MACRO	EFFECTIVE VALUE	TEMPLATE VALUE	GLOBAL VALUE (CONFIGURE)
[\$DSN]	test	Change	“test”
[\$HOST_MACRO]	srktidmsidc334	Remove	
[\$NESTED_TEMPLATE_MACRO]	25864b	Change	“Template App Zabbix Agent: “25864b”
[\$SNMP_COMMUNITY]	public	Change	“public”
[\$TEMPLATE_MACRO]	25864	Remove	“Template OS Linux_b: “25864”

Для удобства имеются ссылки на настройку соответствующих шаблонов и глобальных макросов. Также имеется возможность изменить макрос уровня шаблона/глобальный на уровне узла сети, фактически создав копию этого макроса на узле сети.

Вкладка **Инвентарные данные узла сети** позволяет вам вручную ввести информацию о инвентарных данных узла сети. Вы также можете выбрать Автоматически заполнение инвентарных данных, или деактивировать заполнение инвентарных данных для этого узла сети.

Вкладка **Шифрование** позволяет вам требовать шифрование соединений с узлом сети.

Параметр	Описание
Подключения к узлу сети	Каким образом Zabbix сервер или прокси подключаются к Zabbix агенту на хосте: без шифрования (по умолчанию), используя PSK (pre-shared key) или сертификат.
Соединения с узла сети	Выберите какой тип подключений разрешен 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	Разрешенный эмитент сертификата. Сертификат сначала подтверждается CA (центром сертификации). Если он действительный, подписан с помощью CA, тогда можно использовать поле <i>Издатель</i> для более строгого ограничения разрешенных CA. Это поле предназначено для использования, если ваша Zabbix инсталляция использует сертификаты от нескольких CA. Если поле не заполнено, тогда принимается любой CA.
Тема	Разрешенная тема сертификата. Сертификат сначала подтверждается CA. Если он действительный, подписан с помощью CA, тогда можно использовать поле <i>Тема</i> , чтобы разрешить только одно значение строки <i>Тема</i> . Если поле пустое, тогда принимается любой сертификат, подписанный настроенным CA.
Идентификатор PSK	Строка идентификации pre-shared key.
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 цифр (32-байт PSK), если Zabbix использует библиотеку mbedtls (PolarSSL). Пример: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Создание группы узлов сети

Для создания группы узлов сети в веб-интерфейсе Zabbix, выполните следующее:

- Перейдите в: *Настройка* → *Группы узлов сети*
- Нажмите на *Создать группу узлов сети* в верхнем правом углу экрана
- В диалоге введите параметры группы

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя группы	Введите уникальное имя группы узлов сети. Чтобы создать вложенную группу узлов сети, используйте прямую косую черту '/', например Европа/Латвия/Рига/Zabbix сервера.Вы можете создать эту группу, даже если ни одна из родительских группы узлов сети

Параметр	Описание
	(Европа/Латвия/Рига) не существуют. В этом случае создание этих родительских групп узлов сети зависит от желания пользователя; они не будут созданы автоматически.
	Косые черты в начале и в конце строки, несколько косых черт подряд не поддерживаются. Экранирование символа '/' не поддерживается.
<i>Применить права доступа ко всем подгруппам</i>	Вложенное представление групп узлов сети поддерживается начиная с Zabbix 3.2.0.
	Эта опция доступна только Zabbix Супер Админ пользователям и только при редактировании существующей группы узлов сети.
	Выберите эту опцию и нажмите на Обновить, чтобы применить одинаковый уровень прав доступа ко всем вложенным группам узлов сети. Для групп пользователей, которые могли иметь разные права доступа , назначенные на вложенные группы узлов сети, будет применён уровень прав доступа родительской группы узлов сети ко всем вложенным группам.
	Это действие является однократной опцией, которая не хранится в базе данных.
	Опция поддерживается начиная с Zabbix 3.4.0.

Права доступа к вложенным группам узлов сети

- При создании дочерней группы узлов сети у уже существующей родительской группы узлов сети права доступа [группы пользователей](#) к дочерней группе узлов сети наследуются от родителя (например, при создании Рига/Zabbix сервера, если Рига уже существует)
- При создании родительской группы узлов сети при уже существующей дочерней группе узлов сети права доступа на родительскую группу узлов сети не задаются (например, при создании Рига, если Рига/Zabbix сервера уже существует)

7.1.2 Инвентарные данные

Обзор

Вы можете хранить инвентарные данные сетевых устройств в Zabbix.

В веб-интерфейсе Zabbix существует специальное меню *Инвентарные данные*. Тем не менее, по началу вы не увидите в меню никаких данных и это место не там где вам необходимо ввести 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 *Инвентарные данные* вы можете ввести такие детали как тип устройства, серийный номер, местоположение, ответственное лицо и прочие - данные, которые будут заполнять инвентарные данные.

Если URL включен в инвентарные данные и он начинается с 'http' или 'https', то в результате он будет для нажатия в разделе *Инвентарных данных*.

Автоматический режим

Также инвентарные данные узлов сети могут заполняться автоматически. Чтобы эта функция работала, при настройке узла сети в режиме инвентарных данных на вкладке *Инвентарные данные узла сети* нужно выбрать *Автоматически*.

Затем вы можете [настроить элементы данных узла сети](#), которые будут заполнять любые поля инвентарных данных своими значениями, указав поле назначения с соответствующим атрибутом (называемый *Заполнять поле инвентарных данных*) при настройке элемента данных.

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 инвентарных данных

Режим инвентарных данных можно выбрать в диалоге настройки узла сети.

Режим инвентарных данных по умолчанию для новых узлов сети выбирается на основе настройки *Режим инвентаризации узлов сети по умолчанию* в *Администрирование* → *Общие* → [Прочие](#).

По узлам сети добавляемым с помощью действий по сетевым обнаружениям или авторегистрации имеется возможность указать операцию *Установить режим инвентарных данных узлов сети*, выбрав ручной или автоматический режим. Эта операция переопределит настройку *Режим инвентаризации узлов сети по умолчанию*.

Просмотр инвентарных данных

Детали всех указанных инвентарных данных доступны в меню *Инвентарные данные*.

В *Инвентарные данные* → *Обзор* вы можете сгруппировать и просмотреть доступные данные по различным полям инвентарных данных.

В *Инвентарные данные* → *Узлы сети* вы можете увидеть все узлы сети содержащие инвентарные данные. При нажатии на имя узла сети, в диалоге будут раскрыты все детали.

Host inventory

[Overview](#) [Details](#)

Host name **Zabbix server_1**

Visible name Zabbix server

Agent interfaces	IP address	DNS name	Connect to	Port	Default
	192.168.3.220		<input type="radio"/> IP <input type="radio"/> DNS	10050	<input checked="" type="radio"/>

SNMP interfaces	IP address	DNS name	Connect to	Port	Default
	127.0.0.1		<input type="radio"/> IP <input type="radio"/> DNS	161	<input checked="" type="radio"/>

OS Linux linux-qvvt 3.11.10-21-default #1 SMP Mon Jul 21 15:28:46 U

Description Added on 2015-07-28.

Monitoring [Web](#) [Latest data](#) [Triggers](#) [Problems](#) [Graphs](#) [Screens](#)

Configuration [Host Applications](#) 13 [Items](#) 81 [Triggers](#) 47 [Graphs](#) 12 [Discovery](#) 3 [Web](#) 1

[Cancel](#)

Вкладка **Обзор** отображает:

Параметр	Описание
<i>Имя узла сети</i>	Имя узла сети. Если нажать на имя, появится меню с заданными для этого узла сети скриптами. Имя узла сети отображается с оранжевой иконкой в случае, если узел сети находится в обслуживании.
<i>Видимое имя</i>	Видимое имя узла сети (если задано).
<i>Интерфейсы (Agent, SNMP, JMX, IPMI) узла сети</i>	Этот блок отображает детали всех интерфейсов настроенных у узла сети.
<i>ОС</i>	Поле операционная система из инвентарных данных узла сети (если задано).
<i>Аппаратное обеспечение</i>	Поле аппаратного обеспечения из инвентарных данных узла сети (если задано).
<i>Программное обеспечение</i>	Поле программного обеспечения из инвентарных данных узла сети (если задано).
<i>Описание</i>	Описание узла сети.
<i>Мониторинг</i>	Ссылки на разделы мониторинга этого узла сети: <i>Веб</i> , <i>Последние данные</i> , <i>Триггеры</i> , <i>Проблемы</i> , <i>Графики</i> , <i>Комплексные экраны</i> .
<i>Настройка</i>	Ссылки на разделы настройки этого узла сети: <i>Узел сети</i> , <i>Группы элементов данных</i> , <i>Элементы данных</i> , <i>Триггеры</i> , <i>Графики</i> , <i>Обнаружение</i> , <i>Веб</i> . Количество добавленных объектов указано в скобках после каждой ссылки.

Вкладка **Детали** отображает все поля инвентарных данных, которые заполнены (не пустые).

Макросы инвентарных данных

Имеются доступные узлу сети макросы инвентарных данных {INVENTORY.*}, которые используются в оповещениях, например:

“Сервер {INVENTORY.LOCATION1} имеет проблему, ответственное лицо {INVENTORY.CONTACT1}, номер телефона {INVENTORY.POC.PRIMARY.PHONE.A1}.”

Для получения более подробных сведений, перейдите на страницу [Макросы поддерживаемые по назначению](#).

7.1.3 Массовое обновление

Обзор

Порой вы можете захотеть изменить какие-нибудь атрибуты для нескольких узлов сети одновременно. Вместо открытия каждого узла сети для редактирования, вы можете воспользоваться функцией массового обновления.

Использование массового обновления

Для массового обновления каких-либо узлов сети, выполните следующее:

- Отметьте узлы сети из списка, которые вы хотите обновить, из списка [узлов сети](#)
- Нажмите на *Массовое обновление* ниже списка
- Перейдите в необходимую вкладку атрибутов (*Узел сети*, *Шаблоны*, *IPMI*, *Инвентаризация* или *Шифрование*)
- Отметьте атрибуты для обновления и введите для них новые значения

Hosts

Host Templates IPMI Inventory Encryption

Replace host groups Discovered hosts
type here to search

Add new or existing host groups type here to search

Remove host groups Zabbix servers
type here to search

Description Original

Monitored by proxy Remote proxy

Status Original

Update Cancel

Заменить группы узла сети уберет узел сети из текущих групп и заменит на указанную(ые) в данном поле.

Добавить новые или существующие группы узлов сети позволяет выбрать дополнительные группы узла сети из существующих или назначить совершенно новые группы узлов сети для выбранных узлов сети.

Удалить группы узлов сети позволяет удалить выбранные группы узлов сети с узлов сети. В случае, когда узлы сети уже находятся в выбранных группах, тогда узлы сети будут удалены из этих групп. В случае, когда узлы сети не находятся в выбранных группах, тогда ничего не будет добавлено или удалено. В случае, когда одни и те же группы узлов сети заменяются и удаляются одновременно, узлы сети фактически остаются без групп.

Эти поля имеют функцию авто-дополнения - начните набирать и в них появится выпадающий список совпадающих групп узлов сети. Если группа узла сети новая, она также появится в выпадающем списке и будет отмечена в конце строки как (новая). Просто прокрутите вниз для выбора.

Host Templates IPMI Inventory Encryption

Link templates Template SNMP Device
type here to search

Replace
 Clear when unlinking

Update Cancel

Для обновления присоединенных шаблонов на вкладке **Шаблоны**, отметьте *Присоединить шаблоны* и начните вводить имя шаблона в поле с автодополнением до появления выпадающего списка

предлагающего совпадающие доступные шаблоны. Просто используйте прокрутку для выбора присоединяемого шаблона.

Опция *Заменить* позволяет добавить новый шаблон к узлу сети взамен любых других шаблонов соединенных с узлам сети ранее. Опция *Очистить при отсоединении* позволяет не только отсоединить какие-либо ранее присоединенные шаблоны, но также удалить все элементы унаследованные от них (элементы данных, триггеры, и т.д.).

The screenshot shows the 'IPMI' tab selected in a configuration interface. It includes fields for 'IPMI authentication algorithm' (checkbox), 'IPMI privilege level' (checkbox set to 'Operator'), 'IPMI username' (checkbox), 'IPMI password' (checkbox), and two buttons at the bottom: 'Update' (blue) and 'Cancel'.

The screenshot shows the 'Inventory' tab selected. It includes fields for 'Inventory mode' (checkbox set to 'Disabled', with 'Manual' and 'Automatic' options), 'Type' (checkbox set to 'Switch'), 'Type (Full details)' (checkbox), 'Name' (checkbox), and 'Alias' (checkbox).

Чтобы появилась возможность массового обновления полей инвентарных данных, *Режим инвентарных данных* должен быть задан значениями 'Вручную' или 'Автоматически'.

The screenshot shows the 'Encryption' tab selected. It includes a 'Connections' checkbox, a section for 'Connections to host' with radio buttons for 'No encryption', 'PSK' (selected), and 'Certificate', and sections for 'Connections from host' (radio buttons for 'No encryption' (selected), 'PSK', and 'Certificate') and 'PSK identity' and 'PSK' input fields.

Когда сделаете все необходимые изменения, нажмите на кнопку *Обновить*. Атрибуты будут обновлены соответственно всем выбранным узлам сети.

7.2 Элементы данных

Обзор

Элементы данных являются теми сущностями, кто собирает данные с узла сети.

Как только вы настроили узел сети, вам необходимо добавить какие-нибудь элементы данных для мониторинга, для начала фактического сбора данных.

Каждый элемент данных является отдельной метрикой. Один из способов быстро добавить большое количество элементов данных это присоединить один из предопределенных шаблонов к узлу сети. Для оптимизации производительности системы, вам возможно потребуется выполнить тонкую настройку шаблонов, чтобы у вас было ровно столько элементов данных, а так же насколько часто эти элементы данных будут собирать данные, насколько это действительно необходимо.

В каждом элементе данных вы указываете какого рода данные будут собираться с узла сети.

Для этой цели воспользуйтесь [ключем элемента данных](#). Таким образом 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.

Для того чтобы указать дополнительные параметры для ключа, вы должны включить их в квадратные скобки после имени ключа. Таким образом, **system.cpu.load[avg5]** будет возвращать загрузку процессора за последние 5 минут, а **net.if.in[eth0]** вернет входящий трафик на интерфейсе eth0.

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](#).

7.2.1 Создание элемента данных

Обзор

Для создания элемента данных в веб-интерфейсе Zabbix, выполните следующее:

- Перейдите в: *Настройка → Узлы сети*
- Нажмите на *Элементы данных* в строке с узлом сети
- Нажмите на *Создать элемент данных* в правом верхнем углу экрана
- В диалоге введите параметры элемента данных

Вы также можете создать элемент данных, открыв уже существующий элемент данных, после чего нажать на кнопку *Клонировать* и затем сохранить под другим именем.

Настройка

Вкладка **Элемент данных** содержит следующие атрибуты элементов данных.

Item Preprocessing

* Name	Incoming network traffic on eth0		
Type	Zabbix agent		
* Key	net.if.in[eth0]		
* Host interface	192.168.6.87 : 10050		
Type of information	Numeric (unsigned)		
Units	bps		
* Update interval	1m		
Custom intervals	Type	Interval	Period
	Flexible	Scheduling	50s
	Flexible	Scheduling	1-7,00:00-24:00
	Flexible	Scheduling	[\$FLEX_INTERVAL]
	Flexible	Scheduling	[\$FLEX_PERIOD]
	Flexible	Scheduling	wd1-5h9-18
	Flexible	Scheduling	[\$SCHEDULING]
	Add		
* History storage period	1w		
* Trend storage period	365d		
Show value	As is show value mappings		
New application			
Applications	<ul style="list-style-type: none"> -None- CPU Filesystems General Memory Network interfaces OS Performance Processes Security 		
Populates host inventory field	-None-		
Description			
Enabled	<input checked="" type="checkbox"/>		
	Add	Cancel	

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	Имя элемента данных. Поддерживаются следующие макросы, однако, их использование устарело: \$1, \$2...\$9 - ссылка на первый, второй... девятый параметры ключа элемента данных Например: Свободно дискового пространства в \$1 Если ключ элемента данных "vfs.fs.size[/,free]", описание автоматически изменится на "Свободно дискового пространства в /"
Тип	Тип элемента данных. Смотрите отдельные разделы по типам элементов данных .
Ключ	Ключ элемента данных.

Параметр	Описание
	<p>Поддерживаемые ключи элементов данных описаны в отдельных разделах по типам элементов данных.</p> <p>Ключ должен быть уникальным в пределах одного узла сети.</p> <p>Если тип ключа 'Zabbix агент', 'Zabbix агент (активный)', 'Простая проверка' или 'Zabbix агрегированный', то значение ключа должно поддерживаться Zabbix агентом или Zabbix сервером.</p> <p>Смотрите также: корректный формат ключа.</p>
Интерфейс узла сети	<p>Выбор интерфейса узла сети. Это поле доступно при изменении элемента данных на уровне узла сети.</p> <p>Тип хранения данных в базе данных после всех преобразований, если таковые имеются.</p> <p>Числовой (целое положительное) - целое 64битное число</p> <p>Числовой (с плавающей точкой) - число с плавающей точкой</p> <p>Отрицательные значения могут храниться.</p> <p>Допустимые значения -999999999999.9999 до 999999999999.9999.</p>
Тип информации	<p>Начиная с Zabbix 2.2, также поддерживается получение значений в научном представлении. Например, 1e+7, 1e-4.</p> <p>Символ - короткие текстовые данные</p> <p>Журнал - длинные текстовые данные с необязательными свойствами для журналов (штамп времени, источник, важность, logeventid).</p> <p>Текст - длинные текстовые данные. Смотрите также ограничения по текстовым данных.</p> <p>Если указан символ единицы измерения, Zabbix добавит пост обработку полученного значения и отобразит его с заданным постфиксом единицы измерения. По умолчанию, если исходное значение превышает 1000, оно делится на 1000 и так отображается. Например, если вы задали bps и полученное значение равно 881764, оно будет отображено как 881.76 Kbps.</p> <p>Для единиц измерения B (байт), Bps (байты в секунду) используется специальная обработка, при которой значение делится на 1024. Таким образом, если единица измерения указана как B или Bps, Zabbix будет отображать:</p> <ul style="list-style-type: none"> 1 как 1B/1Bps 1024 как 1KB/1KBps 1536 как 1.5KB/1.5KBps <p>Специальная обработка используется и для следующих единиц измерения связанных со временем:</p> <p>unixtime - переводится в “тттт.мм.дд чч:мм:сс”. Для корректного перевода, возвращаемое значение должно быть с типом данных Числовой (целое положительное).</p> <p>uptime - переводится в “чч:мм:сс” или в “N дней, чч:мм:сс”</p> <p>Например, если вы получили значение равное 881764 (секунд), оно отобразится как “10 дней, 04:56:04”</p> <p>s - переводится в “тттт ммм ддд ччч ммм ссс мс”; параметр рассматривается как количество секунд.</p> <p>Например, если вы получили значение равное 881764 (секунд), оно будет отображаться как “10д 4ч 56м”</p> <p>Отображаются только 3 верхних основы, такие как “1м 15д 5ч” или “2ч 4м 46с”. В случае, если нет дней, то тогда отображаются только два уровня - “1м 5ч” (минуты, секунды или миллисекунды не будут отображаться). Будет переведено в “< 1 ms”, если значение меньше чем 0.001.</p> <p><i>Обратите внимание</i>, что если единица измерения представлена с ! префиксом, тогда к значениям элементов данных префиксы/обработка единиц измерения применяется не будут. Смотрите чёрный список единиц измерения.</p>
Интервал обновления	<p>Получение нового значения по этому элементу данных каждые N секунд. Максимально допустимый интервал обновления - 86400 секунд (1 день).</p>

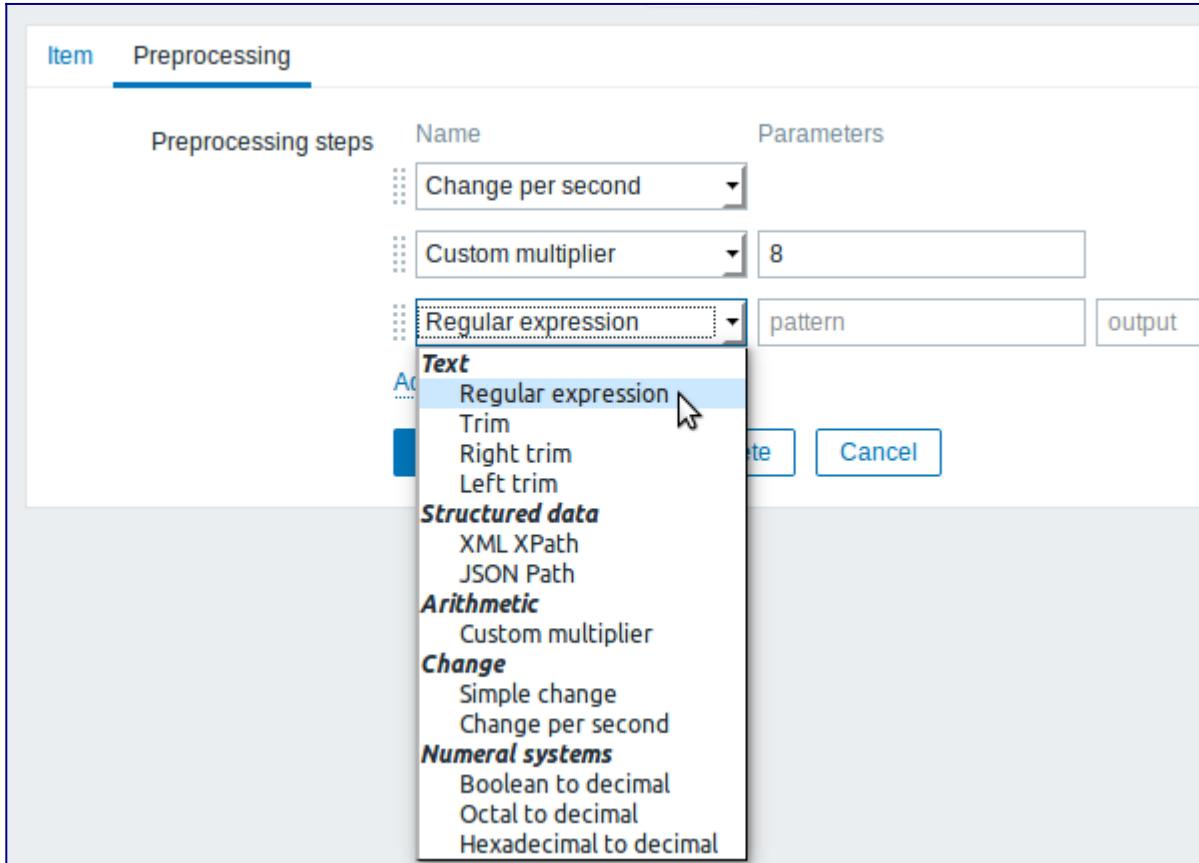
Параметр	Описание						
Пользовательские интервалы	<p><u>Функции времени</u> поддерживаются, например, 30s, 1m, 2h, 1d. Поддерживаются <u>пользовательские макросы</u>.</p> <p><i>Обратите внимание:</i> Если задано значение равное '0', элемент данных не будет обрабатываться. Однако, если также существует пользовательский интервал (гибкий/по расписанию) с ненулевым значением, элемент данных будет обработан в течении длительности пользовательского интервала.</p> <p><i>Обратите внимание,</i> что у существующего пассивного элемента данных можно выполнить опрос значения немедленно, нажав на <u>кнопку Проверить сейчас</u>.</p> <p>Вы можете создавать пользовательские правила проверки элемента данных:</p> <p>Гибкий - создание исключений из <i>Интервала обновления</i> (интервал с другой частотой обновления)</p> <p>По расписанию - создание пользовательского расписания проверки.</p> <p>Для получения более подробной информации смотрите <u>Пользовательские интервалы</u>.</p> <p><u>Функции времени</u> поддерживаются в поле <i>Интервал</i>, например, 30s, 1m, 2h, 1d. Поддерживаются <u>пользовательские макросы</u>.</p> <p>Проверка по расписанию поддерживается начиная с Zabbix 3.0.0.</p> <p><i>Обратите внимание:</i> Недоступно для активных элементов данных Zabbix агента. Количество дней хранения в базе данных детальной истории (от 1 часа до 25 лет). Более старые данные будут удалены с помощью функции автоматической очистки истории базы данных.</p> <p>Хранится в секундах. <u>Функции времени</u> поддерживаются, например, 2h, 1d. Поддерживаются <u>пользовательские макросы</u>.</p> <p>Данное значение можно переопределить глобально в <i>Администрирование → Общие → Очистка истории</i>. Если опция активирована, то вы увидите предупреждение:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding: 2px;">History storage period</td> <td style="padding: 2px; border: 1px solid #ccc;">1w</td> <td style="padding: 2px;">Overridden by global housekeeping settings (1d)</td> </tr> </table> <p>Рекомендуется хранить записанные значения как можно меньшее количество дней для уменьшения размера истории в базе данных. Вместо долговременного хранения истории значений, вы можете хранить более долгий срок данные динамики изменений.</p> <p>Смотрите также <u>История и динамика изменений</u>.</p> <p>Хранение усредненных значений (ежечасные мин, макс, сред, количество) детальной истории N дней в базе данных (от 1 дня до 25 лет). Более старые данные будут удалены с помощью функцией автоматической очистки истории базы данных.</p> <p>Хранится в секундах. <u>Функции времени</u> поддерживаются, например, 24h, 1d. Поддерживаются <u>пользовательские макросы</u>.</p> <p>Данное значение можно переопределить глобально в <i>Администрирование → Общие → Очистка истории</i>. Если опция активирована, то вы увидите предупреждение:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding: 2px;">Trend storage period</td> <td style="padding: 2px; border: 1px solid #ccc;">365d</td> <td style="padding: 2px;">Overridden by global housekeeping settings (7d)</td> </tr> </table> <p><i>Обратите внимание:</i> Хранение динамики изменений недоступно для не числовых данных - символ, журнал и текст.</p> <p>Смотрите также <u>История и динамика изменений</u>.</p> <p>Применение преобразования значений к этому элементу данных. Преобразование значений не меняет полученные значения, оно служит только для отображаемых данных.</p> <p>Работает только с целыми числовыми элементами данных.</p> <p>Например, "Windows service states".</p>	History storage period	1w	Overridden by global housekeeping settings (1d)	Trend storage period	365d	Overridden by global housekeeping settings (7d)
History storage period	1w	Overridden by global housekeeping settings (1d)					
Trend storage period	365d	Overridden by global housekeeping settings (7d)					
Период хранения динамики изменений	<p>Доступен только для элементов данных типа Журнал. Поддерживаемые значения:</p> <ul style="list-style-type: none"> * y: Год (1970-2038) * M: Месяц (01-12) * d: День (01-31) * h: Час (00-23) 						
Отображение значений	<p>Применение преобразования значений к этому элементу данных. Преобразование значений не меняет полученные значения, оно служит только для отображаемых данных.</p> <p>Работает только с целыми числовыми элементами данных.</p> <p>Например, "Windows service states".</p>						
Формат времени журнала	<p>Доступен только для элементов данных типа Журнал. Поддерживаемые значения:</p> <ul style="list-style-type: none"> * y: Год (1970-2038) * M: Месяц (01-12) * d: День (01-31) * h: Час (00-23) 						

Параметр	Описание
	<p>* m: Минута (00-59) * s: Секунда (00-59)</p> <p>Если оставить это поле пустым, то штамп времени не будет обрабатываться. Например, рассмотрим следующую строку из файла журнала Zabbix агента: “23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).” Она начинается с шести символьных позиций PID, далее дата, время и остальная часть строки.</p> <p>Формат времени журнала для этой строки должен быть “rrrrrr:ууууММдд:hhmmss”.</p> <p>Обратите внимание, что символы “р” и “:” являются лишь заменителями и могут быть какими угодно, кроме “уMdhms”.</p>
<i>Новая группа элементов данных</i>	Введите имя новой группы элементов данных для этого элемента данных.
<i>Группы элементов данных</i>	Соединение элемента данных с одним или несколькими существующими группами элементов данных.
<i>Заполнение поля инвентаря узла сети</i>	Вы можете выбрать поле инвентарных данных, которое будет заполняться значением элемента данных. Функция будет работать, если у узла сети включено автоматическое заполнение инвентарных данных .
<i>Описание</i>	Введите описание элемента данных.
<i>Активировано</i>	Отметьте для активации элемента данных, таким образом он будет обрабатываться. При редактировании элемента данных уровня шаблона на уровне узла сети, некоторое количество полей доступно только для чтения. Вы можете воспользоваться ссылкой из заголовка диалога и перейти на уровень шаблона и отредактировать элемент данных там, однако имейте в виду, что изменение элемента данных коснется всех узлов сети к которым присоединен шаблон.

Предобработка значений элемента данных

Вкладка **Предобработка** позволяет задать правила преобразования полученных значений. Можно использовать одно или несколько правил предобработки до сохранения значений в базу данных. Преобразования выполняются в том порядке, в котором они были добавлены. Вся предобработка выполняется Zabbix сервером.

Смотрите также: [Детальная информация по предварительной обработке](#)



В параметрах предварительной обработки значений элементов данных поддерживаются [пользовательские макросы](#) и пользовательские макросы с контекстом.

Преобразование

Описание

Совпадение значения с регулярным выражением <шаблона> и замена значения в соответствии с <выводом>. Регулярное выражение поддерживает извлечение до 10 захватенных групп в \N последовательности. Элемент данных станет неподдерживаемым в случае ошибки при поиске соответствия во входящем значении.

Параметры:

Регулярное выражение

шаблон - регулярное выражение

вывод - шаблон форматирования вывода. \N (где N=1..9) - управляющая последовательность заменяется N-нной совпадающей группой.

Управляющая последовательность \0 заменяется совпадающим текстом. Поддерживается начиная с 3.4.0.

Пожалуйста, обратитесь в разделу [регулярных выражений](#) для ознакомления с некоторыми существующими примерами.

Обрезка

Удаление указанных символов с начала или конца значения.

Обрезка справа

Удаление указанных символов с конца значения.

Обрезка слева

Удаление указанных символов с начала значения.

XML XPath

Извлечение значения или фрагмента из XML данных с использованием XPath функционала.

Чтобы эта опция заработала, Zabbix сервер необходимо скомпилировать с поддержкой libxml.

Примеры:

```
number(/document/item/value) извлечёт 10 из
<document><item><value>10</value></item></document>
number(/document/item/@attribute) извлечёт 10 из
<document><item attribute="10"></item></document>
/document/item извлечёт <item><value>10</value></item> из
<document><item><value>10</value></item></document>
```

Преобразование

Описание

JSON Path

Пользовательский множитель

Простое изменение

Изменение в секунду

Обратите внимание на то, что пространства имен не поддерживаются. Поддерживается начиная с 3.4.0.

Извлечение значения или фрагмента с JSON данных с использованием простейшего набора функций JSONPath функционала.

Примеры:

```
$ .document.item.value извлечёт 10 с {"document": {"item": {"value": 10}}}
```

```
$ .document.item извлечёт {"value": 10} с {"document": {"item": {"value": 10}}}
```

```
$ ['a document'].item.value извлечёт 10 с {"a document": {"item": {"value": 10}}}
```

```
$ .document.items[1].value извлечёт 20 с {"document": {"items": [{"value": 10}, {"value": 20}]}}
```

Обратите внимание, что поддерживаются только прямые пути к одиночным объектам в обозначении точкой или квадратной скобкой.

В JSONPath обозначении точкой (`$.a.b.c`) можно использовать только буквенно-цифровые символы + символ подчёркивания . Если имя объекта JSON содержит другие символы, необходимо использовать обозначение квадратной скобкой (`$['a']['b']['c']`). Допускается смешивание обозначений (`$.a['b'].c`)

Извлечение нескольких значений не поддерживается.

Поддерживается начиная с 3.4.0.

Умножение значения на заданное целое или дробное число.

Использование этой опции сконвертирует полученные значения в KB, MBps, и т.д. в B, Bps. В противном случае Zabbix не сможет корректно подставить префиксы (K, M, G и т.д.).

Начиная с Zabbix 2.2, также поддерживается научное представление.

Например, `1e+70`.

Вычисление разницы между текущим и предыдущим значениями.

Вычисляется **значение-пред_значение**, где

значение - текущее значение; **пред_значение** - ранее полученное значение

Эта опция может быть полезной для измерения постоянно растущего значения. Если текущее значение меньше предыдущего, тогда Zabbix отбрасывает это изменение (ничего не записывает) и ждет следующего значения.

На один элемент данных разрешена только одна операция изменения.

Вычисление изменения значения (разница между текущим и предыдущим значениями) скорость в секунду.

Вычисляется **(значение-пред_значение)/(время-пред_время)**, где

значение - текущее значение; **пред_значение** - ранее полученное значение; **Время** - текущий штамп времени; **пред_значение** - штамп времени предыдущего значения.

Эта настройка крайне полезна при получении скорости в секунду в случае постоянно увеличивающихся значений. Если текущее значение меньше чем предыдущее значение, Zabbix отбрасывает эту разницу (ничего не записывает) и ждет следующее значение. Такое поведение помогает отрабатывать, например, переполнение 32-битных счетчиков SNMP.

Обратите внимание: Так как результатом этого вычисления может быть число с плавающей точкой, рекомендуется использовать 'Тип информации' - Числовой (с плавающей точкой), даже если входящие значения целые положительные числа. Это особенно актуально для маленьких чисел где дробная часть имеет значение. Если числа с плавающей точкой очень большие и могут превысить длину поля в этом случае введенное значение может быть потеряно, в таком случае предлагается использовать тип

Преобразование

Описание

Двоичное в десятичное

Числовой (*целое положительное*) и это отбросит только дробную часть. На один элемент данных разрешена только одна операция изменения. Конвертация значения из логического в десятичный формат. Текстовое представление переводится в значения 0 или 1. Таким образом, 'TRUE' хранится как 1 и 'FALSE' хранится как 0. Все значения сопоставляются чувствительным к регистру способом. В данное время распознаются значения для:

TRUE - true, t, yes, y, on, up, running, enabled, available, ok, master

FALSE - false, f, no, n, off, down, unused, disabled, unavailable, err, slave

Кроме того, любое ненулевое значение считается TRUE, нулевое считается FALSE.

Следующие значения поддерживаются начиная с 4.0.0: ok, master, err, slave.

Восьмеричное в десятичное

Конвертация значения из восьмеричного в десятичный формат.

Шестнадцатеричное в десятичное

Конвертация значения из шестнадцатеричного в десятичный формат.

Если вы используете пользовательский множитель или хранение значений как *Изменение в секунду* у элементов данных с типом информации Числовой (*целое положительное*) и результирующее вычисленное значение в действительности дробное число, тогда вычисленное значение всё же будет считаться приемлемым, однако дробная часть будет обрезана и значение запишется как целое положительное.

Кнопки диалога

Кнопки в нижней части диалога позволяют выполнить несколько видов операций.

Add

Добавление элемента данных. Эта кнопка доступна только для новых элементов данных.

Update

Обновление свойств элемента данных.

Clone

Создание другого элемента данных на основе свойств текущего элемента данных.

Check now

Выполнение немедленной проверки нового значения элемента данных.

Поддерживается только для **пассивных** проверок (смотрите более [детальную информацию](#)).

Обратите внимание, что когда выполняется немедленная проверка значения, кэш конфигурации не обновляется, поэтому значение не будет отражать совсем недавние изменения настроек элемента данных.

Clear history and trends

Удаление истории и динамики изменений элемента данных.

Delete

Удаление элемента данных.

Cancel

Отмена изменения свойств элемента данных.

Ограничения текстовых данных

Ограничения текстовых данных зависят от выбранной базы данных. Перед сохранением текстовых значений в базу данных, эти значения обрезаются в соответствии с ограничением типа значений в базе данных:

База данных	Тип информации		
	Символ	Журнал (лог)	Текст
MySQL	255 символов	65536 байт	65536 байт
PostgreSQL	255 символов	65536 символов	65536 символов
Oracle	255 символов	65536 символов	65536 символов

База данных	Символ	Тип информации	
DB2	255 байт	Журнал (лог)	Текст

Чёрный список единиц измерения

По умолчанию, указывая единицу измерения элементу данных, вы увидите добавленный префикс множителя - например, значение '2048' с единицей измерения 'B' будет отображаться как '2KB'.

Однако, используя ! префикс, можно избежать конвертации любой единицы измерения, например !B. Чтобы лучше проиллюстрировать то, каким образом это преобразование работает с и без чёрного списка, просмотрите следующие примеры значений и единиц измерения:

```
1024 !B -> 1024 B
1024 B -> 1 KB
61 !s -> 61 s
61 s -> 1m 1s
0 !uptime -> 0 uptime
0 uptime -> 00:00:00
0 !! -> 0 !
0 ! -> 0
```

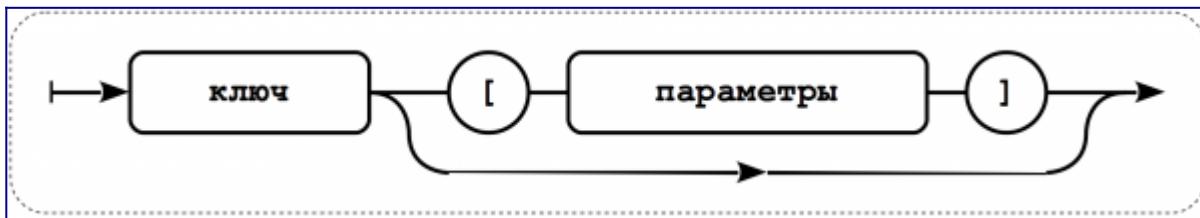
До Zabbix 4.0, имелся зашитый в код чёрный список единиц измерения, который состоял из ms, rpm, RPM, %. Этот чёрный список считается устаревшим, поэтому правильным способом внесения в чёрный список подобных единиц измерения будет !ms, !rpm, !RPM, !%.

Неподдерживаемые элементы данных

Элемент данных может стать неподдерживаемым, если его значение не было получено по некоторым причинам. Такие элементы данных будут перепроверены через определенный интервал времени, настраиваемый в [разделе Администрирования](#).

7.2.1.1 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 следя линиям - если в каком то блоке может быть достигнута линия, в таком случае разрешено, если линия не может быть достигнута - тогда не разрешено.



Для построения правильного ключа элемента данных, вы должны начать с указания имени ключа, то есть выбрать имеет ли ключ параметры или нет - о чем свидетельствуют две линии, которые вы можете достичь.

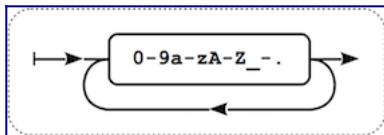
Имя ключа

Имя ключа имеет ограниченный диапазон разрешенных символов, которые просто следуют друг за другом. Разрешенные символы:

0-9a-zA-Z_- .

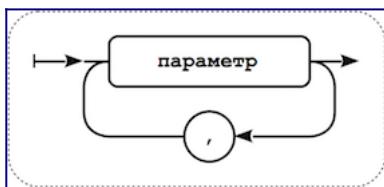
Что означает:

- все цифры;
- все буквы в нижнем регистре;
- все буквы в верхнем регистре;
- подчеркивание;
- дефис;
- точка.



Параметры ключа

Ключ элемента данных может принимать множество параметров, которые должны быть разделены запятой.



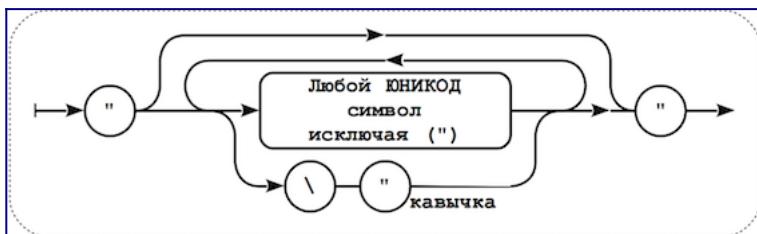
Каждый параметр ключа может быть одним из: строка заключенная в кавычки, строка без кавычек, массив.



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 заключенная в кавычки

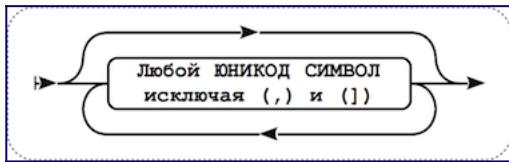
Если параметр ключа это строка, заключенная в кавычки, тогда разрешен любой символ в Юникоде, и если требуется включение двойных кавычек, то они должны быть экранированы обратной наклонной чертой.



Чтобы заключить параметры ключа элемента данных в кавычки, используйте только двойные кавычки. Одинарные кавычки не поддерживаются.

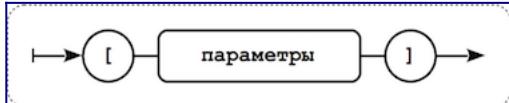
Параметр - строка не заключенная в кавычки

Если параметр ключа это строка без кавычек, тогда разрешен любой символ в Юникоде, за исключением запятой и правой квадратной скобки (]). Параметр, который не заключен в кавычки, не может начинаться с левой квадратной скобки ([).



Параметр - массив

Если параметр ключа это массив, тогда он должен быть заключен в квадратные скобки, в которых каждый индивидуальный параметр следует один за другим, согласно правилам и синтаксису.



Многоуровневые массивы параметров, например [a, [b, [c, d]], e], недопускаются.

7.2.1.2 Пользовательские интервалы

Обзор

Имеется возможность создания пользовательских правил относительно времени, когда элемент данных будет опрашиваться. Для этого есть два способа, *Переменные интервалы*, который позволяет переопределить интервал обновления по умолчанию, и *По расписанию*, посредством чего элемент данных может быть опрошен в конкретное время или последовательность времени.

Гибкие интервалы

Гибкие интервалы позволяют переопределить интервал обновления по умолчанию указанными периодами времени. Гибкий интервал указывается с *Интервал* и *Период*, где:

- *Интервал* – интервал обновления в указанный период времени
- *Период* – период времени, когда гибкий интервал активен (смотрите [периоды времени](#) для подробного описания формата *Период*)

Возможно указать до семи переменных интервалов. Если несколько гибких интервалов перекрываются, то используется *Интервал* с наименьшим значением для перекрывающегося периода. Обратите внимание, что если наименьший интервал из перекрывающих имеет значение "0", то проверка не будет произведена вообще. В остальных случаях будет использоваться интервал обновления по умолчанию.

Обратите внимание что, если гибкий интервал совпадает с длительностью периода, элемент данных будет опрошен только один раз. Если гибкий интервал больше чем период, элемент данных может быть опрошен один раз или может быть не проверен вовсе (поэтому такая настройка не рекомендуется). Если гибкий интервал меньше чем период, элемент данных будет опрошен по крайней мере один раз.

Если гибкий интервал задан равным '0', элемент данных не будет обрабатываться в течении периода гибкого интервала и опрос возобновится в соответствии с *Интервал обновления* по умолчанию сразу как только период завершится. Примеры:

Интервал	Период	Описание
10	1-5,09:00-18:00	Элемент данных будет опрашиваться каждые 10 секунд в течении рабочего времени.
0	1-7,00:00-7:00	Элемент данных не будет опрашиваться в течении ночи.
0	7-7,00:00-24:00	Элемент данных не будет опрашиваться по Воскресеньям.
60	1-7,12:00-12:01	Элемент данных будет опрошен ровно в 12:00 каждый день. Обратите внимание, что такой формат использовался как обходной путь при опросе по расписанию и начиная с Zabbix 3.0 рекомендуется использовать интервалы по расписанию для таких проверок.

Интервалы по расписанию

Интервалы по расписанию используются для опроса элементов данных в заданное время. Тогда как гибкие 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<фильтр>wd<фильтр>h<фильтр>m<фильтр>s<фильтр> where:

- **md** - дни месяца
- **wd** - дни недели
- **h** - часы
- **m** - минуты
- **s** – секунды

<фильтр> используется для того, чтобы указать значения по своему префиксу (дни, часы, минуты, секунды) и задается как: [<от>[-<до>]] [/<шаг>] [, <фильтр>] где:

- <от> и <до> определяют диапазон соответствующих значений (включительно). Если <до> не указывается, тогда фильтр совпадает с диапазоном <от> - <от>. Если <от> также не указан, тогда фильтр совпадает со всеми возможными значениями.
- <шаг> определяет пропуски количества значений через диапазон. По умолчанию значение <шага> 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 определен, либо, в противном случае, все возможные значения. Например, если фильтр часа опущен, тогда будет соответствовать только '0', если фильтры минут и секунд также опущены, в противном случае пустой фильтр часа будет соответствовать всем значениям часа.

Допустимые значения <от> и <до> 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 <от> должно быть меньше или равно значению <до>. Значение <шага> должно быть больше или равно 1 и меньше или равно <от> - <до>.

К одиночным 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 точкой с запятой ;.

Если время совпадает по нескольким интервалам, опрос выполнится только раз. Например, `wd1h9;h9` выполнится только один раз в Понедельник в 9 утра.

Примеры:

Интервал	Когда выполнится
<code>m0-59</code>	каждую минуту
<code>h9-17/2</code>	каждые 2 часа, начиная с 9:00 (9:00, 11:00 ...)
<code>m0,30 или m/30</code>	каждый час в чч:00 и чч:30
<code>m0,5,10,15,20,25,30,35,40,45,50,55 или m/5</code>	каждые пять минут
<code>wd1-5h9</code>	каждый Понедельник до Пятницы в 9:00
<code>wd1-5h9-18</code>	каждый Понедельник до Пятницы в 9:00,10:00,...,18:00
<code>h9,10,11 или h9-11</code>	каждый день в 9:00, 10:00 и 11:00
<code>md1h9m30</code>	каждый 1ый день каждого месяца в 9:30
<code>md1wd1h9m30</code>	каждый 1ый день каждого месяца в 9:30, если этот день Понедельник
<code>h9m/30</code>	каждый день в 9:00, 9:30
<code>h9m0-59/30</code>	каждый день в 9:00, 9:30
<code>h9,10m/30</code>	каждый день в 9:00, 9:30, 10:00, 10:30
<code>h9-10m30</code>	каждый день в 9:30, 10:30
<code>h9m10-40/30</code>	каждый день в 9:10, 9:40
<code>h9,10m10-40/30</code>	каждый день в 9:10, 9:40, 10:10, 10:40
<code>h9-10m10-40/30</code>	каждый день в 9:10, 9:40, 10:10, 10:40
<code>h9m10-40</code>	каждый день в 9:10, 9:11, 9:12, ... 9:40
<code>h9m10-40/1</code>	каждый день в 9:10, 9:11, 9:12, ... 9:40
<code>h9-12,15</code>	каждый день в 9:00, 10:00, 11:00, 12:00, 15:00
<code>h9-12,15m0</code>	каждый день в 9:00, 10:00, 11:00, 12:00, 15:00
<code>h9-12,15m0s30</code>	каждый день в 9:00:30, 10:00:30, 11:00:30, 12:00:30, 15:00:30
<code>h9-12s30</code>	каждый день в 9:00:30, 9:01:30, 9:02:30 ... 12:58:30, 12:59:30
<code>h9m/30;h10</code>	каждый день в 9:00, 9:30, 10:00

7.2.2 Типы элементов данных

Обзор

Типы элементов данных охватывают различные методы получения данных с вашей системы. Каждый тип элемента данных поставляется со своим собственным набором поддерживаемых ключей элементов данных и требуемых параметров.

В настоящее время Zabbix предлагает следующие типы элементов данных:

- [Zabbix агент проверки](#)
- [SNMP агент проверки](#)
- [SNMP тралы](#)

- [IPMI проверки](#)
- [Простые проверки](#)
 - [VMware мониторинг](#)
- [Мониторинг файлов журналов](#)
- [Вычисляемые элементы данных](#)
- [Внутренние проверки Zabbix](#)
- [SSH проверки](#)
- [Telnet проверки](#)
- [Внешние проверки](#)
- [Агрегированные проверки](#)
- [Траппер элементы данных](#)
- [JMX мониторинг](#)
- [ODBC проверки](#)
- [Зависимые элементы данных](#)
- [HTTP проверки](#)

Детальная информация по всем типам элементов данных включена в подстраницы этого раздела. Несмотря на это, некоторые типы элементов данных предлагают большее количество опций по сбору данных, дополнительные опции через [пользовательские параметры](#) или [подгружаемые модули](#).

Некоторые проверки выполняются Zabbix сервером в одиночку (так называемый безагентный мониторинг) в то время как остальные требуют Zabbix агента или даже Zabbix Java gateway (с JMX мониторингом).

Если отдельный тип элемента данных требует определенный интерфейс (например, IPMI проверка требует IPMI интерфейс на узле сети), то этот интерфейс должен существовать в определении этого узла сети.

Можно задвать несколько интерфейсов в определении узла сети: Zabbix агент, SNMP агент, JMX и IPMI. Если элемент данных может использовать более чем один интерфейс, он будет искать доступные интерфейсы у узла сети (в следующем порядке: Агент → SNMP → JMX → IPMI) и будет связан с первым подходящим ему интерфейсом.

Все элементы данных которые возвращают текст (символ, журнал, текстовый типы информации) теперь могут возвращать только пробелы (при необходимости), при этом возвращенное значение становится пустой строкой (поддерживается начиная с 2.0).

7.2.2.1 Zabbix агент

Обзор

Эти проверки используют общение с Zabbix агентом для сбора данных.

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

Поддерживаемые ключи элементов данных

В таблице приводится подробная информация о ключах элементов данных, которые вы можете использовать элементами данных Zabbix агента.

Смотрите также:

- [Элементы данных поддерживаемые по платформам](#)
- [Специфичные ключи элементов данных для Windows агента](#)

Обязательные и необязательные параметры

Параметры без угловых скобок обязательны. Параметры, отмеченные угловыми скобками < >, необязательны.

Описание	Возвращаемое значение	Ключ	Параметры	Комментарии
Имя хоста агента.	Строка	agent.hostname		Возвращает действительное значение hostname агента из файла конфигурации.
Проверка доступности агента.	Ничего - недоступен 1 - доступен	agent.ping		Используйте nodata() функцию триггера для проверки недоступности узла сети.
Версия Zabbix агента.	Строка	agent.version		Пример возвращаемого значения: 1.8.2
Максимальное количество открытых файлов поддерживающее ОС.	Целое число	kernel.maxfiles		
Максимальное количество процессов поддерживающее ОС.	Целое число	kernel.maxproc		
Мониторинг файлов журналов (логов).	Журнал (лог)	log[файл,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<вывод>,<максзадержка>]	файл - абсолютный путь и имя файла журнала регулярное выражение - регулярное выражение описывающее требуемый шаблон содержимого кодировка - идентификатор кодовой страницы макс. кол-во строк - максимальное количество новых строк в секунду,	Элемент данных должен быть настроен активной проверкой . Если файл не существует или доступ к нему не разрешен, элемент данных переходит в состояние неподдерживается.
				Если параметр вывод оставить пустым - будет возвращена вся строка соответствующая регулярному выражению. Заметьте, что все глобальные регулярные выражения исключая 'Результат ИСТИНА' всегда возвращают всю строку соответствующую выражению и параметр вывод игнорируется.

Ключ

Описание	Возвращаемое значение	Параметры	Комментарии
	<p>которое агент будет отправлять Zabbix серверу или прокси. Этот параметр заменяет значение 'MaxLinesPerSecond' в <u>zabbix_agentd.conf</u></p> <p>режим - возможные значения: <i>all</i> (по умолчанию), <i>skip</i> - пропуск обработки старых данных (влияет только на недавно созданные элементы данных, которые еще не получали никаких данных).</p> <p>вывод - дополнительный шаблон форматирования вывода.</p> <p>\0 - управляющая последовательность заменяется найденным текстом, тогда</p> <p>\N (где N=1..9) - управляющая последовательность заменяется N-ной совпадающей группой (или пустой строкой, если N превышает количество найденных групп).</p> <p>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 течении</p>	<p>Извлечение содержания согласно параметра вывод производится на стороне агента.</p> <p><i>Примеры:</i> <code>⇒ log[/var/log/syslog]</code> <code>⇒ log[/var/log/syslog,error]</code> <code>⇒ log[/home/zabbix/logs logfile,,,100]</code></p> <p><i>Пример использования параметра вывод для извлечения числа из записи в журнале:</i> <code>log[/app1/app.log,"task run [0-9.]+ sec, processed ([0-9.]+) records, [0-9.]+ errors",,,,\1]</code> → будет соответствовать записи в журнале "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" и на сервер отправится только число 6080. Так как отправляется число, у этого элемента данных журнала можно изменить "Тип информации" с "Журнал (лог)" на "Числовой (целое положительное)" и это значение можно будет использовать на на графиках, в триггерах и т.д.</p> <p><i>Пример использования параметра вывод для изменения вывода записи журнала перед отправкой на сервер:</i> <code>log[/app1/app.log,"([0-9 :]+) task run ([0-9.]+) sec, processed ([0-9.]+) records, ([0-9.]+) errors",,,,"\\1 RECORDS: \\3, ERRORS: \\4, DURATION: \\2"]</code> → будет соответствовать записи в журнале "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" и на сервер отправится измененная запись "2015-11-13 10:08:26 RECORDS: 6080, ERRORS: 0, DURATION: 6.08".</p> <p>Параметр режим поддерживается начиная с Zabbix 2.0. Параметр вывод поддерживается начиная с Zabbix 2.2. Параметр максзадержка поддерживается начиная с Zabbix 3.2.</p>	

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
Подсчёт количества совпадающих строк в мониторинге файла журнала (логов).	Целое число	<p>“максзадержка” секунд. Перед использованием прочтайте заметки по максзадержка!</p> <p>log.count[файл,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<максзадержка>]</p> <p>файл - абсолютный путь и имя файла журнала</p> <p>регулярное выражение - регулярное выражение описывающее требуемый шаблон содержимого</p> <p>кодировка - идентификатор кодовой страницы</p> <p>макс. кол-во строк - максимальное количество новых строк в секунду, которое агент будет анализировать.</p> <p>Значением по умолчанию является $10 * \text{MaxLinesPerSecond}$</p> <p>d' в zabbix_agentd.conf.</p> <p>режим - возможные значения: <i>all</i> (по умолчанию), <i>skip</i> - пропуск обработки старых данных (влияет только на недавно созданные элементы данных).</p> <p>максзадержка - максимальная задержка в секундах.</p> <p>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</p>	<p>Смотрите дополнительную информацию о мониторинге файлов журналов.</p> <p>Элемент данных должен быть настроен активной проверкой. Если файл не существует или доступ к нему не разрешен, элемент данных переходит в состояние неподдерживается.</p> <p>Также смотрите дополнительную информацию о мониторинге файлов журналов.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>

Описание	Возвращаемое значение	Параметры	Комментарии
Мониторинг файлов журналов (логов) с поддержкой ротации логов.	Журнал (лог)	<p>наиболее новых строк проанализированных в течении “максзадержка” секунд. Перед использованием прочтайте заметки по максзадержка!</p> <p>logrt[файл_regexp,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<вывод>,<максзадержка>,<опции>]</p> <p>файл_regexp - абсолютный путь к файлу и регулярное выражение описывающее его имя</p> <p>регулярное выражение - регулярное выражение описывающее требуемый шаблон содержимого</p> <p>кодировка - идентификатор кодовой страницы</p> <p>макс. кол-во строк - максимальное количество новых строк в секунду, которое агент будет отправлять Zabbix серверу или прокси. Этот параметр заменяет значение 'MaxLinesPerSecond' в zabbix_agentd.conf</p> <p>режим - возможные значения: <i>all</i> (по умолчанию), <i>skip</i> - пропуск обработки старых данных (влияет только на недавно созданные элементы данных, которые еще не получали никаких данных).</p> <p>вывод - дополнительный шаблон форматирования вывода:</p>	<p>Элемент данных должен быть настроен активной проверкой. Ротация журналов основывается на времени последнего изменения файлов.</p> <p>Если параметр вывод оставить пустым - будет возвращена вся строка соответствующая регулярному выражению. Заметьте, что все глобальные регулярные выражения исключая 'Результат ИСТИНА' всегда возвращают всю строку соответствующую выражению и параметр вывод игнорируется.</p> <p>Извлечение содержимого согласно параметра вывод производится на стороне агента.</p> <p>Примеры:</p> <p>⇒ logrt["/home/zabbix/logs/^logfile[0-9]{1,3}\$","","100"] → совпадает с файлом наподобии "logfile1" (но не совпадает с ".logfile1")</p> <p>⇒ logrt["/home/user/logfile_*_[0-9]{1,3}","pattern_to_match","UTF-8",100] → будет собирать данные с файлов таких как "logfile_abc_1" или "logfile__001".</p> <p><i>Пример использования параметра вывод для извлечения числа из записи в журнале:</i></p> <p>logrt[/app1/^test.*log\$,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors","","1"] → будет соответствовать записи в журнале "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" и на сервер отправится только число 6080. Так</p>

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
		<p>\0 - управляющая последовательность заменяется найденным текстом, тогда</p> <p>\N (где N=1..9) - управляющая последовательность заменяется N-ной совпадающей группой (или пустой строкой, если N превышает количество найденных групп).</p> <p>максзадержка - максимальная задержка в секундах.</p> <p>Тип: число с плавающей точкой.</p> <p>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 течении "максзадержка" секунд. Перед использованием прочтайте заметки по максзадержка!</p> <p>опции - тип ротации файлов журналов.</p> <p>Возможные значения: <i>rotate</i> (по умолчанию), <i>copytruncate</i>.</p> <p>Обратите внимание, что <i>copytruncate</i> нельзя использовать вместе с <i>maxdelay</i>. В этом случае <i>maxdelay</i> должен быть равен 0 или не задан.</p> <p>Смотрите заметки по copytruncate.</p>	<p>как отправляется число, у этого элемента данных журнала можно изменить "Тип информации" с "Журнал (лог)" на "Числовой (целое положительное)" и это значение можно будет использовать на на графиках, в триггерах и т.д.</p> <p><i>Пример использования параметра вывод для изменения вывода записи журнала перед отправкой на сервер:</i></p> <pre>logrt[/app1/^test.*log\$,"([0-9 :]+) task run ([0-9.]*) sec, processed ([0-9]+) records, ([0-9]+) errors","",""1 RECORDS: \3, ERRORS: \4, DURATION: \2"] → будет соответствовать записи в журнале "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors " и на сервер отправится измененная запись "2015-11-13 10:08:26 RECORDS: 6080, ERRORS: 0, DURATION: 6.08".</pre>
Подсчёт количества	Целое число	файл_regexp -	Элемент данных должен быть

Ключ

Описание	Возвращаемое значение	Параметры	Комментарии
совпадающих строк в мониторинге файла журнала (логов) с поддержкой ротации логов.		<p>абсолютный путь к файлу и регулярное выражение описывающее его имя</p> <p>регулярное выражение - регулярное выражение описывающее требуемый шаблон содержимого</p> <p>кодировка - идентификатор кодовой страницы</p> <p>макс. кол-во строк - максимальное количество новых строк в секунду, которое агент будет анализировать.</p> <p>Значением по умолчанию является $10 * \text{MaxLinesPerSecond}$ в zabbix_agentd.conf.</p> <p>режим - возможные значения: <i>all</i> (по умолчанию), <i>skip</i> - пропуск обработки старых данных (влияет только на недавно созданные элементы данных).</p> <p>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 течении “максзадержка” секунд. Перед использованием прочтайте заметки</p>	<p>настроен активной проверкой. Ротация журналов основывается на времени последнего изменения файлов.</p> <p>Смотрите дополнительную информацию о мониторинге файлов журналов.</p> <p>Параметр ОПЦИИ поддерживается начиная с Zabbix 4.0.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>

Описание	Возвращаемое значение	Параметры	Комментарии
Проверка, работает ли DNS сервис.	<p>Возвращаемое значение</p> <p>0 - DNS не работает (сервер не ответил или ответ DNS некорректен)</p> <p>1 - DNS работает</p>	<p>по максзадержка!</p> <p>опции - тип ротации файлов журналов. Возможные значения: <i>rotate</i> (по умолчанию), <i>copytruncate</i>.</p> <p>Обратите внимание, что <i>copytruncate</i> нельзя использовать вместе с <i>maxdelay</i>. В этом случае <i>maxdelay</i> должен быть равен 0 или не задан.</p> <p>Смотрите заметки по copytruncate.</p> <p>ip - IP адрес DNS сервера (оставьте пустым, чтобы использовать DNS сервера по умолчанию, игнорируется в Windows)</p> <p>имя - проверяемое имя DNS</p> <p>тип - запрашиваемый тип записи (по умолчанию SOA)</p> <p>время ожидания (игнорируется в Windows) - время ожидания ответа в секундах (по умолчанию 1 секунда)</p> <p>количество (игнорируется в Windows) - количество попыток запросов (по умолчанию 2)</p> <p>протокол - используемый протокол при выполнении DNS запросов: <i>udp</i> (по умолчанию) или <i>tcp</i></p>	<p><i>Пример ключа:</i> ⇒ net.dns[8.8.8.8,zabbix.com,MX,2,1]</p> <p>Возможные значения для тип: ANY, A, NS, CNAME, MB, MG, MR, PTR, MD, MF, MX, SOA, NULL, WKS (исключая Windows), HINFO, MINFO, TXT, SRV</p> <p>Национальные доменные имена не поддерживаются вместо них, пожалуйста, используйте имена в IDNA кодировке.</p> <p>Параметр протокол поддерживается начиная с Zabbix 3.0. Тип записи SRV поддерживается Zabbix агентом начиная с версий 1.8.6 (Unix) и 2.0.0 (Windows).</p> <p>Название до Zabbix 2.0 (еще поддерживается): <i>net.tcp.dns</i></p>
Выполнение DNS запроса.	Возвращаемое значение	Строка символов ip - IP адрес DNS с требуемым типом	<p><i>Пример ключа:</i> ⇒</p> <p>net.dns.record[8.8.8.8,zabbix.com,MX,</p>

Описание	Возвращаемое значение	Параметры	Комментарии
Коллизии за пределами окна (out-of-window).	информации	<p>использовать DNS сервера по умолчанию, игнорируется в Windows)</p> <p>имя - проверяемое имя DNS</p> <p>тип - запрашиваемый тип записи (по умолчанию SOA)</p> <p>время ожидания (игнорируется в Windows) - время ожидания ответа в секундах (по умолчанию 1 секунда)</p> <p>количество (игнорируется в Windows) - количество попыток запросов (по умолчанию 2)</p> <p>протокол - используемый протокол при выполнении DNS запросов: <i>udp</i> (по умолчанию) или <i>tcp</i></p> <p>net.if.collisions[if]</p> <p>if - имя сетевого интерфейса</p>	<p>2,1]</p> <p>Возможные значения для тип: ANY, A, NS, CNAME, MB, MG, MR, PTR, MD, MF, MX, SOA, NULL, WKS (исключая Windows), HINFO, MINFO, TXT, SRV</p> <p>Национальные доменные имена не поддерживаются вместо них, пожалуйста, используйте имена в IDNA кодировке.</p>
Список сетевых интерфейсов. Используется низкоуровневым обнаружением.	Объект JSON	net.if.discovery	Поддерживается Zabbix агентом начиная с версии 2.0.
Статистика по входящему трафику сетевого интерфейса.	Целое число	<p>net.if.in[if,<режим>]</p> <p>if - имя сетевого интерфейса (Unix); полное описание сетевого интерфейса</p>	На FreeBSD, OpenBSD и NetBSD поддерживается начиная с версии Zabbix агента 2.2
			В некоторых версиях Windows (например, Server 2008) может потребоваться установка последних обновлений для поддержки не-ASCII символов в именах интерфейсов.

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
		<p>или IPv4 адрес (Windows)</p> <p>режим - возможные значения:</p> <ul style="list-style-type: none"> <i>bytes</i> - количество байт (по умолчанию) <i>packets</i> - количество пакетов <i>errors</i> - количество ошибок <i>dropped</i> - количество отброшенных пакетов <i>overruns (fifo)</i> - количество ошибок FIFO буфера <i>frame</i> - количество ошибок кадрирования пакетов <i>compressed</i> - количество сжатых пакетов, полученных драйвером устройства <i>multicast</i> - количество кадров многоадресного вещания, которые получены драйвером устройства 	<p>введены в Windows Vista и Windows Server 2008. Если 64-битный счетчики недоступны, агент будет использовать 32-битные счетчики.</p> <p>Начиная с Zabbix агента 1.8.6 на Windows поддерживаются мультибайтные имена интерфейса.</p> <p>Примеры: $\Rightarrow \text{net.if.in}[\text{eth0},\text{errors}]$ $\Rightarrow \text{net.if.in}[\text{eth0}]$</p> <p>Вы можете получить описания сетевых интерфейсов в Windows, используя элементы данных <code>net.if.discovery</code> или <code>net.if.list</code>.</p>
Статистика по исходящему трафику сетевого интерфейса.	Целое число	<p>net.if.out[if,<режим>]</p> <p>if - имя сетевого интерфейса (Unix); полное описание сетевого интерфейса или IPv4 адрес (Windows)</p> <p>режим - возможные значения:</p> <ul style="list-style-type: none"> <i>bytes</i> - количество байт (по умолчанию) <i>packets</i> - количество пакетов <i>errors</i> - количество ошибок <i>dropped</i> - количество отброшенных пакетов <i>overruns (fifo)</i> - количество ошибок FIFO буфера <i>collisions (colls)</i> - количество коллизий обнаруженных на интерфейсе 	<p>Вы можете использовать этот ключ с шагом предобработки <i>Изменение в секунду</i> для получения статистики байт в секунду.</p> <p>В Windows элемент данных берет значения с 64-битный счетчиками, если они доступны. 64-битные счетчики статистики по интерфейсам введены в Windows Vista и Windows Server 2008. Если 64-битный счетчики недоступны, агент будет использовать 32-битные счетчики.</p> <p>Начиная с Zabbix агента 1.8.6 на Windows поддерживаются мультибайтные имена интерфейса.</p> <p>Примеры: $\Rightarrow \text{net.if.out}[\text{eth0},\text{errors}]$ $\Rightarrow \text{net.if.out}[\text{eth0}]$</p> <p>Вы можете получить описания сетевых интерфейсов в Windows, используя элементы данных <code>net.if.discovery</code> или <code>net.if.list</code>.</p> <p>Вы можете использовать этот ключ с</p>

Описание	Возвращаемое значение	Параметры	Комментарии
Суммарная статистика входящего и исходящего трафика на сетевом интерфейсе.	Целое число	<p><i>carrier</i> - количество потерь несущей, обнаруженных драйвером устройства <i>compressed</i> - количество сжатых пакетов, переданных драйвером устройства</p> <p>net.if.total[if,<режим>]</p> <p>if - имя сетевого интерфейса (Unix); полное описание сетевого интерфейса или IPv4 адрес (Windows) режим - возможные значения: <i>bytes</i> - количество байт (по умолчанию) <i>packets</i> - количество пакетов <i>errors</i> - количество ошибок <i>dropped</i> - количество отброшенных пакетов <i>overruns (fifo)</i> - количество ошибок FIFO буфера <i>compressed</i> - количество сжатых пакетов, переданных или полученных драйвером устройства</p> <p>net.tcp.listen[порт]</p> <p>порт - номер TCP порта</p>	<p>шагом предобработки <i>Изменение в секунду</i> для получения статистики байт в секунду.</p> <p>В Windows элемент данных берет значения с 64-битный счетчиков, если они доступны. 64-битные счетчики статистики по интерфейсам введены в Windows Vista и Windows Server 2008. Если 64-битный счетчики недоступны, агент будет использовать 32-битные счетчики.</p> <p>Примеры: ⇒ net.if.total[eth0,errors] ⇒ net.if.total[eth0]</p> <p>Вы можете получить описания сетевых интерфейсов в Windows, используя элементы данных net.if.discovery или net.if.list.</p> <p>Вы можете использовать этот ключ с шагом предобработки <i>Изменение в секунду</i> для получения статистики байт в секунду.</p> <p>Обратите внимание, отброшенные пакеты поддерживаются только, если оба ключа net.if.in и net.if.out работают с отброшенными пакетами на вашей платформе.</p> <p>Пример: ⇒ net.tcp.listen[80]</p> <p>В Linux поддерживается Zabbix агентом начиная с версии 1.8.4</p> <p>Начиная с Zabbix 3.0.0, в ядрах Linux 2.6.14 и более новых, информация о TCP сокетах в состоянии listen берется из интерфейса ядра NETLINK, если возможно. В противном случае, информация берется из файлов /proc/net/tcp и</p>
Проверка, находится ли TCP порт в состоянии LISTEN.	0 - не находится в состоянии LISTEN. 1 - находится в состоянии LISTEN		

Описание	Возвращаемое значение	Параметры	Ключ Комментарии
Проверка возможности создания TCP подключения на указанный номер порта.	0 - не удалось подключиться 1 - удалось подключиться	ip - IP адрес (по умолчанию 127.0.0.1) порт - номер порта	/proc/net/tcp6. net.tcp.port[<ip>,порт] Пример: ⇒ net.tcp.port[,80] → можно использовать для проверки доступности веб-сервера, работающего на 80 порту. Для простого тестирования производительности TCP используйте net.tcp.service.perf[tcp,<ip>,<порт>]
Проверка, запущен ли сервис и принимает ли он TCP подключения.	0 - сервис недоступен 1 - сервис запущен	сервис - один из: <i>ssh, ntp, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet</i> (смотри детали) ип - IP адрес (по умолчанию 127.0.0.1) порт - номер порта (по умолчанию используется стандартный номер порта сервиса)	Обратите внимание, что эти проверки могут привести к дополнительным записям в системных файлах журналов (обычно сессии SMTP и SSH журналируются). Старое название: check_port[*] net.tcp.service[сервис,<ип>,<порт>] Пример: ⇒ net.tcp.service[ftp,,45] - можно использовать для тестирования доступности FTP сервера на TCP 45 порту. Обратите внимание, что эти проверки могут привести к дополнительным записям в системных файлах журналов (обычно сессии SMTP и SSH журналируются). Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. Как решение, пожалуйста, для подобных проверок используйте net.tcp.port . Проверка LDAP и HTTPS через Windows агента в настоящее время не поддерживается. Обратите внимание, что telnet проверка ищет запрос на вход (с ':' в конце).

Описание	Возвращаемое значение	Параметры	Ключ
Проверка производительности TCP сервиса.	0 - сервис недоступен секунды - количество секунд потраченное на подключение к сервису	сервис - один из: <i>ssh, ntp, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet</i> (смотри детали) ip - IP адрес (по умолчанию 127.0.0.1) порт - номер порта (по умолчанию используется стандартный номер порта сервиса)	по проверке HTTPS сервиса. Сервисы <i>https</i> и <i>telnet</i> поддерживаются Zabbix с версии 2.0. Старое название: <code>check_service[*]</code> net.tcp.service.perf[сервис,<ip>,<порт>] Пример: ⇒ <code>net.tcp.service.perf[ssh]</code> - можно использовать для тестирования скорости изначального ответа от SSH сервера.
Проверка, находится ли UDP порт в состоянии LISTEN.	0 - не находится в состоянии LISTEN 1 - находится в состоянии LISTEN	порт - номер UDP порта	Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. Как решение, пожалуйста, для подобных проверок используйте <code>net.tcp.service.perf[tcp,<ip>,<порт>]</code> . Проверка LDAP и HTTPS через Windows агента в настоящее время не поддерживается. Обратите внимание, что telnet проверка ищет запрос на вход (с ':' в конце). Смотрите также известные проблемы по проверке HTTPS сервиса.
Проверка, запущен ли сервис и принимает ли он UDP подключения.	0 - сервис недоступен 1 - сервис	сервис - <i>ntp</i> (смотри детали) ип - IP адрес (по умолчанию 127.0.0.1)	Сервисы <i>https</i> и <i>telnet</i> поддерживаются Zabbix с версии 2.0. Старое название: <code>check_service_perf[*]</code> net.udp.listen[порт] Например: ⇒ <code>net.udp.listen[68]</code> В Linux поддерживается Zabbix агентом начиная с версии 1.8.4
		net.udp.service[сервис,<ип>,<порт>]	Пример: ⇒ <code>net.udp.service[ntp,,45]</code> → можно использовать для тестирования доступности NTP сервиса на 45

Описание	Возвращаемое значение	Параметры	Комментарии
	запущен	порт - номер порта (по умолчанию используется стандартный номер порта сервиса)	порту UDP.
Проверка производительности UDP сервиса.	0 - сервис недоступен секунды - количество секунд потраченное на подключение к сервису	сервис - <i>ntp</i> (смотри детали) ip - IP адрес (по умолчанию 127.0.0.1) порт - номер порта (по умолчанию используется стандартный номер порта сервиса)	Этот элемент данных поддерживается начиная с Zabbix 3.0.0, но <i>ntp</i> сервис был доступен в <code>net.tcp.service[]</code> элементе данных и в предыдущих версиях. net.udp.service.perf[сервис,<ip>,<порт>] Пример: ⇒ <code>net.udp.service.perf[ntp]</code> → можно использовать для тестирования времени ответа от NTP сервиса.
Использование CPU процесса в процентах.	Число с плавающей точкой	имя - имя процесса (по умолчанию <i>все процессы</i>) пользователь - имя пользователя (по умолчанию <i>все пользователи</i>) тип - тип использования CPU: <i>total</i> (по умолчанию), <i>user</i> , <i>system</i> cmdline - фильтр по командной строке (является регулярным выражением) режим - режим сбора данных: <i>avg1</i> (по умолчанию), <i>avg5</i> , <i>avg15</i> зона - целевая зона: <i>current</i> (по умолчанию), <i>all</i> . Этот параметр поддерживается только на Solaris платформе.	 proc.cpu.util[<имя>,<пользователь>,<тип>,<cmdline>,<режим>,<зона>] Примеры: ⇒ <code>proc.cpu.util[,root]</code> → Использование CPU по всем процессам работающими под “root” пользователем ⇒ <code>proc.cpu.util[zabbix_server,zabbix]</code> → Использование CPU по всем процессам zabbix_server работающими под пользователем zabbix Возвращаемое значение основывается на использовании одного ядра CPU, в процентах. Например, утилизация CPU процесса полного использования двух ядер равна 200%.

Описание	Возвращаемое значение	Параметры	Ключ	Комментарии
Количество памяти используемое процессом в байтах.	Целое число	<p>имя - имя процесса (по умолчанию <i>все процессы</i>) пользователь - имя пользователя (по умолчанию <i>все пользователи</i>) режим - возможные значения: <i>avg, max, min, sum</i> (по умолчанию) cmdline - фильтр по командной строке (является регулярным выражением) тип памяти - <u>тип памяти</u> используемый процессом.</p>	<p>на более новом Solaris, где зоны поддерживаются, тогда агент вернёт NOTSUPPORTED (агент не может ограничить результаты в пределах только текущей зоны). Однако, значение <i>all</i> в этом случае поддерживается.</p> <p>Этот ключ поддерживается начиная с Zabbix 3.0.0 и также доступен на нескольких платформах (смотри Поддерживаемые элементы данных по платформам).</p> <p>proc.mem[<имя>,<пользователь>,<режим>,<cmdline>,<тип памяти>]</p> <p>Примеры:</p> <ul style="list-style-type: none"> ⇒ proc.mem[,root] - память используемая всеми процессами запущенными под пользователем “root” ⇒ proc.mem[zabbix_server,zabbix] - память используемая всеми процессами zabbix_server запущенными под пользователем zabbix ⇒ proc.mem[,oracle,max,oracleZABBIX] - максимальное значение используемой памяти процессами запущенными под пользователем oracle и имеющими oracleZABBIX в содержимом командной строки <p><i>Обратите внимание:</i> Когда несколько процессов используют разделяемую память, сумма памяти используемой процессами в результате может быть большой, нереальной величиной.</p> <p>Смотрите заметки по выбору процессов с параметрами имя и cmdline (специфика для Linux).</p> <p>Когда этот элемент данных вызывается с командной строки и содержит параметр командной строки (например, при использовании тестового режима агента: <code>zabbix_agentd -t proc.num[,, apache2]</code>), будет засчитан один дополнительный процесс, так как агент посчитает</p>	

Описание	Возвращаемое значение	Параметры	Ключ
Количество процессов.	Целое число	<p>имя - имя процесса (по умолчанию “все процессы”) пользователь - имя пользователя (по умолчанию “все пользователи”) состояние - возможные значения: <i>all</i> (по умолчанию), <i>disk</i> - непрерывный сон, <i>run</i> - в процессе работы, <i>sleep</i> - прерываемый сон, <i>trace</i> - остановлен, <i>zomb</i> - зомби cmdline - фильтр по командной строке (является регулярным выражением) зона - целевая зона: <i>current</i> (по умолчанию), <i>all</i>. Этот параметр поддерживается только на Solaris платформе.</p>	<p>Ключ</p> <p>самого себя.</p> <p>Параметр тип памяти поддерживается на нескольких <u>платформах</u> начиная с Zabbix 3.0.0.</p> <p>proc.num[<имя>,<пользователь>,<состояние>,<cmdline>,<зона>]</p> <p>Примеры ключей:</p> <ul style="list-style-type: none"> ⇒ proc.num[,mysql] - количество процессов выполняемых под пользователем mysql ⇒ proc.num[apache2,www-data] - количество процессов apache2 выполняемых под пользователем www-data ⇒ proc.num[,oracle,sleep,oracleZABBIX] - количество процессов в спящем состоянии выполняемых под oracle и имеющих oracleZABBIX в содержимом командной строки <p>Смотрите заметки по выбору процессов с параметрами имя и cmdline (специфика для Linux).</p> <p>В Windows, поддерживаются только параметры имя и пользователь.</p> <p>Когда этот элемент данных вызывается с командной строки и содержит параметр командной строки (например, при использовании тестового режима агента: <code>zabbix_agentd -t proc.num[,,apache2]</code>), будет засчитан один дополнительный процесс, так как агент посчитает самого себя.</p> <p><i>Обратите внимание</i>, что в случае когда агент скомпилирован на Solaris без поддержки зон, если задать параметр зона значением <i>current</i> (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 <i>all</i> в этом случае поддерживается.</p>

Описание	Возвращаемое значение	Параметры	Ключ	Комментарии
Чтение аппаратного сенсора.	Число с плавающей точкой	устройство - имя устройства сенсор - имя сенсора режим - возможные значения: <i>avg, max, min</i> (если этот параметр не указан, то устройство и сенсор обрабатываются дословно).	sensor[устройство,сенсор,<режим>]	Значения <i>disk</i> и <i>trace</i> для параметра state поддерживаются начиная с Zabbix 3.4.0. В Linux 2.4, читается <i>/proc/sys/dev/sensors</i> .
Штамп времени (timestamp) загрузки системы.	Целое число (UNIX timestamp)		system.boottime	Пример: ⇒ sensor[w83781d-i2c-0-2d,temp1] До Zabbix 1.8.4, использовался формат sensor[temp1] . В Linux 2.6+, читается <i>/sys/class/hwmon</i> . Более подробное описание смотрите у элемента данных сенсора в Linux. В OpenBSD, читается MIB <i>hw.sensors</i> .
Список найденных CPU/CPU ядер. Используется низкоуровневым обнаружением.	Объект JSON		system.cpu.discovery	Примеры: ⇒ sensor[cpu0,temp0] - температура одного CPU ⇒ sensor[cpu[0-2]\$,temp,avg] - средняя температура первых трех CPU Поддерживается в OpenBSD Zabbix агентом начиная с версии 1.8.4.
Прерывания устройств.	Целое число		system.cpu.intr	
Загрузка CPU .	Число с плавающей точкой	cpu - возможные значения: <i>all</i> (по умолчанию), <i>percru</i> (общая загрузка делится на	system.cpu.load[<cpu>,<режим>]	Поддерживается на всех платформах с 2.4.0. Пример: ⇒ system.cpu.load[,avg5] Параметр <i>percru</i> поддерживается начиная с Zabbix 2.0.0.

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
Количество CPU.	Целое число	<p>количество CPU онлайн)</p> <p>режим - возможные значения: <i>avg1</i> (усреднение за одну минуту, по умолчанию), <i>avg5</i>, <i>avg15</i></p> <p>system.cpu.num[<тип>]</p> <p>тип - возможные значения: <i>online</i> (по умолчанию), <i>max</i></p>	Старое название: system.cpu.loadX Пример: ⇒ system.cpu.num
Количество переключений контекста.	Целое число	system.cpu.switches	Старое название: system[switches]
Утилизация CPU в процентах.	Число с плавающей точкой	<p>system.cpu.util[<cpu>,<тип>,<режим>]</p> <p>cpu - номер CPU (по умолчанию все CPU)</p> <p>тип - возможные значения: <i>idle</i>, <i>nice</i>, <i>user</i> (по умолчанию), <i>system</i> (по умолчанию для Windows), <i>iowait</i>, <i>interrupt</i>, <i>softirq</i>, <i>steal</i>, <i>guest</i> (на ядре Linux 2.6.24 и выше), <i>guest_nice</i> (на ядре Linux 2.6.33 и выше). Параметры времени <i>user</i> и <i>nice</i> более не включают время <i>guest</i> и время <i>guest_nice</i> начиная с Zabbix 3.0.14, 3.4.5 и 4.0.0.</p> <p>режим - возможные значения: <i>avg1</i> (усреднение за одну минуту, по умолчанию), <i>avg5</i>, <i>avg15</i></p> <p>system.hostname[<тип>]</p> <p>тип (только для Windows, не должен использоваться на остальных системах) - возможные значения: <i>netbios</i> (по</p>	Пример: ⇒ system.cpu.util[0,user,avg5] Старое название: system.cpu.idleX, system.cpu.niceX, system.cpu.systemX, system.cpu.userX
Системное имя хоста.	Строка		Получение значения либо функцией GetComputerName() (для netbios), либо функцией gethostname() (для host) в Windows и с помощью команды “hostname” на других системах.

Описание	Возвращаемое значение	Параметры	Ключ	Комментарии
				<p>Примеры возвращаемых значений: в Linux: ⇒ system.hostname → linux-w7x1 ⇒ system.hostname → www.zabbix.com в Windows: ⇒ system.hostname → WIN-SERV2008-I6 ⇒ system.hostname[host] → Win-Serv2008-I6LonG</p>
				<p>Параметр ТИП поддерживается этим элементом данных начиная с версии 1.8.6.</p> <p>Смотрите также более детальную информацию.</p>
Информация о шасси.	Строка	информация - одно из full (по умолчанию), model, serial, type или vendor	system.hw.chassis[<информация>]	<p>Пример: system.hw.chassis[full] Hewlett-Packard HP Pro 3010 Small Form Factor PC CZXXXXXXXXX Desktop]</p> <p>Этот ключ зависит от наличия SMBIOS таблицы.</p> <p>Будет выполнена попытка чтения из DMI таблицы с sysfs, если доступ к sysfs будет неудачным, тогда будет попытка чтения напрямую из памяти.</p> <p>Требуются Права root, потому что значение читается из sysfs или памяти.</p> <p>Поддерживается Zabbix агентом начиная с версии 2.0.</p>
Информация о CPU.	Строка или целое число	cpu - номер CPU или all (по умолчанию) информация - возможные значения: full (по умолчанию), curfreq, maxfreq, model или vendor	system.hw.cpu[<cpu>,<информация>]	<p>Пример: ⇒ system.hw.cpu[0,vendor] → AuthenticAMD</p> <p>Информация собирается из /proc/cpuinfo и из /sys/devices/system/cpu/[cpunum]/cpufreq/cpuinfo_max_freq.</p> <p>Если указаны номер CPU и <i>curfreq</i> или <i>maxfreq</i>, то возвращается</p>

Описание	Возвращаемое значение	Параметры	Ключ
Список PCI или USB устройств.	Текст	тип - pci (по умолчанию) или usb	числовое значение (Гц). Поддерживается Zabbix агентом начиная с версии 2.0.
Список MAC адресов.	Строка	интерфейс - all (по умолчанию) или регулярное выражение формат - full (по умолчанию) или short	system.hw.devices[<тип>] Пример: ⇒ system.hw.devices[pci] → 00:00.0 Host bridge: Advanced Micro Devices [AMD] RS780 Host Bridge [...] Возвращает вывод утилит lspci или lsusb (выполняются без каких либо параметров) Поддерживается Zabbix агентом начиная с версии 2.0.
Системное время.	Целое число - с типом как <i>utc</i> Строка - с типом как <i>local</i> .	utc - (по умолчанию) время с начала Эпохи (00:00:00 UTC, Январь 1, 1970), измеренное в секундах. local - время в формате 'тггг-мм-дд,чч:мм:сс.ннн, +чч:мм'	system.hw.macaddr[<интерфейс>,<формат>] Список MAC адресов чьи имена интерфейсов совпадают с заданным в интерфейс регулярным выражением (<i>all</i> список всех интерфейсов). Пример: ⇒ system.hw.macaddr["eth0\$",full] ⇒ [eth0] 00:11:22:33:44:55 Если формат задан как <i>short</i> , имена интерфейсов и одинаковые MAC адреса не попадают в список. Поддерживается Zabbix агентом начиная с версии 2.0.
Выполнение указанной команды на узле сети.	Текст с результатом выполнения команды	команда - выполняемая команда режим - один из wait (по умолчанию,	system.localtime[<тип>] Параметры для этого элемента данных поддерживаются начиная с версии 2.0. Например: ⇒ system.localtime[local] → создайте элемент данных, используя этот ключ, и далее используйте его для отображения времени узла сети в <u>элементе комплексного экрана Часы</u> . system.run[команда,<режим>] Возвращается до 512КБ (64КБ до Zabbix 2.0.5), включая пробелы в конце, которые отрезаются. Для корректной обработки,

Описание	Возвращаемое значение	Параметры	Ключ
1 - в режиме <i>nowait</i> (независимо от результата выполнения команды)	ожидать завершения выполнения), nowait (не ожидать)		<p>возвращаемые данные должны быть текстом.</p> <p>Пример: ⇒ system.run[ls -l /] → подробный список файлов в папке root.</p> <p><i>Обратите внимание:</i> Для того чтобы включить этот функционал, в файле конфигурации агента должна быть указана опция EnableRemoteCommands=1.</p> <p>Возвращаемое значение элемента данных является стандартным выводом совместно со стандартным выводом ошибок, которые получены от команды. Код выхода не проверяется.</p> <p>Пустой результат разрешается начиная с 2.4.0.</p> <p>Смотрите также: Выполнение команд.</p>

system.stat[источник,<тип>]

Статистика системы. Целое число или число с плавающей точкой

ent - количество процессорных единиц выделенных на ресурс (с плавающей точкой)

kthr,<тип> - информация о состояниях потоков ядра:

- r* - среднее количество запускаемых потоков ядра (дробное)
- b* - среднее количество потоков ядра помещенных в очередь ожидания Менеджера Виртуальной Памяти (с плавающей точкой)

memory,<тип> - информация о использовании виртуальной и реальной памяти:

- avm* - активные виртуальные страницы (целое)
- fre* - размер свободного списка (целое)

page,<тип> - информация об ошибках страниц и активности страниц:

- fi* - операции страничного ввода файлов в секунду (с плавающей точкой)
- fo* - операции страничного вывода файлов в секунду (с плавающей точкой)
- pi* - pages paged in from paging space (с плавающей точкой)
- po* - pages paged out to paging space (с плавающей точкой)
- fr* - pages freed (page replacement) (с плавающей точкой)
- sr* - количество сканированных страниц алгоритмом замещения страниц (с плавающей точкой)

faults,<тип> - соотношение ловушек и прерываний:

- in* - прерывания устройства (с плавающей точкой)
- sy* - системные вызовы (с плавающей точкой)
- cs* - переключения контекста потока ядра (с плавающей

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
		<p>точкой)</p> <p>сри,<тип> - раздельное использование процессорного времени в процентах:</p> <p><i>us</i> - пользовательское время (с плавающей точкой)</p> <p><i>sy</i> - системное время (с плавающей точкой)</p> <p><i>id</i> - время простоя (с плавающей точкой)</p> <p><i>wa</i> - время простоя в ходе которого система обрабатывала запросы I/O диска/NFS (с плавающей точкой)</p> <p><i>pc</i> - количество использованных физических процессоров (с плавающей точкой)</p> <p><i>ec</i> - процентное соотношение размеченного занятого пространства (с плавающей точкой)</p> <p><i>lbusy</i> - показывает утилизация в процентах для логических процессор(ов), которая произошла при выполнении уровней пользователя и системы (с плавающей точкой)</p> <p><i>app</i> - отображает количество доступных физических процессоров в распределенном пуле (с плавающей точкой)</p> <p>disk,<тип> - статистика о дисках:</p> <p><i>bps</i> - отображает суммарное количество данных переданных (чтением или записью) на диск в байтах в секунду (целое)</p> <p><i>tps</i> - показывает количество передач за секунду, которые произошли на физическом диске/ленте (с плавающей точкой)</p> <p>Этот элемент данных поддерживается начиная с версии 1.8.1.</p>	
		system.sw.arch	<p>Пример: ⇒ system.sw.arch → i686</p> <p>Для получения информации используется функция <code>uname()</code>.</p> <p>Поддерживается Zabbix агентом начиная с версии 2.0.</p>
Информация о архитектуре программного обеспечения.	Строка		
Информация об операционной системе.	Строка	<p>информация - возможные значения: full (по умолчанию), short или name</p>	<p>Пример: ⇒ system.sw.os[short] → Ubuntu 2.6.35-28.50-generic 2.6.35.11</p> <p>Информация получается с (обратите внимание, на то что не все файлы и опции представлены во всех дистрибутивах):</p> <ul style="list-style-type: none"> /proc/version (full) /proc/version_signature (short) <p>Параметр PRETTY_NAME из /etc/os-release на системах, которые его поддерживают, или /etc/issue.net (name)</p> <p>Поддерживается Zabbix агентом</p>

Описание	Возвращаемое значение	Параметры	Ключ
Список установленных пакетов.	Текст	<p>пакет - <i>all</i> (по умолчанию) или регулярное выражение</p> <p>менеджер - <i>all</i> (по умолчанию) или конкретный менеджер пакетов</p> <p>формат - <i>full</i> (по умолчанию) или <i>short</i></p>	<p>начиная с версии 2.0.</p> <p>system.sw.packages[<пакет>,<менеджер>,<формат>]</p> <p>Список (в алфавитном порядке) установленных пакетов имена которых совпадают с регулярным выражением заданным в пакет (<i>all</i> списки всех).</p> <p>Пример: ⇒ system.sw.packages[mini,dpkg,short] → python-minimal, python2.6-minimal, ubuntu-minimal</p> <p>пакет - <i>all</i> (по умолчанию) или регулярное выражение</p> <p>менеджер - <i>all</i> (по умолчанию) или конкретный менеджер пакетов</p> <p>формат - <i>full</i> (по умолчанию) или <i>short</i></p>
Статистика файла подкачки в (с устройства в память).	Целое число	<p>устройство - устройство используемое для файла подкачки (по умолчанию <i>all</i>)</p> <p>тип - возможные значения:</p> <p><i>count</i> (количество swapins), <i>sectors</i> (секторов swapped in), <i>pages</i> (страниц swapped in). Смотрите поддержку по платформам для получения деталей по умолчаниям.</p>	<p>Поддерживаемые менеджеры пакетов (выполняемые команды):</p> <p>dpkg (dpkg --get-selections)</p> <p>pkgtool (ls /var/log/packages)</p> <p>rpm (rpm -qa)</p> <p>rasman (rasman -Q)</p> <p>Если указан формат равный <i>full</i>, пакеты группируются по менеджерам пакетов (каждый менеджер с отдельной строки, которая начинается с его имени заключенном в квадратные скобки). Если указан формат равный <i>short</i>, пакеты не группируются и отображаются одной строкой.</p> <p>Поддерживается Zabbix агентом начиная с версии 2.0.</p> <p>system.swap.in[<устройство>,<тип>]</p> <p>устройство - устройство используемое для файла подкачки (по умолчанию <i>all</i>)</p> <p>тип - возможные значения:</p> <p><i>count</i> (количество swapins), <i>sectors</i> (секторов swapped in), <i>pages</i> (страниц swapped in). Смотрите поддержку по платформам для получения деталей по умолчаниям.</p>
			<p>Пример: ⇒ system.swap.in[,pages]</p> <p>Источником этой информации является:</p> <p>/proc/swaps, /proc/partitions, /proc/stat (Linux 2.4) /proc/swaps, /proc/diskstats, /proc/vmstat (Linux 2.6)</p> <p>system.swap.out[<устройство>,<тип>]</p>

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
Статистика файла подкачки из (из памяти в устройство).	Целое число.	<p>устройство - устройство используемое для файла подкачки (по умолчанию <i>all</i>)</p> <p>тип - возможные значения:</p> <ul style="list-style-type: none"> <i>count</i> (количество swapouts), <i>sectors</i> (секторов swapped out), <i>pages</i> (страниц swapped out). <p>Смотрите поддержку по платформам для получения деталей по умолчаниям.</p>	<p>Пример: ⇒ system.swap.out[,pages]</p> <p>Источником этой информации является: /proc/swaps, /proc/partitions, /proc/stat (Linux 2.4) /proc/swaps, /proc/diskstats, /proc/vmstat (Linux 2.6)</p>
Размер файла подкачки в байтах или процентах от общего размера.	<p>Целое число - для количества байт</p> <p>Число с плавающей точкой - для процентов.</p>	<p>устройство - устройство используемое для файла подкачки (по умолчанию <i>all</i>)</p> <p>тип - возможные значения:</p> <ul style="list-style-type: none"> <i>free</i> (свободно в файле подкачки, по умолчанию), <i>pfree</i> (свободно в файле подкачки, в процентах), <i>pused</i> (используемое место в файле подкачки, в процентах), <i>total</i> (полный размер файла подкачки), <i>used</i> (используется места в файле подкачки) 	<p>Если устройство не указано, Zabbix агент будет брать во внимание только swap устройства (файлы), физическая память будет игнорироваться. Например, на Solaris системах команда <i>swap -s</i> включает часть физической памяти и swap устройства (в отличии от <i>swap -l</i>).</p> <p>Обратите внимание, этот ключ может возвращать некорректные данные на виртуализированных (VMware ESXi, VirtualBox) Windows платформах. В этом случае используйте ключ <i>perf_counter[\700(_Total)\702]</i> для получения корректного использования файла подкачки в процентах.</p> <p>Старое название: system.swap.free, system.swap.total</p>
Идентификация системы.	Строка	system.uname	Пример возвращаемого значения (Unix): FreeBSD localhost 4.2-RELEASE FreeBSD 4.2-RELEASE #0: Mon Nov i386
			Пример возвращаемого значения

Описание	Возвращаемое значение	Параметры	Ключ Комментарии
Время работы в секундах.	Целое число		(Windows): Windows ZABBIX-WIN 6.0.6001 Microsoft® Windows Server® 2008 Standard Service Pack 1 x86 В Unix начиная с Zabbix 2.2.0 значение этого элемента извлекается при помощи системного вызова uname(). Ранее эта информация запрашивалась с помощью системного вызова “uname -a”. Значение этого элемента данных может отличаться от вывода “uname -a” и может не включать дополнительную информацию, так как “uname -a” выводит информацию основываясь на других источниках.
Количество пользователей находящихся в системе.	Целое число		В Windows начиная с Zabbix 3.0 значение этого элемента данных извлекается при помощи Win32_OperatingSystem и Win32_Processor классов WMI. Ранее эта информация запрашивалась при помощи непостоянных вызовов Windows API и недокументированных ключей реестра. Имя ОС (включая редакцию) может быть переведено на пользовательский язык интерфейса. Некоторые версии Windows содержат символы товарных знаков и дополнительные пробелы.
Статистика чтения		system.uptime	Обратите внимание, что этот элемент данных в Windows возвращает архитектуру ОС, тогда как в Unix он возвращает архитектуру CPU.
		system.users.num	При настройке элемента данных , используйте единицы измерения s или uptime для получения читаемых значений.
			Для получения значений используется команда who на стороне агента.
		vfs.dev.read[<устройство>,<тип>,<режим>]	Целое число при устройство - дисковое Значения по умолчанию параметра

Описание	Возвращаемое значение	Параметры	Ключ
диска.	устройство (по умолчанию <i>all</i> ²) тип - возможные значения: <i>sectors, operations, bytes, sps, ops, bps</i> Этот параметр необходимо указывать, т.к. умолчания типа равном: <i>sectors, operations, bytes</i>	отличаются для разных ОС. <i>sps, ops, bps</i> соответствуют: секторам, операциям, байтам в секунду соответственно режим - возможные значения: <i>avg1</i> (усреднение за минуту, по умолчанию), <i>avg5</i> (усреднение за 5 минут), <i>avg15</i> (усреднение за 15 минут). Третий параметр поддерживается только, если тип один из: <i>sps, ops, bps</i> .	'Тип' для различных ОС: AIX - operations FreeBSD - bps Linux - sps OpenBSD - operations Solaris - bytes
Статистика записи на диск.	Целое число при устройство - дисковое типа равном: <i>sectors, operations, bytes</i> Число с	устройством (по умолчанию <i>all</i> ¹) тип - возможные значения: <i>sectors, operations,</i>	Пример: ⇒ <code>vfs.dev.read[,operations]</code>
			<i>ops, bps и sps</i> на поддерживаемых платформах ограничено 8 устройствами (7 отдельных устройств и одно <i>all</i>). Начиная с Zabbix 2.0.1 этот лимит увеличен до 1024 (1023 отдельных устройств и один для <i>all</i>).
			По умолчанию используется <i>all</i> первым параметром, который возвращает суммарную статистику, включая: все блочные устройства такие как <i>sda, sdb</i> и их разделы <i>sda1, sda2, sdb3</i> ... и несколько устройств (MD raid) на основе этих блочных устройств/разделов и логические разделы (LVM) на основе этих блочных устройств/разделов. В некоторых случаях возвращаемые значения следует рассматривать как относительные значения (изменяющиеся во времени), но не как абсолютные значения.
			LVM поддерживается начиная с Zabbix 1.8.6.
			До Zabbix 1.8.6, можно было использовать только относительные имена устройств (например, sda), начиная с 1.8.6 можно использовать optionalный префикс /dev/ (например, /dev/sda)
			Старое название: <i>io[*]</i>
			vfs.dev.write[<устройство>,<тип>,<режим>]
			Значения по умолчанию параметра 'Тип' для различных ОС: AIX - operations FreeBSD - bps Linux - sps OpenBSD - operations

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
		Solaris - bytes	
		Пример: ⇒ vfs.dev.write[,operations]	
		<i>ops, bps и sps</i> на поддерживаемых платформах ограничено 8 устройствами (7 отдельных устройств и одно <i>all</i>). Начиная с Zabbix 2.0.1 этот лимит увеличен до 1024 (1023 отдельных устройств и один для <i>all</i>).	
		<i>bytes, sps, ops, bps</i> Этот параметр необходимо указывать, т.к. умолчания отличаются для разных ОС.	По умолчанию используется <i>all</i> первым параметром, который возвращает суммарную статистику, включая: все блочные устройства такие как <i>sda</i> , <i>sdb</i> и их разделы <i>sda1</i> , <i>sda2</i> , <i>sdb3</i> ... и несколько устройств (MD raid) на основе этих блочных устройств/разделов и логические разделы (LVM) на основе этих блочных устройств/разделов. В некоторых случаях возвращаемые значения следует рассматривать как относительные значения (изменяющиеся во времени), но не как абсолютные значения.
		<i>sps, ops, bps</i> соответствуют: секторам, операциям, байтам в секунду соответственно режим - возможные значения: <i>avg1</i> (усреднение за минуту, по умолчанию), <i>avg5</i> (усреднение за 5 минут), <i>avg15</i> (усреднение за 15 минут). Третий параметр поддерживается только если ТИП один из: <i>sps, ops, bps</i> .	LVM поддерживается начиная с Zabbix 1.8.6.
			До Zabbix 1.8.6, можно было использовать только относительные имена устройств (например, sda), начиная с 1.8.6 можно использовать optionalный префикс /dev/ (например, /dev/sda)
			Старое название: <i>io[*]</i>
vfs.dir.count[директория,<regex_вкл>,<regex_искл>,<типы_вкл>,<типы_искл>,<макс_глубина>,<мин_размер>,<макс_размер>,<мин_возраст>,<макс_возраст>]	Количество записей в директории.	Целое число	<p>директория - абсолютный путь к директории</p> <p>regex_вкл - регулярное выражение описывающее включаемые файл, директорию и символьскую ссылку (если пусто,</p> <p>Переменные сред, такие как %APP_HOME%, \$HOME и %TEMP%, не поддерживаются.</p> <p>Псевдо-директории “.” и “..” никогда не учитываются.</p> <p>Символические ссылки никогда проверяются для обхода директорий.</p> <p>В Windows символические ссылки</p>

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
	<p>включаются все файлы, директории и символические ссылки; пустая строка является значением по умолчанию)</p> <p>regex_искл - регулярное выражение описывающее исключаемые файл, директорию и символическую ссылку (если пусто, ничего не исключать; пустая строка является значением по умолчанию)</p> <p>типы_вкл - перечисление типов записей в директории, которые будут подсчитываться, возможные значения: <i>file</i> - обычный файл, <i>dir</i> - поддиректория, <i>sym</i> - символическая ссылка, <i>sock</i> - соект, <i>bdev</i> - блочное устройство, <i>cdev</i> - устройство посимвольного ввода-вывода, <i>fifo</i> - FIFO, <i>dev</i> - синонимы к “<i>bdev,cdev</i>”, <i>all</i> - все вышеупомянутые типы, т.е. “<i>file,dir,sym,sock,bdev,cdev,fifo</i>”. Это значение по умолчанию, если параметр оставлен пустым. Несколько типов необходимо разделять запятой и весь список нужно заключать в кавычки “”.</p> <p>типы_искл - перечисление типов записей в директории, которые НЕ будут подсчитываться, те же</p>	<p>директорий пропускаются и жесткие ссылки учитываются лишь один раз.</p> <p><code>regex_incl</code> и <code>regex_excl</code> являются Perl совместимыми регулярными выражениями (PCRE).</p> <p>Оба параметра <code>regex_vkl</code> и <code>regex_искл</code> применяются к файлам и директориям при вычислении общего размера, но игнорируются при выборе поддиректорий (если <code>regex_vkl</code> равен "(?i)^.+\.zip\$" и <code>maxs_глубина</code> не задана, тогда проход будет выполнен по всем поддиректориям, но учтены будут только файлы с типом zip).</p> <p>Если имя файла совпадает как с <code>regex_vkl</code>, так и с <code>regex_искл</code>, такой файл не будет учитываться.</p>	<p>Время выполнения ограничено значением времени ожидания по умолчанию, которое равно 3 секундам (параметр “<i>Timeout</i>” в файле конфигурации агента). Та как обход большой директории может занять более длительное время чем время ожидания, тогда никакие данные не возвращаются и элемент данных отметится как “Не поддерживается”. Частичный подсчет не возвращается.</p> <p>При фильтрации по размеру только обычные файлы имеют осмысленные размеры. В Linux и BSD директории также имеют не нулевые размеры (в основном несколько Кб). Устройства имеют нулевые размеры, например, размер <code>/dev/sda1</code> не влияет на соответствующий размер раздела. Поэтому, при использовании параметров <code><мин_размер></code> и <code><макс_размер></code> рекомендуется указать <code><типы_вкл></code> равным “<i>file</i>”, чтобы избежать сюрпризов.</p> <p>Примеры:</p> <ul style="list-style-type: none"> ⇒ <code>vfs.dir.count[/dev]</code> - мониторинг количества устройств в <code>/dev</code> (Linux) ⇒ <code>vfs.dir.count["C:\Users\ADMINI~1\AppData\Local\Temp"]</code> - мониторинг

Ключ

Описание	Возвращаемое значение	Параметры	Комментарии
		<p>самые значения и синтаксис что и для <типы_вкл>. Если некоторые типы записей указаны как в <типы_вкл>, так и в <типы_искл>, записи этих типов в директории НЕ будут подсчитываться.</p> <p>макс_глубина - максимальная глубина сканируемых поддиректорий. -1 (по умолчанию) - без ограничения, 0 - без проверки нижестоящих поддиректорий.</p> <p>мин_размер - минимальный размер подсчитываемых файлов. Файлы, которые меньше этого значения не будут подсчитаны. Значение в байтах. Можно использовать суффиксы памяти.</p> <p>макс_размер - максимальный размер подсчитываемых файлов. Файлы, которые больше этого значения не будут подсчитаны. Значение в байтах. Можно использовать суффиксы памяти.</p> <p>мин_возраст - минимальный возраст подсчитываемой записи директории. Записи, которые изменены раньше не будут подсчитаны. Целочисленное значение в виде секунд. Можно использовать суффиксы времени.</p> <p>макс_возраст - максимальный</p>	<p>количество файлов во временной директории (Windows)</p> <p>Поддерживается начиная с Zabbix 4.0.0.</p>

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
Размер директории (в байтах).	Целое число	<p>возраст подсчитываемой записи директории. Записи, которые настолько старые не будут подсчитаны (время модификации). Целочисленное значение в виде секунд. Можно использовать суффиксы времени.</p> <p>vfs.dir.size[директория,<regex_вкл>,<regex_искл>,<режим>,<макс_глубина>]</p> <p>директория - абсолютный путь к директории</p> <p>regex_вкл - регулярное выражение описывающее включаемые файл, директорию и символьическую ссылку (если пусто, включаются все файлы, директории и символические ссылки; пустая строка является значением по умолчанию)</p> <p>regex_искл - регулярное выражение описывающее исключаемые файл, директорию и символьическую ссылку (если пусто, ничего не исключать; пустая строка является значением по умолчанию)</p> <p>режим - возможные значения: <i>apparent</i> (по умолчанию) - получение реальных размеров файлов вместо использования диска (работает как <i>du -sb</i> директория), <i>disk</i> -</p>	<p>Подсчитываются только те директории к которым zabbix пользователь имеет по крайней мере права чтения.</p> <p>В Windows любая символическая ссылка пропускается и жесткие ссылки принимаются во внимание только один раз.</p> <p>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.</p> <p>Примеры: ⇒ <code>vfs.dir.size[/tmp,log]</code> - вычисление размеров всех файлов в /tmp, которые содержат в своём имени 'log' ⇒ <code>vfs.dir.size[/tmp,log,^.+\old\$]</code> - вычисление размеров всех файлов в /tmp, которые содержат в своём имени 'log', исключая файлы содержащие в своем имени '.old'</p> <p>Ограничение размера файлов зависит от наличия поддержки больших файлов.</p> <p>Поддерживается начиная с Zabbix 3.4.0.</p>

Описание	Возвращаемое значение	Параметры	Комментарии
		<p>получение использования диска (работает как du -s -B1 директория). В отличии от du команды, элемент данных vfs.dir.size принимает во внимание скрытые файлы при вычислении размера директории (работает как du -sb .[^.]* * в пределах директории).</p> <p>макс_глубина - максимальная глубина сканируемых поддиректорий. -1 (по умолчанию) - без ограничения, 0 - без проверки нижестоящих поддиректорий.</p>	
		vfs.file.cksum[файл]	<p>Пример: ⇒ vfs.file.cksum[/etc/passwd]</p>
Контрольная сумма файла, вычисленная по алгоритму используемом в UNIX cksum.	Целое число	файл - абсолютный путь к файлу	<p>Пример возвращаемого значения: 1938292000</p> <p>Старое название: cksum</p> <p>Ограничение размера файла зависит от поддержки больших файлов.</p>
		vfs.file.contents[файл,<кодировка>]	<p>Возвращает пустую строку, если файл пустой или содержит только LF/CR символы.</p>
Получение содержимого файла.	Текст	<p>файл - абсолютный путь к файлу</p> <p>кодировка - идентификатор кодовой страницы</p>	<p>Пример: ⇒ vfs.file.contents[/etc/passwd]</p> <p>Этот элемент данных ограничен файлами не превышающими 64 Кбайт.</p> <p>Поддерживается Zabbix агентом начиная с версии 2.0.</p>
		vfs.file.exists[файл]	

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
Проверка существования файла	0 - файл не найден 1 - обычный файл или ссылка (символическая или жесткая) на обычный файл существует.	файл - абсолютный путь к файлу	Пример: ⇒ vfs.file.exists[/tmp/application.pid] Возвращаемое значение зависит от того, что вернет S_ISREG POSIX макрос.
		vfs.file.md5sum[файл]	Ограничение размера файла зависит от поддержки больших файлов .
MD5 контрольная сумма файла	Строка (MD5 хэш файла)	файл - абсолютный путь к файлу	Пример: ⇒ vfs.file.md5sum[/usr/local/etc/zabbix_agentd.conf] Пример возвращаемого значения: b5052decb577e0ffd622d6ddc017e82 Ограничение (64МБ) на размер файла для этого элемента данных удалено в версии 1.8.6.
			Ограничение размера файла зависит от поддержки больших файлов .
Поиск строки в файле.	Строка содержащая совпадающую подстроку или то, что определено в дополнительном параметре вывод.	файл - абсолютный путь к файлу регулярное выражение - Perl совместимое регулярное выражение (PCRE) или расширенное POSIX регулярное выражение до Zabbix 3.4 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).	Возвращается только первая совпадающая строка. Будет возвращена пустая строка, если не найдено совпадений с выражением. Извлечение содержимого при использовании параметра вывод выполняется агентом.
			Параметры начальная строка , конечная строка и вывод поддерживаются начиная с версии 2.2.
			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] → получение ID пользователя zabbix

Описание	Возвращаемое значение	Параметры	Комментарии
Поиск строки в файле.	0 - совпадение не найдено 1 - найдено	вывод - дополнительный шаблон форматирования вывода. \0 - управляющая последовательность заменяется найденным текстом, тогда \N (где N=1..9) - управляющая последовательность заменяется N-нной совпадающей группой (или пустой строкой, если N превышает количество найденных групп).	файл - абсолютный путь к файлу регулярное выражение - Perl совместимое регулярное выражение (PCRE) или расширенное POSIX регулярное выражение до Zabbix 3.4 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).
Размер файла (в байтах).	Целое число	vfs.file.size[файл]	Пользователь zabbix должен иметь права на чтение файла Например: <code>vfs.file.size[/var/log/app.log,error]</code>

Описание	Возвращаемое значение	Параметры	Ключ
Информация о времени файла.	Целое число (Unix timestamp)	vfs.file.time[файл,<режим>] файл - полный путь к файлу режим - возможные значения: <i>modify</i> (по умолчанию) - время последней модификации содержимого файла, <i>access</i> - время последнего чтения файла, <i>change</i> - время последнего изменения свойств файла	⇒ vfs.file.size[/var/log/syslog] Ограничение размера файла зависит от поддержки больших файлов .
Список примонтированных файловых систем. Используется низкоуровневым обнаружением.	JSON объект	vfs.fs.discovery	Пример: ⇒ vfs.file.time[/etc/passwd,modify] Ограничение размера файла зависит от поддержки больших файлов .
Количество или процент inodes.	Целое число - для количества Число с плавающей точкой - для процентов	vfs.fs.inode[fs,<режим>] fs - файловая система режим - возможные значения: <i>total</i> (по умолчанию), <i>free</i> , <i>used</i> , <i>pfree</i> (свободно, в процентах), <i>pused</i> (использовано, в процентах)	Поддерживается Zabbix агентом начиная с версии 2.0. Макрос {#FSDRIVETYPE} поддерживается в Windows начиная с Zabbix агента версии 3.0.
Размер диска в байтах или процентах от общего размера.	Целое число - для байт Число с плавающей точкой - для процентов.	vfs.fs.size[fs,<режим>] fs - файловая система режим - возможные значения: <i>total</i> (по умолчанию), <i>free</i> , <i>used</i> , <i>pfree</i> (доступно, в процентах), <i>pused</i> (использовано, в процентах)	В случае примонтированного раздела, будет возвращен размер диска локального файла системы. Пример: ⇒ vfs.size[/tmp,free] Зарезервированное место на файловой системе принимается во внимание и не включено при использовании режима <i>free</i> . Старые названия: <i>vfs.free[*]</i> ,

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
Размер памяти в байтах или в процентах от общего количества.	Целое число - для байт Число с плавающей точкой - для процентов.	режим - возможные значения: <i>total</i> (по умолчанию), <i>active</i> , <i>anon</i> , <i>buffers</i> , <i>cached</i> , <i>exec</i> , <i>file</i> , <i>free</i> , <i>inactive</i> , <i>pinned</i> , <i>shared</i> , <i>slab</i> , <i>wired</i> . (использовано, в процентах), <i>available</i> , <i>available</i> (доступно, в процентах)	<i>vfs.fs.total[*], vfs.fs.used[*], vfs.fs.pfree[*], vfs.fs.pused[*]</i> Этот элемент данных принимает три категории параметров: 1) <i>total</i> - общего количества памяти. 2) специфичные для платформ типы памяти: <i>active</i> , <i>anon</i> , <i>buffers</i> , <i>cached</i> , <i>exec</i> , <i>file</i> , <i>free</i> , <i>inactive</i> , <i>pinned</i> , <i>shared</i> , <i>slab</i> , <i>wired</i> . 3) оценка на уровне пользователя как много памяти используется и доступно: <i>used</i> , <i>pused</i> , <i>available</i> , <i>available</i> .
Получение содержимого веб-страницы.	Исходная веб-страница как текст (включая заголовки)	хост - имя хоста путь - путь к HTML документу (по умолчанию /) порт - номер порта (по умолчанию 80)	Смотрите более подробное описание параметров vm.memory.size . Старые названия: <i>vm.memory.buffers</i> , <i>vm.memory.cached</i> , <i>vm.memory.free</i> , <i>vm.memory.shared</i> , <i>vm.memory.total</i>
Время полной загрузки веб-страницы (в секундах).	Число с плавающей точкой	хост - имя хоста путь - путь к HTML документу (по умолчанию /) порт - номер порта (по умолчанию 80)	Возвращает пустую строку при ошибке. Например: ⇒ <code>web.page.get[www.zabbix.com,index.php,80]</code>
Поиск строки на веб-странице.	Совпадающая строка, или как указан опциональный параметр вывод	хост - имя хоста путь - путь к HTML документу (по умолчанию - /) порт - номер порта (по умолчанию - 80) регулярное выражение - Perl совместимое регулярное выражение (PCRE) или расширенное	Возвращает 0 при ошибке. Например: ⇒ <code>web.page.perf[www.zabbix.com,index.php,80]</code> Возвращается пустая строка, если нет совпадений или при ошибке. Извлечение содержимого согласно дополнительного параметра вывод осуществляется агентом. Параметр вывод поддерживается начиная с версии 2.2. Пример: ⇒
web.page regexp[хост,<путь>,<порт>,регулярное выражение,<длина>,<вывод>]			

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
		<p>POSIX регулярное выражение до Zabbix 3.4</p> <p>длина - максимальное количество возвращаемых символов</p> <p>вывод - дополнительный шаблон форматирования вывода. \0 - управляющая последовательность заменяется найденным текстом, тогда \N (где N=1..9) - управляющая последовательность заменяется N-нной совпадающей группой (или пустой строкой, если N превышает количество найденных групп).</p>	web.page-regexp[www.zabbix.com,inde x.php,80,OK,2]

Заметка специфики для Linux. Zabbix агент должен иметь только права чтения файловой системы /proc. Патчи к ядру для ограничения прав непrivилегированных пользователей вы можете найти на странице www.grsecurity.org.

Доступные кодировки

Параметр Кодировка используется для того, чтобы указать кодировку при обработке соответствующих проверок элементов данных, так чтобы полученные данные не были повреждены. Для получения списка поддерживаемых кодировок (идентификаторов кодовых страниц), пожалуйста, обратитесь к соответствующей документации, такой как libiconv (GNU Project) или Microsoft Windows SDK документации по “Идентификаторам кодовых страниц”.

Если задана пустая Кодировка, тогда по умолчанию используются UTF-8 (языковой стандарт по умолчанию для новых Unix/Linux дистрибутивов, смотрите настройки вашей системы) или ANSI с определенным расширением в системе (Windows).

Поиск проблем с элементами данных агента

- Если используются с пассивным агентом, значение *Timeout* конфигурации сервера, возможно, потребуется выше, чем *Timeout* в файле конфигурации агента. В противном случае элемент данных может не возвращать никаких значений по причине того, что запрос сервера к агенту превысит время ожидания раньше агента.

Специфичные ключи элементов данных для Windows

Ключи элементов данных

В таблице приводится подробная информация о ключах элементов данных, которые вы можете использовать только с Zabbix Windows агентом.

▲	Описание	Возвращаемое значение	Ключ	Параметры	Комментарии
	Мониторинг журналов событий.	Журнал (лог)	<code>eventlog[имя,<регулярное выражение>,<важность>,<источник>,<eventid>,<макс. кол-во строк>,<режим>]</code>	<p>имя - имя журнала событий</p> <p>регулярное выражение - регулярное выражение описывающее требуемый шаблон содержимого</p> <p>важность - регулярное выражение описывающее важность</p> <p>Параметр может принимать следующие значения:</p> <p>“Information”, “Warning”, “Error”, “Critical”, “Verbose” (начиная с Zabbix 2.2, работающих на Windows Vista или на более новых версиях)</p> <p>источник - регулярное выражение, описывающее идентификатор источника (регулярное выражение поддерживается начиная с версии Zabbix 2.2.0)</p> <p>eventid - регулярное выражение описывающее идентификатор(ы) событий</p> <p>макс. кол-во строк - максимальное количество новых строк в секунду, которое агент будет отправлять Zabbix серверу или прокси. Этот параметр заменяет значение</p>	<p>Элемент данных должен быть настроен активной проверкой.</p> <p>Примеры:</p> <p>⇒ <code>eventlog[Application]</code></p> <p>⇒ <code>eventlog[Security,"Failure Audit",,529 680]</code></p> <p>⇒ <code>eventlog[System,"Warning Error"]</code></p> <p>⇒ <code>eventlog[System,,,^1\$]</code></p> <p>⇒ <code>eventlog[System,,,@TWOSHORT]</code> - здесь используется ссылка на пользовательское регулярное выражение с именем TWOSHORT (заданное с типом Результат ИСТИНА, само выражение равно <code>^1\$ ^70\$</code>).</p> <p><i>Обратите внимание</i>, агент не может отправлять события из "Пересланные события" журнала.</p> <p>Параметр режим поддерживается начиная с версии 2.0.0. “Windows Eventing 6.0” поддерживается начиная с Zabbix 2.2.0.</p> <p>Обратите внимание, что выбор не журнального типа информации для этого элемента данных приведет к потере локального штампа времени, а также важности журнала и информации о источнике.</p> <p>Смотрите дополнительную информацию о мониторинге файлов журналов.</p>

	Описание	Возвращаемое значение	Параметры	Комментарии
▲	Список сетевых интерфейсов (включая тип, состояние, IPv4 адрес, описание интерфейса).	Текст	'MaxLinesPerSecond' в zabbix_agentd.win.conf режим - возможные значения: <i>all</i> (по умолчанию), <i>skip</i> - пропустить обработку старых данных (влияет только на недавно созданные элементы данных). net.if.list	Поддерживается Zabbix агентом начиная с версии 1.8.1. Начиная с версии 1.8.6 Zabbix агента поддерживаются мультибайтные имена интерфейса. Отключенные интерфейсы не входят в список.
	Значение любого счетчика производительности Windows.	Целое число, число с плавающей точкой, строка или текст (в зависимости от запроса)	perf_counter[счетчик,<период>] счетчик - путь к счетчику период - последние N секунд для сохранения усредненного значения. Значение период должно быть равно значению с 1 до 900 секунд (включительно), значение по умолчанию 1.	Обратите внимание, что включение/отключение некоторых компонентов Windows могут изменить порядок имён интерфейсов в Windows. В некоторых версиях Windows (к примеру, Server 2008) может потребоваться установка последних обновления для поддержки не-ASCII символов в именах интерфейсов.

Ключ

▲ Описание	Возвращаемое значение	Параметры	Комментарии
Различная информация о указанном процессе(ах).	<code>proc_info[<процесс>,<атрибут>,<тип>]</code>	Число с плавающей точкой процесса <code><процесс></code> - имя запрашиваемый атрибут процесса. <code><тип></code> - тип представления (имеет смысл, когда есть более одного процесса с одним именем)	среднее значение за последние "период" секунд. Смотрите также: Счетчики производительности в Windows .

Описание	Возвращаемое значение	Ключ	Параметры	Комментарии
				<процесс> max - максимальное значение среди всех процессов с именем <процесс> avg - среднее значение среди всех процессов с именем <процесс> sum - сумма значений для всех процессов с именем <процесс>
				Примеры: ⇒ proc_info[iexplore.exe,wkset,sum] - для получения общего количества физической памяти выделенной под все процессы Internet Explorer ⇒ proc_info[iexplore.exe,pf,avg] - для получения среднего количества ошибок на страницах для процессов Internet Explorer
				Обратите внимание, что для корректной работы этого элемента данных на 64-битной системе потребуется 64-битный Zabbix агент.
				Обратите внимание: Все атрибуты <i>io_*</i> , <i>gdiobj</i> и <i>userobj</i> доступны только в Windows 2000 и более поздних версиях Windows, не в Windows NT 4.0.
			service.discovery	
Список служб Windows. Используется низкоуровневым обнаружением.	Объект JSON			Поддерживается Zabbix агентом начиная с версии 3.0.
Информация о службе.			service.info[служба,<парам>]	
	Целое число - с парам равным <i>state, startup</i>	служба - действительное имя службы или её отображаемое имя как		Примеры: ⇒ service.info[SNMPTRAP] - состояние службы SNMPTRAP ⇒ service.info[SNMP Trap] - состояние этой же службы, но указано отображаемое имя ⇒ service.info[EventLog,startup] - состояние запуска при загрузке службы Журнала событий
	Строка - с парам равным <i>displayname, path, user</i>	в оснастке MMC Службы парам - state (по умолчанию), <i>displayname, path, user, startup</i> или <i>description</i>		
	Текст - с парам			

Описание	Возвращаемое значение	Параметры	Комментарии
	<p>равным <i>description</i></p> <p>В частности при <i>state</i>:</p> <p>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</p> <p>В частности при <i>startup</i>:</p> <p>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</p>		<p>Элементы данных service.info[служба,state] and service.info[служба] вернут одинаковую информацию.</p> <p>Обратите внимание, что только параметр равный <i>state</i> у этого элемента данных возвращает значение по несуществующим службам (255).</p> <p>Этот элемент данных поддерживается начиная с Zabbix 3.0.0. Его необходимо использовать вместо устаревшего элемента данных service_state[служба].</p>
Список служб.	<p>services[<тип>,<состояние>,<исключение>]</p> <p>0 - если список служб пуст.</p> <p>Текст - список служб, разделенных новой строкой.</p>	<p>тип - <i>all</i> (по умолчанию), <i>automatic</i>, <i>manual</i>, <i>disabled</i></p> <p>состояние - <i>all</i> (по умолчанию), <i>stopped</i>, <i>started</i>, <i>start_pending</i>, <i>stop_pending</i>, <i>running</i>, <i>continue_pending</i>, <i>pause_pending</i>, <i>paused</i></p> <p>исключение - список</p>	<p>Примеры:</p> <p>⇒ services[<i>started</i>] - список запущенных служб</p> <p>⇒ services[<i>automatic</i>, <i>stopped</i>] - список остановленных служб, которые должны быть запущены</p> <p>⇒ services[<i>automatic</i>, <i>stopped</i>, "<i>service1,service2,service3</i>"] - список остановленных служб.</p>

	Описание	Возвращаемое значение	Параметры	Комментарии
Выполнение WMI запроса и получение первого выбранного объекта.		Службы исключенных из результата. Исключенные службы должны быть указаны в двойных кавычках, разделенные запятой, без пробелов.		которые должны быть запущены, исключая службы с именами service1,service2 и service3 Параметр исключения поддерживается начиная с версии 1.8.1.
Размер виртуального пространства в байтах или в процентах от общего размера.	wmi.get[<пространство_имен>,<запрос>]	Целое число, число с плавающей точкой, строка или текст(в зависимости от запроса.)	пространство_имен - название пространства имен WMI запрос - WMI запрос, возвращающий один объект	Пример: ⇒ wmi.get[root\cimv2,select status from Win32_DiskDrive where Name like '%PHYSICALDRIVE0%'] - возвращает состояние первого физического диска Этот ключ поддерживается начиная с Zabbix 2.2.0.
	vm.vmemory.size[<тип>]	Целое число - для байт Число с плавающей точкой - для процентов	тип - возможные значения: <i>available</i> (доступно виртуальной памяти), <i>availble</i> (доступно виртуальной памяти, в процентах), <i>pused</i> (использовано виртуальной памяти, в процентах), <i>total</i> (всего виртуальной памяти, по умолчанию), <i>used</i> (использовано виртуальной памяти)	Пример: ⇒ vm.vmemory.size[available] → доступно виртуальной памяти, в процентах Мониторинг статистики виртуальной памяти на основе: Максимального количества памяти, которое может занять Zabbix агент. Текущий предел выделенной памяти в системе или Zabbix агенте, смотря что меньше. Этот ключ поддерживается начиная с Zabbix 3.0.7 и 3.2.3.

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, перейдя в оснастку MMC Службы и открыв свойства службы. На вкладке Общие вы должны увидеть поле называемое 'Имя службы'. Значение которого и будет именем желаемой службы, которое вы будете использовать при настройке элемента данных для наблюдения.

Например, если вы хотите наблюдать службу "workstation", то ваша служба скорее всего будет: **lanmanworkstation**.

Шаг 2

Настройте элемент данных для наблюдения за службой.

Элемент данных service.info[служба,<парам>] возвращает информацию о указанной службе. В зависимости от требуемой вам информации, укажите опцию *парам*, которая принимает следующие значения: *displayname*, *state*, *path*, *user*, *startup* или *description*. Значением по умолчанию является *state*, если *парам* не указан (service.info[служба]).

Тип возвращаемого значения зависит от выбранного *парам*: целое число при *state* и *startup*; строка символов при *displayname*, *path* и *user*; текст при *description*.

Пример:

- Ключ: service.info[lanmanworkstation]
- Тип информации: Целочисленное (положительное)
- Отображение значений: выберите преобразование значений Windows service state

Имеется два преобразования значений Windows service state и Windows service startup type, которые сопоставляют числовое значение в веб-интерфейсе его текстовому представлению.

Обнаружение служб Windows

Низкоуровневое обнаружение дает возможность автоматического создания элементов данных, триггеров и графиков по различным 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.

7.2.2.2 SNMP агент

Обзор

Вы возможно захотите использовать SNMP мониторинг устройств таких как принтеры, сетевые коммутаторы, маршрутизаторы или ИБП, как правило, которые как правило поддерживают SNMP и для которых было бы непрактично пытаться настраивать комплексные системы управления или Zabbix агенты.

Чтобы была возможность получать данные переданные SNMP агентами с этих устройств, Zabbix сервер должен быть изначально сконфигурирован с поддержкой SNMP.

SNMP проверки выполняются только через UDP протокол.

Начиная с версии 2.2.3 демоны Zabbix сервера и прокси опрашивают устройства SNMP множественными значениями за один запрос. Это поведение повлияет на все виды SNMP элементов данных (простые SNMP элементы данных, элементы данных с динамическими индексами и также низкоуровневые SNMP обнаружения) и обработка SNMP элементов данных сейчас должна быть более эффективной. Пожалуйста обратите внимание на раздел с техническими подробностями ниже, описывающий как работает изнутри этот функционал. Начиная с Zabbix 2.4 у каждого интерфейса также имеется настройка “Использовать массовые запросы”, которая позволяет отключать массовые запросы у устройств, которые не способны обработать их должным образом.

Начиная с Zabbix 2.2.7 и Zabbix 2.4.2 процессы сервера и прокси будут журналировать строки похожие на следующие в случае получения неправильного/искаженного SNMP ответа:

SNMP response from host "gateway" does not contain all of the requested variable bindings

Пока они не покрывают все возможные проблемные случаи, но они являются удобным идентификатором отдельных SNMP устройств на которых необходимо отключить массовые запросы.

Начиная с версии Zabbix 2.2 демоны сервера и прокси корректно обрабатывают параметр конфигурации `Timeout` при выполнении SNMP проверок. Дополнительно демоны не выполняют повторных запросов после одного неуспешного (по превышении времени ожидания/неверные настройки учетных данных) SNMP запроса. Ранее на самом деле использовались стандартные для библиотеки SNMP значения времени ожидания и количества повторов (1 секунда и 5 повторов соответственно).

Начиная с версии Zabbix 2.2.8 и Zabbix 2.4.2 демоны сервера и прокси всегда выполняют один повторный запрос: либо через механизм библиотеки SNMP, либо через [внутренний механизм сбора множества значений за один запрос \(bulk\)](#).

Если выполняется мониторинг устройств по SNMPv3, убедитесь что `msgAuthoritativeEngineID` (также известное как `snmpEngineID` или “Engine ID”) никогда не будет общим для двух и более устройств. Согласно RFC 2571 (раздел 3.1.1.1) оно должно быть уникальным для каждого устройства.

Настройка мониторинга по SNMP

Для начала мониторинга устройства по SNMP, должны быть выполнены следующие шаги:

Шаг 1

[Создайте узел сети](#) для устройства с SNMP интерфейсом.

Ведите IP адрес. Вы можете использовать один из поставляемых шаблонов SNMP (*Template SNMP Device* и другие), которые автоматически добавят некоторый набор элементов данных. Тем не менее, шаблон может быть не совместим с узлом сети. Нажмите на *Добавить* для сохранения узла сети.

SNMP проверки не используют *Порт агента*, он игнорируется.

Шаг 2

Узнайте строку SNMP (или OID) элемента данных, которую вы хотите мониторить.

Для получения списка строк SNMP, используйте команду `snmpwalk` (часть программного обеспечения [net-snmp](#), которое вы должны были установить как часть инсталляции Zabbix) или эквивалентную утилиту:

```
shell> snmpwalk -v 2c -c public <IP хоста> .
```

'2c' здесь означает версию SNMP, вы также можете заменить его на '1', чтобы использовать 1 версию SNMP на устройстве.

Эта команда должна показать вам список SNMP строк и их последние значения. Если это не произойдет, то возможно что SNMP 'community' отличается от стандартного 'public', в этом случае вам необходимо узнать это имя.

Вы можете пройтись по списку пока не найдете строку которую вы хотите мониторить, например, если вы хотите мониторить входящее количество байт на вашем коммутаторе на 3 порту вы могли бы использовать `IF-MIB::ifInOctets.3` из этой строки:

```
IF-MIB::ifInOctets.3 = Counter32: 3409739121
```

Сейчас вы можете воспользоваться командой `snmpget` для того чтобы определить цифровой OID для 'IF-MIB::ifInOctets.3':

```
shell> snmpget -v 2c -c public -On 10.62.1.22 IF-MIB::ifInOctets.3
```

Обратите внимание, что последнее число в строке это номер порта, который вы ищите для мониторинга. Смотрите также: [Динамические индексы](#).

Вывод команды покажет вам что-то наподобие этого:

```
.1.3.6.1.2.1.2.2.1.10.3 = Counter32: 3472126941
```

Опять же, последнее число в OID является номером порта.

ЗСОМ кажется использует номера портов сотнями, например 1 порт = 101 порт, 3 порт = 103 порт, но в Cisco используются обычные номера, например, 3 порт = 3.

Некоторые из наиболее часто используемых SNMP OID'ов [автоматически конвертируются Zabbix'ом в числовое представление](#).

В последнем примере выше тип значение “Counter32” (32-битный счетчик), что внутренне соответствует типу ASN_COUNTER. Полный список поддерживаемых типов ASN_COUNTER, ASN_COUNTER64, ASN_UNSIGNED, ASN_INTEGER, ASN_UNSIGNED64, ASN_INTEGER64, ASN_FLOAT, ASN_DOUBLE, ASN_TIMETICKS, ASN_GAUGE, ASN_IPADDRESS, ASN_OCTET_STR и ASN_OBJECT_ID (с 2.2.8, 2.4.3). Приведенные типы грубо соответствуют “Counter32”, “Counter64”, “UInteger32”, “INTEGER”, “Float”, “Double”, “Timeticks”, “Gauge32”, “IpAddress”, “OCTET STRING”, “OBJECT IDENTIFIER” в выводе **snmpget** утилиты, но могут также отображаться как “STRING”, “Hex-STRING”, “OID” и другие, в зависимости от наличия полученной подсказки.

Шаг 3

Создайте элемент данных для мониторинга.

Итак, вернитесь назад в Zabbix и нажмите на Элементы данных, выберите созданный ранее узел сети SNMP. В зависимости от того использовали ли вы шаблон при создании узла сети или нет, вы должны будете увидеть список элементов данных SNMP, связанных с вашим узлом сети или попросту окно нового элемента данных. Мы будем исходить из предположения, что вы собираетесь создать элемент данных самостоятельно, с помощью информации, которую вы только что собрали используя snmpwalk или snmpget, так что введите простое описание на русском языке (или английском) в поле 'Описание' в диалоге нового элемента данных. Убедитесь, что в поле 'Узел сети' находится ваш коммутатор/роутер и измените поле 'Тип' в значение “SNMPv* агент”. Введите community (обычно public) и укажите текстовый или числовой OID, который вы получили ранее, в поле 'SNMP OID', например: .1.3.6.1.2.1.2.2.1.10.3

Введите 'Порт SNMP' - 161 и 'Ключ' - что-то осмысленное, например, SNMP-InOctets-Bps. Установите 'Тип информации' в значение равное Числовой (с плавающей точкой) и шаг предобработки как Скоро в секунду (важно, в противном случае вы будете получать накопленные значения с SNMP устройства вместо последнего изменения). Выберите множитель, если желаете, и укажите 'Интервал обновления', и 'Хранение истории', если вы хотите чтобы значения параметров отличались от умолчаний.

Items

All hosts / Zabbix server Enabled ZBX SNMP JMX IPMI Applications 13 Items 81 Triggers 47

Item Preprocessing

* Name	SNMP: InOctets (Bps)
Type	SNMPv3 agent
* Key	SNMP-InOctets-Bps
* Host interface	127.0.0.1 : 161
* SNMP OID	.1.3.6.1.2.1.2.2.1.10.3
Context name	
Security name	
Security level	authPriv
Authentication protocol	MD5 SHA
Authentication passphrase	
Privacy protocol	DES AES
Privacy passphrase	
Port	161
Type of information	Numeric (float)

Все обязательные поля ввода отмечены красной звёздочкой.

Теперь сохраните элемент данных и перейдите в *Мониторинг → Последние данные*, чтобы увидеть ваши данные SNMP!

Обратите внимание на специфичные опции доступные только для SNMPv3 элементов данных:

Параметр	Описание
Имя контекста	Введите контекстное имя для определения элемента данных в SNMP подсети. Имя контекста поддерживается для SNMPv3 элементов данных с Zabbix 2.2.
Имя безопасности	В данном поле раскрываются пользовательские макросы. Введите имя безопасности. В данном поле раскрываются пользовательские макросы.
Уровень безопасности	Выберете уровень безопасности: noAuthNoPriv - ни аутентификация, ни протокол безопасности не используются AuthNoPriv - используется протокол аутентификации, протокол безопасности нет AuthPriv - используются и протокол аутентификации, и протокол безопасности

Параметр	Описание
<i>Протокол аутентификации</i>	Выберете протокол аутентификации - <i>MD5</i> или <i>SHA</i> .
<i>Фраза-пароль аутентификации</i>	Введите фразу-пароль для аутентификации
<i>Протокол безопасности</i>	В данном поле раскрываются пользовательские макросы.
<i>Фраза-пароль безопасности</i>	Введите протокол безопасности - <i>DES</i> или <i>AES</i> .
	Введите фразу-пароль безопасности.
	В данном поле раскрываются пользовательские макросы.
	В случае некорректных учётных данных SNMPv3 (имя безопасности, протокол/фраза-пароль аутентификации, протокол безопасности) Zabbix получит ERROR от net-snmp, за исключением ошибочного <i>Фразы-пароль безопасности</i> , в этом случае Zabbix получит ошибку ВРЕМЕНИОЖИДАНИЯ от net-snmp.

При изменениях в *Протокол аутентификации*, *Фраза-пароль аутентификации*, *Протокол безопасности* или *Фраза-пароль безопасности*, чтобы эти изменения применились, необходимо перезапустить сервер/прокси.

Пример 1

Общий пример:

Параметр	Описание
Community public	
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>	
	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:

```
shell> snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

Мониторинг SNMP параметров возможен, если указан флаг --with-net-snmp при конфигурировании исходных кодов Zabbix.

Пример 2

Мониторинг времени работы:

Параметр	Описание
Community	public
Oid	MIB::sysUpTime.0
Ключ	router.uptime
Тип информации	Числовой (с плавающей точкой)
Единица измерения	uptime
Множитель	0.01

Обработка массовых SNMP запросов

Начиная с 2.2.3 Zabbix сервер и прокси одним опросом запрашивают множество SNMP элементов данных. Такое поведение затрагивает следующие типы SNMP элементов данных:

- обычные SNMP элементы данных;
- [SNMP элементы данных с динамическими индексами](#);
- [Правила низкоуровневого SNMP обнаружения](#).

Все элементы данных SNMP с одного интерфейса запланированы на опрос в одно время. Первые два типа элементов данных собираются поллерами порциями не более чем по 128 элементов данных, в то время как правила низкоуровневого обнаружения обрабатываются индивидуально как и ранее.

На низком уровне, есть два вида операций выполняемых при опросе значений: получение нескольких заданных объектов и прохождение дерева OID-ов.

Для “получения” используется GetRequest-PDU с не более чем 128 привязанных переменных. Для “прохождения”, используется GetNextRequest-PDU для SNMPv1 и GetBulkRequest с полем “max-repetitions” с наибольшим количеством в 128 полученных значений используется для SNMPv2 и SNMPv3.

Таким образом преимущества массовой обработки для каждого типа SNMP элемента данных описаны ниже:

- простые SNMP элементы данных получают преимущество от улучшения “получения”;
- SNMP элементы данных с динамическими индексами получают преимущество и от улучшений “получения” и “прохождения”: “получение” используется для проверки индексов, а “прохождение” для построения кэша значений;
- правила низкоуровневого SNMP обнаружения получают преимущество от улучшения “прохождения”.

Тем не менее, есть техническая проблема что не все устройства способны вернуть 128 значений за один запрос. Некоторые всегда возвращают корректный ответ, но другие либо отвечают с ошибкой “tooBig(1)”, либо не отвечают вообще, когда потенциальный запрос превышает определенный лимит.

Для вычисления оптимального количества запрашиваемых объектов с устройства, Zabbix использует следующую стратегию. Начинается с осторожного запроса одного значения. Это запрос выполнен успешно, запрашивается 2 значения за один запрос. Если запрос снова выполнен успешно, запрашивается 3 значения за запрос и продолжается аналогично умножением количества запрашиваемых значений на 1.5, в результате получается следующая последовательность размера запросов: 1, 2, 3, 4, 6, 9, 13, 19, 28, 42, 63, 94, 128.

Однако если устройство отказывается от ответа на определенный запрос (к примеру, 42 переменных), Zabbix делает 2 вещи.

Первое, для текущей серии элементов данных Zabbix делит пополам количество элементов данных за один запрос и запрашивает 21 переменных. Если устройство доступно, далее запросы должны работать в большинстве случаев, потому что известно что 28 переменных забиралось, а 21 значительно меньше. Тем не менее если проблема с запросами продолжается, Zabbix уменьшает количество запросов последовательно согласно этому алгоритму. Если и далее проблемы с запросами все еще актуальны, значит устройство определенно не отвечает и количество запросов это корень проблемы.

Второе дело, которое делает Zabbix для дальнейших порций элементов данных - это, начиная с последнего удачного количества переменных (28 в нашем случае), продолжает увеличивать количество переменных за запрос на 1 до достижения лимита. Например, предположим что максимально возможное количество запросов для данного устройства это 32, последующие запросы будут следующими 29,30,31,32 и 33. Последний запрос будет неудачным и Zabbix никогда более не запросит 33 значения за один запрос. С этого момента, Zabbix всегда будет запрашивать 32 значения для этого устройства.

Если большие запросы неудачно завершаются с определенным количеством переменных, это может означать одно из двух. Точный критерий по которому устройство может ограничивать запросы неизвестен, но мы можем приблизительно рассчитать количество переменных. Первая вероятность -

что количество значений примерно равно действительному лимиту размера для данного устройства в общем случае: иногда запросов либо меньше чем лимит, иногда больше. Вторая вероятность, что UDP пакет был потерян. В этом случае, если Zabbix сталкивается с неудачным запросом, он уменьшает максимальное количество запрашиваемых значение за запрос для попытки получения с устройства корректного диапазона, но (начиная с 2.2.8) только до 2 раз.

В примере выше, если запрос с 32 переменными будет неудачен, Zabbix уменьшит количество до 31. Если неудача случиться снова, Zabbix уменьшит количество до 30. Тем не менее, Zabbix не будет уменьшать количество ниже 30, потому что он предположит, что следующие проблемы по причине потерянных UDP пакетов, чем скорее ограничение устройства.

Если, однако, устройство не может обрабатывать массовые запросы корректно и по другим причинам, начиная с Zabbix 2.4 имеется настройка “Использовать массовые запросы” у каждого интерфейса, которая позволяет отключить массовые запросы у этого устройства.

7.2.2.1 Динамические индексы

Обзор

Хотя вы можете быть и найдете требуемый номер индекса (например, сетевого интерфейса) из всех SNMP OID'ов, иногда вы не можете полностью положиться на номер индекса 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

Для OID используется специальный синтаксис:

```
<OID данных>["index", "<базовый OID индекса>", "<искомая строка>"]
```

Параметр	Описание
OID данных	Основной OID, который используется для получения данных по элементу данных.
index	Метод обработки. В настоящее время поддерживается только один метод: index – поиск индекса и присоединение его к OID'у данных
базовый OID индекса	Этот OID будет просматриваться для получения значения индекса соответствующего заданной строке.
искомая строка	Строка используемая для точного совпадения со значением при выполнении поиска. Регистрозависимо.

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 просматриваться следующим образом:

```
...
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 присоединен к OID'у данных, чтобы получить интересуемое 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 ссылается на тот же самый базовый OID - Zabbix будет искать индекс в кэше, вместо еще одного запроса к наблюдаемому устройству. Обратите внимание, 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 перстраивается - каждый поллер, который встречает измененный индекс снова проходит и кэширует таблицу индексов SNMP.

7.2.2.2 Специальные OID'ы

Некоторые OID, наиболее часто используемые в SNMP, автоматически преобразовываются в числовые представления самим Zabbix. Например, **ifIndex** преобразовывается в **1.3.6.1.2.1.2.2.1.1**, **ifIndex.0** преобразовывается в **1.3.6.1.2.1.2.2.1.1.0**.

В таблице содержится перечень специальных OID.

Специальный OID	Идентификатор	Описание
ifIndex	1.3.6.1.2.1.2.2.1.1	Уникальное значение каждого интерфейса.
ifDescr	1.3.6.1.2.1.2.2.1.2	Текстовая строка, которая содержит информацию о интерфейсе. Эта строка может включать в себя название компании производителя, имя продукта, аппаратную версию 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 байтах
ifSpeed	1.3.6.1.2.1.2.2.1.5	Оценочная скорость интерфейса текущей пропускной способности 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	Полное число полученных байтов, включая символы заголовков.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	Количество пакетов одноадресной рассылки, доставленных

Специальный OID	Идентификатор	Описание
		на верхний уровень стека сетевых протоколов.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	Количество пакетов НЕ одноадресной рассылки (broadcast и multicast), доставленных на верхний уровень стека сетевых протоколов.
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 на верхний уровень стека сетевых протоколов. Одна 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 на верхний уровень стека сетевых протоколов.
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15	Количество пакетов, полученных через интерфейс, но отвергнутых из-за неизвестного или не поддерживаемого протокола.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	Полное количество отправленных октетов с интерфейса, включая символы заголовков.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	Полное количество пакетов, которые пытались отправить верхний уровень стека сетевых протоколов, и которые адресованы НЕ на broadcast или multicast адреса на этом субуровне, включая те которые были отвергнуты или не отправлены.
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	Полное количество пакетов, которые пытались отправить верхний уровень стека сетевых протоколов, и которые адресованы на broadcast или multicast адреса на этом субуровне, 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. Одна 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 могут быть отправлены из-за ошибок.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	Длина очереди исходящих пакетов (в пакетах).

7.2.2.3 MIB файлы

Введение

MIB обозначает Базу Управляющей Информации. MIB файлы позволяют вам использовать текстовое представление OID'a (Идентификатор Объекта).

Например,

`ifHCOutOctets`

является текстовым представлением OID'a

`1.3.6.1.2.1.31.1.1.1.10`

Вы можете использовать оба варианта при мониторинге SNMP устройств с Zabbix, но если вам удобнее пользоваться текстовым представлением, вам нужно установить MIB файлы.

Установка MIB файлов

На системах на основе Debian:

```
# apt install snmp-mibs-downloader  
# download-mibs
```

На системах на основе RedHat:

```
# yum install net-snmp-libs
```

Включение MIB файлов

На системах на основе RedHat mib файлы должны быть включены по умолчанию. На системах на основе Debian вам нужно отредактировать файл `/etc/snmp/snmp.conf` и раскомментировать 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 reenable  
# loading them by commenting out the following line.  
#mibs :
```

Тестирование MIB файлов

Тестирование snmp MIB можно выполнить с использованием `snmpwalk` утилиты. Если у вас эта утилита не установлена, вы можете использовать следующие инструкции.

На системах на основе Debian:

```
# apt install snmp
```

На системах на основе RedHat:

```
# yum install net-snmp-utils
```

После чего следующая команда не выдаст ошибку при выполнении запроса к сетевому устройству:

```
$ snmpwalk -v 2c -c public <IP СЕТЕВОГО УСТРОЙСТВА> 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 в Zabbix

Самое главное - иметь в виду, что процессы Zabbix не знают об изменениях, которые сделаны в MIB файлы. Поэтому после каждого изменения вам необходимо перезапустить Zabbix сервер или прокси, например:

```
# service zabbix-server restart
```

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.

Скажем, если вы хотите использовать [CISCO-SMI](#) MIB файл. Следующие инструкции загрузят и установят 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/local/share/snmp/mibs" >> /etc/snmp/snmp.conf
# cp /tmp/CISCO-SMI.my /usr/local/share/snmp/mibs
```

Теперь вы сможете 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 получаете 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.



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.

7.2.2.3 SNMP трапы

Обзор

Получение SNMP трапов является полной противоположностью запросам к SNMP устройствам.

В этом случае информация отправляется с SNMP устройства и собирается или “ловится” Zabbix'ом.

Обычно трапы отправляются при наступлении некоторых условий и агент подключается на 162 порт сервера (в отличии от 161 порта на стороне агента, который используется для запросов).

Использование трапов может помочь обнаружить некоторые кратковременные проблемы, которые происходят между интервалами опроса и могут быть пропущены при запросах данных.

Получение SNMP трапов в Zabbix рассчитано на работу с **snmptrapd** и с одним из встроенных механизмов передачи трапов в Zabbix - либо perl скрипт, либо SNMPTT.

Последовательность действий при получении трапа:

1. **snmptrapd** получает трап
2. snmptrapd передает трап в SNMPTT или вызывает получателя трапов Perl
3. SNMPTT или получатель трапов Perl, форматируют и записывают трап в файл
4. Zabbix SNMP траппер читает и анализирует файл с трапами
5. Zabbix ищет все соответствующие элементы данных с типом “SNMP трап” на интерфейсах узлов сети по каждому трапу, которые совпадают с полученным адресом из трапа. Возьмите на заметку, только выбранный “IP” или “DNS” у интерфейса узла сети используется в процессе поиска совпадения.
6. По каждому найденному SNMP интерфейсу, трап сравнивается со всеми регулярными выражениями из элементов данных “snmptrap[регулярное выражение]”. Если совпадение найдено, трап записывается значением для **всех** совпавших элементов данных. Если совпадений не найдено, но существует элемент данных “snmptrap.fallback”, трап записывается значением для этого элемента данных.
7. Если совпадений не было найдено ни с одним из соответствующих SNMP интерфейсов, Zabbix по умолчанию журнализирует несовпадающие трапы. (Это поведение настраивается в “Журналировать не совпадающие SNMP трапы” в Администрирование → Общие → Прочие).

1 Настройка SNMP трапов

Настройка следующих полей в веб-интерфейсе является специфичной для этого типа элементов данных:

- Ваш узел сети должен иметь SNMP интерфейс

В **Настройка → Узлы сети**, в поле **Интерфейсы узла сети** добавьте SNMP интерфейс с корректными IP или DNS адресами. Адрес из каждого полученного трапа сравнивается с IP и DNS адресами всех SNMP интерфейсов для поиска соответствующих узлов сети.

- Настройка элемента данных

В поле **Ключ** используйте один из ключей SNMP трапов:

Описание	Ключ	Возвращаемое значение	Комментарии
Отлов всех SNMP трапов , который совпадают с регулярным выражением , указанном в поле регулярное выражение . Если регулярное выражение не указано, принимаются все трапы.	snmptrap[регулярное выражение]	SNMP трап	Этот элемент данных можно привязать только к SNMP интерфейсам. Элемент данных поддерживается начиная с Zabbix 2.0.0 . <i>Обратите внимание:</i> Начиная с Zabbix 2.0.5, в параметре этого ключа элемента данных поддерживаются пользовательские макросы и глобальные регулярные выражения.
Отлов всех SNMP трапы, которые не совпадают ни с одним из элементов данных <code>snmptrap[]</code> для этого интерфейса. Многострочное совпадение по регулярному выражению в данный момент времени не поддерживается.	snmptrap.fallback	SNMP трап	Этот элемент данных можно привязать только к SNMP интерфейсам. Элемент данных поддерживается начиная с Zabbix 2.0.0 .

Укажите **Тип информации** равным 'Журнал (лог)' для обработки штампов времени. Обратите внимание, что другие форматы, такие как 'Числовой' также приемлемы, но для этого может потребоваться пользовательский обработчик трапов.

Для того чтобы мониторинг SNMP трапов работал, он должен быть сначала корректно настроен.

2 Настройка мониторинга SNMP трапов

Настройка Zabbix сервера/прокси

Для чтения трапов, Zabbix сервер или прокси должны быть настроены на запуск процесса SNMP трассера и должны знать абсолютный путь к файлу с трапами, который заполняется при помощи SNMPTT или получателя трапов Perl. Чтобы это сделать, измените файл конфигурации ([zabbix_server.conf](#) или [zabbix_proxy.conf](#)):

1. StartSNMPTrapper=1
2. SNMPTrapperFile=[ФАЙЛ С ТРАПАМИ]

Если используется systemd параметр **PrivateTmp**, этот файл вряд ли заработает в `/tmp`.

Настройка SNMPTT

Для начала, snmptrapd необходимо настроить на использование SNMPTT.

Для лучшей производительности, SNMPTT необходимо сконфигурировать демоном с использованием **snmpthandler-embedded** для передачи ему трапов. Смотрите инструкции по настройке SNMPTT на его сайте:

<http://snmptt.sourceforge.net/docs/snmptt.shtml>

При конфигурировании SNMPTT на получение трапов, настройте SNMPTT на журналирование этих трапов:

1. журналирование трапов в файл с трапами, который Zabbix будет читать:

```
log_enable = 1  
log_file = [ФАЙЛ С ТРАПАМИ]
```

2. задайте формат времени/даты:

```
date_time_format = %H:%M:%S %Y/%m/%d = [ФОРМАТ ВРЕМЕНИ]
```

Теперь отформатируйте трапы, чтобы они распознавались Zabbix'ом (измените 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 траповсмотрите ниже.

Не используйте неизвестные трапы - Zabbix может их не распознать. Неизвестные трапы могут быть обработаны, если задать общее событие в snmptt.conf:

```
EVENT general .* "General event" Normal
```

Настройка получателя Perl трапов

Требования: Perl, Net-SNMP скомпилированный с --enable-embedded-perl (компилируется по умолчанию начиная с Net-SNMP 5.4)

Получатель трапов Perl (ищите в misc/snmptrap/zabbix_trap_receiver.pl) можно использовать для передачи трапов в Zabbix сервер напрямую с snmptrapd. Для его настройки:

- добавьте perl скрипт в файл конфигурации snmptrapd (snmptrapd.conf), Например:
perl do "[ПОЛНЫЙ ПУТЬ К СКРИПТУ ПОЛУЧАТЕЛЯ PERL]";
- настройте сам получатель, например:
\$SNMPTrapperFile = '[ФАЙЛ С ТРАПАМИ]';
\$DateTimeFormat = '[ФОРМАТ ДАТЫ/ВРЕМЕНИ]';

Если имя скрипта не заключено в кавычки, snmptrapd откажется запускаться с сообщениями наподобие этих:

```
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
```

Формат SNMP трапа

Все пользовательские получатели трапов perl и конфигурация SNMPTT трапов должны форматировать трап следующим образом:

[штамп времени] [трап, часть 1] ZBXTRAP [адрес] [трап, часть 2], где

- [штамп времени] - штамп времени используемый в элементах данных типа 'Журнал (лог)'

- ZBXTRAP - заголовок, который указывает, что с этой строки начался новый трап
- [адрес] - IP адрес, используемый для поиска узла сети для этого трапа

Обратите внимание, что “ZBXTRAP” и “[адрес]” при обработке отрезаются из сообщения. Если трап форматируется как-то иначе, Zabbix может разобрать такие трапы неожиданным образом.

Пример трапа:

11:30:15 2011/07/27 .1.3.6.1.6.3.1.1.5.3 Normal “Status Events” localhost - ZBXTRAP 192.168.1.1 Link down on interface 2. Admin state: 1. Operational state: 2

Такое сообщение будет результатом следующего трапа для SNMP интерфейса с IP=192.168.1.1:

11:30:15 2011/07/27 .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

3 Требования к системе

Поддержка больших файлов

У Zabbix имеется “Поддержка больших файлов” для файлов SNMP трапов. Максимальный размер файла, который Zabbix может прочитать, это 2^{63} (8 Эбайт). Обратите внимание, что файловая система может иметь меньшее ограничение на максимальный размер файлов.

Ротация журнала

Zabbix не предоставляет какую-либо систему ротации журналов - должно быть обработано пользователем. Ротация журналов должна начинаться с переименования старого файла и только после этого удалять его, чтобы никакие трапы не пропали:

1. Zabbix открывает файл с трапами с последней известной позиции и переходит к 3 шагу.
2. Zabbix проверяет был ли сротирован в данный момент открытый файл, сравнивая номера inode с заданным у файла трапов номером inode. Если нет открытого файла, Zabbix сбрасывает последнюю позицию и переходит к 1 шагу.
3. Zabbix читает данные из открытого в данный момент файла и устанавливает новую позицию.
4. Обрабатываются новые данные. Если этот файл был ротирован, то он закрывается и Zabbix переходит назад ко 2 шагу.
5. Если не было новых данных, Zabbix засыпает на 1 секунду и возвращается ко 2 шагу.

Файловая система

Из-за реализации файла с трапами, Zabbix'у требуется файловая система с поддержкой inode для того чтобы различать файлы (эта информация берется из вызова stat()).

4 Пример установки

Этот пример использует snmptrapd + SNMPTT для передачи трапов Zabbix серверу. Установка:

1. **zabbix_server.conf** - настройте Zabbix, чтобы запускался SNMP траппер процесс и укажите абсолютный путь к файлу с трапами:

```
StartSNMPTrapper=1
SNMPTrapperFile=/tmp/my_zabbix_traps.tmp
```
2. **snmptrapd.conf** - добавьте SNMPTT как обработчик трапов:

```
traphandle default snmptt
```
3. **snmptt.ini** - настройте выходной файл и формат времени:

```
log_file = /tmp/my_zabbix_traps.tmp
date_time_format = %H:%M:%S %Y/%m/%d
```
4. **snmptt.conf** - укажите формат трапов по умолчанию:

```
EVENT general .* "General event" Normal
FORMAT ZBXTRAP $aA $ar
```

5. Создайте ТЕСТ элемент данных с типом SNMP трап:

Узел сети с IP адресом SNMP: 127.0.0.1

Ключ: snmptrap["General"]

Формат времени журнала: hh:mm:ss yyyy/MM/dd

В результате:

1. Используйте следующую команду для отправки трапа:

```
snmptrap -v 1 -c public 127.0.0.1 '1.3.6.1.6.3.1.1.5.3' '0.0.0.0' 6 33 '55' .1.3.6.1.6.3.1.1.5.3 s  
"teststring000"
```

2. Полученный трап:

```
15:48:18 2011/07/26 .1.3.6.1.6.3.1.1.5.3.0.33 Normal "General event" localhost - ZBXTRAP  
127.0.0.1 127.0.0.1
```

3. Значение ТЕСТ элемента данных:

```
15:48:18 2011/07/26 .1.3.6.1.6.3.1.1.5.3.0.33 Normal "General event" localhost - 127.0.0.1
```

В этом примере используется SNMPTT как **traphandle**. Для лучшей производительности на продуктивных системах используйте встроенный Perl для передачи трапов от snmptrapd к SNMPTT или напрямую Zabbix'у.

5 Смотрите также

- [Руководство по настройке SNMP трапов на CentOS на сайте zabbix.org \[en\]](#)

7.2.2.4 Проверки IPMI

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 управления “lights-out” или “out-of-band” компьютерными системами. Он позволяет наблюдать за состоянием аппаратного обеспечения 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.).

Начиная с Zabbix 3.4, добавлен новый процесс IPMI менеджер, который выполняет распределение проверок IPMI между IPMI поллерами. Теперь, узел сети всегда опрашивается одним и тем же IPMI поллером, таким образом снижается количество открытых подключений к BMC контроллерам. 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. Необходимо добавить интерфейс IPMI, с соответствующими IP адресом и номером порта, и задать параметры аутентификации IPMI.

Смотрите [настройку узлов сети](#) для получения более подробных сведений.

Настройка сервера

По умолчанию, Zabbix сервер не запускает IPMI поллеры, таким образом 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 следующей:

```
StartIPMIPollers=3
```

Сохраните файл и затем перезапустите zabbix_server.

Настройка элемента данных

Для [настройки элемента данных](#) на уровне узла сети:

- Для *Интерфейса узла сети* выберите IPMI IP и порт
- Выберите 'IPMI агент' как *Tun*
- Укажите *IPMI датчик* (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.
- Введите [ключ](#) элемента данных, уникальный в пределах узла сети (скажем, ipmi.fan.rpm)
- Выберите соответствующий тип информации ('Числовой (с плавающей точкой)' в этом случае, для дискретных датчиков - 'Числовой (целый)'), единицы измерения (скорее всего 'rpm') и любые другие требуемые атрибуты элемента данных.

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 выполняются от Zabbix к конкретному BMC в течении более чем времени ожидания сессии, настроенное в BMC, тогда следующая проверка IPMI после того, как истечет время ожидания, приведет к отдельным сообщениям о превышении времени ожидания, попыток или ошибке при получении. После того, как откроется новая сессия и сделано полное повторное сканирование BMC. Если вы хотите избежать лишних сканирований BMC, рекомендуется установить интервал опроса IPMI элементов данных ниже времени ожидания неактивности IPMI сессии, которое настраивается в BMC.

Примечания о дискретных датчиках IPMI

Для поиска датчиков на узле сети запустите Zabbix сервер с включенным **DebugLevel=4**. Подождите пару минут и найдите 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:0x3 ('discrete_state') type:0x7 ('processor')
full_name:'(r0.32.3.0).CATERR'
```

```

8358:20130318:111122.170 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:15
id:'CPU Therm Trip' reading_type:0x3 ('discrete_state') type:0x1 ('temperature')
full_name:'(7.1).CPU Therm Trip'
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17
id:'System Event Log' reading_type:0x6f ('sensor specific') type:0x10
('event_logging_disabled') full_name:'(7.1).System Event Log'
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17
id:'PhysicalSecurity' reading_type:0x6f ('sensor specific') type:0x5
('physical_security') full_name:'(23.1).PhysicalSecurity'
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14
id:'IPMI Watchdog' reading_type:0x6f ('sensor specific') type:0x23 ('watchdog_2')
full_name:'(7.7).IPMI Watchdog'
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16
id:'Power Unit Stat' reading_type:0x6f ('sensor specific') type:0x9 ('power_unit')
full_name:'(21.1).Power Unit Stat'
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16
id:'P1 Therm Ctrl %' reading_type:0x1 ('threshold') type:0x1 ('temperature')
full_name:'(3.1).P1 Therm Ctrl %'
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16
id:'P1 Therm Margin' reading_type:0x1 ('threshold') type:0x1 ('temperature')
full_name:'(3.2).P1 Therm Margin'
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:13
id:'System Fan 2' reading_type:0x1 ('threshold') type:0x4 ('fan')
full_name:'(29.1).System Fan 2'
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:13
id:'System Fan 3' reading_type:0x1 ('threshold') type:0x4 ('fan')
full_name:'(29.1).System Fan 3'
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14
id:'P1 Mem Margin' reading_type:0x1 ('threshold') type:0x1 ('temperature')
full_name:'(7.6).P1 Mem Margin'
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17
id:'Front Panel Temp' reading_type:0x1 ('threshold') type:0x1 ('temperature')
full_name:'(7.6).Front Panel Temp'
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:15
id:'Baseboard Temp' reading_type:0x1 ('threshold') type:0x1 ('temperature')
full_name:'(7.6).Baseboard Temp'
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:9 id:'BB
+5.0V' reading_type:0x1 ('threshold') type:0x2 ('voltage') full_name:'(7.1).BB +5.0V'
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14
id:'BB +3.3V STBY' reading_type:0x1 ('threshold') type:0x2 ('voltage')
full_name:'(7.1).BB +3.3V STBY'
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:9 id:'BB
+3.3V' reading_type:0x1 ('threshold') type:0x2 ('voltage') full_name:'(7.1).BB +3.3V'
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17
id:'BB +1.5V P1 DDR3' reading_type:0x1 ('threshold') type:0x2 ('voltage')
full_name:'(7.1).BB +1.5V P1 DDR3'
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17
id:'BB +1.1V P1 Vccp' reading_type:0x1 ('threshold') type:0x2 ('voltage')
full_name:'(7.1).BB +1.1V P1 Vccp'
8358:20130318:111122.174 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14
id:'BB +1.05V PCH' reading_type:0x1 ('threshold') type:0x2 ('voltage')
full_name:'(7.1).BB +1.05V PCH'

```

Для расшифровки типов датчиков IPMI и их состояния, скачайте экземпляр спецификаций IPMI 2.0 с <http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-specifications.html> (Во время подготовки документации новейшим документом был <http://www.intel.com/content/dam/www/public/us/en/documents/product-briefs/second-gen-interface-spec-v2.pdf>)

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” - температурный датчик, “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 считываемые типы кодов 02h-0Ch это “Общий Дискретный” датчик. Дискретные датчики имеют до 15 возможных состояний (другими словами - до 15 значимых бит). К примеру, для датчика 'CATERR' с “type:0x7” “Table 42-3, Sensor Type Codes” показывает, что этот тип обозначает “Процессор” и значение отдельных бит: 00h (наименьший значимый бит) - IERR(внутренняя ошибка процессора), 01h - Перегрев процессора и т.д.

Есть в нашем примере 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 бит. Например, датчик 'Power Unit Stat' имеет тип “type:0x9”, который означает “Блок питания”. Смещение 00h означает “Выключено/Обесточено”. Другими словами, если младший значимый бит равен 1, то сервер выключен. Для проверки этого бита можно воспользоваться функция **band** с маской 1. Выражение триггера может выглядеть следующим образом

```
{www.zabbix.com:Power Unit Stat.band(#1,1)}=1
```

для предупреждения о выключенном сервере.

Заметки о 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), присоединенный к концу 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 следующее именование “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 обновлен в новом Linux дистрибутиве, который использует OpenIPMI-2.0.19 (или более позднюю), элементы данных с такими IPMI дискретными датчиками перейдут в “NOT SUPPORTED”. Вам потребуется изменить их имена *IPMI датчик* (удалить с конца '0') и подождать некоторое время пока они станут “Активированными” снова.

Заметки о одновременной доступности пороговых и дискретных датчиках

Некоторые IPMI агенты предоставляют одновременно пороговые и дискретные датчики под одним именем. В версиях Zabbix до 2.2.8, выбирался первый полученный датчик. Начиная с версии 2.2.8, предпочтение всегда отдается 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/удален, узел сети в обслуживании и так далее), то в этом случае Zabbix сервер/прокси продолжит сбор данных по IPMI с этого узла сети до перезагрузки сервера/прокси.

7.2.2.5 Простые проверки

Обзор

Простые проверки в основном используются для удаленных безагентных проверок сервисов.

Обратите внимание, что для простых проверок Zabbix агент не требуется. За обработку (созданием внешних подключений и т.д.) простых проверок 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]
```

Поля *Имя пользователя* и *пароль* в конфигурации простых элементов данных используются для элементов данных мониторинга VMware; иначе игнорируются.

Поддерживаемые простые проверки

Список поддерживаемых простых проверок:

Смотрите также:

- [Ключи элементов данных для мониторинга VMware](#)

▲	Описание	Возвращаемое значение	Ключ	Параметры	Комментарии
	Доступность хоста через пинг по ICMP.	0 - ошибка при пинге по ICMP 1 - успешный пинг по ICMP	icmpping[<цель>, <пакеты>, <интервал>, <размер>, <время ожидания>]	цель - IP хоста или DNS имя пакеты - количество пакетов интервал - время между успешными пакетами в миллисекундах размер - размер пакета в байтах время ожидания - время ожидания в миллисекундах	Пример: ⇒ icmpping[4] → если по крайней мере один пакет из четырех вернется, элемент данных возвратит 1. Смотрите также таблицу со значениями по умолчанию .
	Процентное отношение потерянных пакетов.	Число с плавающей точкой.	icmppingloss[<цель>, <пакеты>, <интервал>, <размер>, <время ожидания>]	цель - IP хоста или DNS имя пакеты - количество пакетов интервал - время между успешными пакетами в миллисекундах размер - размер пакета в байтах время ожидания - время ожидания в миллисекундах	Смотрите также таблицу со значениями по умолчанию .
	Время ответа на пинг по ICMP (в секундах).	Число с плавающей точкой	icmppingsec[<цель>, <пакеты>, <интервал>, <размер>, <время ожидания>, <режим>]	цель - IP хоста или DNS имя пакеты -	Если хост недоступен (превышено время ожидания), элемент данных вернет 0.

Описание	Возвращаемое значение	Параметры	Комментарии
Проверка запущен ли сервис и отвечает ли на TCP подключения.	0 - сервис недоступен 1 - сервис работает	<p>количество пакетов</p> <p>интервал - время между успешными пакетами в миллисекундах</p> <p>размер - размер пакета в байтах</p> <p>время ожидания - время ожидания в миллисекундах</p> <p>режим - один из min, max, avg (по умолчанию)</p>	<p>Если элемент данных “icmppingsec” вернет значение меньше 0.0001 секунд, значение будет равно 0.0001 секунд.</p> <p>Смотрите также таблицу со значениями по умолчанию.</p>
Проверка производительности сервиса.	<p>net.tcp.service[сервис,<ip>,<порт>]</p> <p>Проверка запущен ли сервис и отвечает ли на TCP подключения.</p> <p>Число с плавающей точкой.</p> <p>0.000000 - сервис недоступен</p> <p>сек - количество секунд потребовавшихся для подключения к сервису</p>	<p>сервис - один из ssh, ntp, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet (смотри детали)</p> <p>ip - IP адрес или DNS имя (по умолчанию, используется IP/DNS узла сети)</p> <p>порт - номер порта (по умолчанию для сервиса используется стандартный номер порта).</p>	<p>Пример:</p> <p>⇒ net.tcp.service[ftp,,45] → можно использовать для проверки доступности FTP сервера на 45 порту TCP.</p> <p>Обратите внимание, для сервиса <i>tcp</i> обязательно нужно указывать порт.</p> <p>Эти проверки могут привести к дополнительным записям в системных лог файлах (обычно сессии SMTP и SSH журналируются).</p> <p>Проверка шифрованных протоколов (таких как IMAP на 993 порту или POP на 995 порту) в настоящее время не поддерживается. Как решение, пожалуйста, для подобных проверок используйте net.tcp.service[tcp,<ip>,порт]. Сервисы <i>https</i> и <i>telnet</i> поддерживаются Zabbix начиная с версии 2.0.</p> <p>net.tcp.service.perf[сервис,<ip>,<порт>]</p> <p>Число с плавающей точкой.</p> <p>0.000000 - сервис недоступен</p> <p>сек - количество секунд потребовавшихся для подключения к сервису</p>

Описание	Возвращаемое значение	Параметры	Комментарии
		порт - номер порта (по умолчанию для сервиса используется стандартный номер порта).	993 порту или POP на 995 порту) в настоящее время не поддерживается. Как решение, пожалуйста, для подобных проверок используйте <code>net.tcp.service.perf[tcp,<ip>,порт]</code> .
Проверка запущен ли сервис и отвечает ли на UDP подключения.	0 - сервис недоступен 1 - сервис работает	сервис - возможные значения: <i>ntp</i> (смотри детали) ip - IP адрес или DNS имя (по умолчанию, используется IP/DNS узла сети) порт - номер порта (по умолчанию для сервиса используется стандартный номер порта).	Сервисы <i>https</i> и <i>telnet</i> поддерживаются Zabbix начиная с версии 2.0. Назывался <code>tcp_perf</code> до Zabbix 2.0.
Проверка производительности UDP сервиса.	Число с плавающей точкой. 0.000000 - сервис недоступен секунды - количество секунд прошедшее на ожидания ответа от сервиса	сервис - возможные значения: <i>ntp</i> (смотри детали) ip - IP адрес или DNS имя (по умолчанию, используется IP/DNS узла сети) порт - номер порта (по умолчанию для сервиса используется стандартный номер порта).	Пример: ⇒ <code>net.udp.service[ntp,45]</code> → можно использовать для тестирования доступности NTP сервиса на 45 порту UDP. Этот элемент данных поддерживается начиная с Zabbix 3.0, но <i>ntp</i> сервис был доступен в <code>net.tcp.service[]</code> элементе данных и в предыдущих версиях.

Обработка времени ожидания

Zabbix не будет обрабатывать простую проверку дольше Timeout (времени ожидания) секунд, заданных в файле конфигурации Zabbix сервера/прокси.

ICMP пишет

Для обработки ICMP пинг Zabbix использует внешнюю утилиту **fping**.

Эта утилита не является частью дистрибутива Zabbix и должна быть установлена дополнительно. Если утилиты нет, у нее выставлены неверные разрешения и её размещение не совпадает с размещением заданным в файле конфигурации Zabbix сервера/прокси (параметры 'FpingLocation'), ICMP пинг (**icmpping**, **icmppingloss**, **icmppingsec**) не будет обрабатываться.

Смотрите также: [известные проблемы](#)

fping должен быть выполняемым под пользователем Zabbix демонов и должен иметь setuid root. Выполните эти команды из под **root** для выставления корректных разрешений:

```
shell> chown root:zabbix /usr/sbin/fping
shell> chmod 4710 /usr/sbin/fping
```

После выполнения этих двух команд выше проверьте владельца исполняемого файла **fping**. В некоторых случаях владелец может сбрасываться при выполнении chmod команды.

Также проверьте, принадлежит ли пользователь zabbix к группе zabbix, запустив команду:

```
shell> groups zabbix
```

и если нет добавьте следующей командой:

```
shell> usermod -a -G zabbix zabbix
```

Значения по умолчанию, ограничения и описания значений для параметров ICMP проверок:

Параметр	Ед. изм	Описание	Флаг у fping	Значения по умолчанию		Разрешенные ограничения в Zabbix	
				у fping	Zabbix	мин	макс
пакеты	количество	количество требуемых пакетов к цели	-C			3	1 10000
интервал	миллисекунды	время ожидания между успешными пакетами	-p	1000		20	без ограничений
размер	байты	размер пакета в байтах 56 байт в x86, 68 байт в x86_64	-b	56 или 68		24	65507
время ожидания	миллисекунды	fping v3.x - время ожидания после отправки последнего пакета (влияет с "-C" флагом) fping v4.x - отдельное время ожидания для каждого пакета	-t	fping v3.x - 500 fping v4.x - наследуется от -p флага, но не более чем 2000		50	без ограничений

Предупреждение: Значения по умолчанию для **fping** могут различаться в зависимости от платформы и версии - если сомневаетесь, проверьте документацию по **fping**.

Zabbix записывает проверяемые IP адреса во временный файл по всем трем *icmpping** ключам, который затем передается утилите **fping**. Если элементы данных имеют различные параметры ключа, то только элементы данных с идентичными параметрами ключа записываются в один файл.

Все записанные в один файл IP адреса проверяются fping утилитой в параллельном режиме, таким образом процесс Zabbix icmp pinger тратит фиксированное время вне зависимости от количества IP адресов в файле.

7.2.2.5.1 Ключи элементов данных для мониторинга VMware

Ключи элементов данных

Таблица содержит подробные сведения о простых проверках, которые можно использовать для мониторинга VMware сред.

Описание	Возвращаемое значение	Параметры	Комментарии
Обнаружение кластеров VMware.	JSON объект	url - URL службы VMware	vmware.cluster.discovery[<url>]
Состояние кластера VMware.	Целое число 0 - серый; 1 - зеленый; 2 - желтый; 3 - красный.	url - URL службы VMware имя - имя кластера VMware	vmware.cluster.status[<url>, <имя>]
Журнал событий VMware.	Лог (журнал)	url - URL службы VMware	vmware.eventlog[<url>]
Полное имя сервиса VMware.	Строка	url - URL службы VMware	vmware.fullname[<url>]
Имя кластера гипервизора VMware.	Строка	url - URL службы VMware uuid - имя хоста гипервизора VMware	vmware.hv.cluster.name[<url>,<uuid>]
Использование процессора гипервизором VMware (Гц).	Целое число	url - URL службы VMware uuid - имя хоста гипервизора VMware	vmware.hv.cpu.usage[<url>,<uuid>]
Имя центра обработки данных гипервизора VMware.	Строка	url - URL службы VMware uuid - имя хоста гипервизора VMware	vmware.hv.datacenter.name[<url>,<uuid>]
Обнаружение хранилищ у гипервизора VMware	JSON объект	url - URL службы VMware uuid - имя хоста гипервизора VMware	vmware.hv.datastore.discovery[<url>,<uuid>]
Среднее время операций чтения с хранилища данных (в Целое число ² миллисекундах)		url - URL службы VMware uuid - имя хоста гипервизора VMware хранилище - имя хранилища	vmware.hv.datastore.read[<url>,<uuid>,<хранилище>,<режим>]

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
		режим - задержка (по умолчанию)	
	vmware.hv.datastore.size[<url>,<uuid>,<ханилище>,<режим>]	url - URL службы VMware uuid - имя хоста гипервизора VMware ханилище - имя хранилища режим - возможные значения: total (по умолчанию), free, pfree (свободного места, в процентах), uncommitted	
Размер пространства VMware хранилища в байтах или в процентах от общего размера.	Целое число - для байт Число с плавающей точкой - для процентов		Доступно начиная с Zabbix версий 3.0.6, 3.2.2
	vmware.hv.datastore.write[<url>,<uuid>,<ханилище>,<режим>]	url - URL службы VMware uuid - имя хоста гипервизора VMware ханилище - имя хранилища режим - задержка (по умолчанию)	
Среднее время операций записи в хранилище данных (в Целое число ² миллисекундах)			
Обнаружение гипервизоров VMware.	JSON объект	url - URL службы VMware	
	vmware.hv.discovery[<url>]		
Имя гипервизора VMware.	Строка		
	vmware.hv.fullname[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста гипервизора VMware	
Частота процессора гипервизора VMware (Гц).	Целое число		
	vmware.hv.hw.cpu.freq[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста гипервизора VMware	
Модель процессора гипервизора VMware.	Строка		
	vmware.hv.hw.cpu.model[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста гипервизора VMware	
Количество ядер процессора у гипервизора VMware.	Целое число		
	vmware.hv.hw.cpu.num[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста гипервизора VMware	
Количество потоков процессора у гипервизора VMware.	Целое число		
	vmware.hv.hw.cpu.threads[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста гипервизора VMware	
Общее количество памяти	Целое число	url - URL службы VMware	
	vmware.hv.hw.memory[<url>,<uuid>]		

Ключ			
Описание	Возвращаемое значение	Параметры	Комментарии
гипервизора VMware (в байтах).		uuid - имя хоста гипервизора VMware vmware.hv.hw.model[<url>,<uuid>]	
Модель гипервизора VMware. Стока		url - URL службы VMware uuid - имя хоста гипервизора VMware	
UUID BIOS гипервизора VMware.	Строка	 vmware.hv.hw.uuid[<url>,<uuid>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware
Имя поставщика гипервизора VMware.	Строка	 vmware.hv.hw.vendor[<url>,<uuid>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware
Размер раздутой памяти гипервизора VMware (в байтах).	Целое число	 vmware.hv.memory.size.balloon[<url>,<uuid>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware
Размер используемой памяти гипервизором VMware (в байтах).	Целое число	 vmware.hv.memory.used[<url>,<uuid>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware
Статистика по входящему трафику на сетевом интерфейсе гипервизора VMware (в байтах в секунду).	Целое число 2	 vmware.hv.network.in[<url>,<uuid>,<mode>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware режим - bps
Статистика по исходящему трафику на сетевом интерфейсе гипервизора VMware (в байтах в секунду).	Целое число 2	 vmware.hv.network.out[<url>,<uuid>,<режим>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware режим - bps
Значение счетчика производительности гипервизора VMware.	Целое число 2	 vmware.hv.perfcounter[<url>,<uuid>,<путь>,<экземпляр>]	 url - URL службы VMware uuid - имя хоста гипервизора VMware путь - путь к счетчику производительности 1 экземпляр - экземпляр счетчика производительности. Оставьте пустым для получения агрегированных значений (по умолчанию)
			Доступно начиная с Zabbix 2.2.9, 2.4.4
		 vmware.hv.sensor.health.state[<url>,<uuid>]	

Ключ				
Описание	Возвращаемое значение	Параметры	Комментарии	
Датчик накопительного состояния гипервизора VMware.	Целое число: 0 - серый; 1 - зеленый; 2 - желтый; 3 - красный	url - URL службы VMware uuid - имя хоста гипервизора VMware	Доступно начиная с Zabbix 2.2.16, 3.0.6, 3.2.2	
Состояние гипервизора VMware.	Целое число: 0 - серый; 1 - зеленый; 2 - желтый; 3 - красный.	url - URL службы VMware uuid - имя хоста гипервизора VMware	Используется свойство общего состояния хост системы начиная с Zabbix 2.2.16, 3.0.6, 3.2.2	
Время работы гипервизора VMware (в секундах).	Целое число	url - URL службы VMware uuid - имя хоста гипервизора VMware		
Версия гипервизора VMware.	Строка	url - URL службы VMware uuid - имя хоста гипервизора VMware		
Количество виртуальных машин на гипервизоре VMware.	Целое число	url - URL службы VMware uuid - имя хоста гипервизора VMware		
Версия сервиса VMware.	Строка	url - URL службы VMware		
Имя виртуальной машины VMware.	Строка	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Количество процессоров на виртуальной машине VMware.	Целое число	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Время (в миллисекундах), когда виртуальная машина была готова, но не запланирована на выполнение на физическом CPU. Время готовности CPU зависит от количества виртуальных машин на хосте и их загрузки CPU (%).	Целое число ²	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	Доступно начиная с Zabbix версии 3.0.0	

Описание	Возвращаемое значение	Параметры	Комментарии
Использование процессора виртуальной машины VMware (Гц).	vmware.vm.cpu.usage[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Имя центра обработки данных виртуальной машины VMware.	vmware.vm.datacenter.name[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Обнаружение виртуальных машин VMware.	vmware.vm.discovery[<url>]	url - URL службы VMware	
Имя гипервизора виртуальной машины VMware.	vmware.vm.hv.name[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Общий размер памяти виртуальной машины VMware (в байтах).	vmware.vm.memory.size[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Размер раздутой памяти виртуальной машины VMware(в байтах).	vmware.vm.memory.size.balloon[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Размер сжатой памяти витруальной машины VMware (в байтах).	vmware.vm.memory.size.compressed[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Размер приватной памяти витруальной машины VMware (в байтах).	vmware.vm.memory.size.private[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Размер разделяемой памяти виртуальной машины VMware (в байтах).	vmware.vm.memory.size.shared[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Размер памяти выгруженной в файл подкачки витруальной	vmware.vm.memory.size.swapped[<url>,<uuid>]	url - URL службы VMware uuid - имя хоста	

Ключ				
Описание	Возвращаемое значение	Параметры	Комментарии	
машины VMware (в байтах).		виртуальной машины VMware		
Размер используемой памяти гостевой виртуальной машиной VMware (в байтах).	Целое число	vmware.vm.memory.size.usage.guest[<url>,<uuid>] url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Размер используемой памяти гостевой виртуальной машиной VMware (в байтах).	Целое число	vmware.vm.memory.size.usage.host[<url>,<uuid>] url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Обнаружение сетевых интерфейсов виртуальной машины VMware.	JSON объект	vmware.vm.net.if.discovery[<url>,<uuid>] url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Статистика по входящему трафику на сетевого интерфейсе виртуальной машины VMware (в байтах/пакетах в секунду).	Целое число ²	vmware.vm.net.if.in[<url>,<uuid>,<экземпляр>,<режим>] url - URL службы VMware uuid - имя хоста виртуальной машины VMware экземпляр - экземпляр сетевого интерфейса режим - bps (по умолчанию)/pps - байты/пакеты в секунду		
Статистика по исходящему трафику на сетевом интерфейсе виртуальной машины VMware (в байтах/пакетах в секунду).	Целое число ²	vmware.vm.net.if.out[<url>,<uuid>,<экземпляр>,<режим>] url - URL службы VMware uuid - имя хоста виртуальной машины VMware экземпляр - экземпляр сетевого интерфейса режим - bps (по умолчанию)/pps - байты/пакеты в секунду		
Значение счетчика производительности у виртуальной машины VMware.	Целое число ²	vmware.vm.perfcounter[<url>,<uuid>,<путь>,<экземпляр>] url - URL службы VMware uuid - имя хоста гипервизора VMware путь - путь к счетчику производительности ¹ экземпляр - экземпляр счетчика производительности. Оставьте пустым для получения	Доступно начиная с Zabbix 2.2.9, 2.4.4	

Ключ				
Описание	Возвращаемое значение	Параметры	Комментарии	
Состояние питания виртуальной машины VMware.	Целое число: 0 - выключено; 1 - включено; 2 - приостановлено.	агрегированных значений (по умолчанию)	url - URL службы VMware uuid - имя хоста виртуальной машины VMware	
Объем записанных данных на хранилище данных виртуальной машины VMware (в байтах).	Целое число	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Объем не записанных данных на хранилище данных виртуальной машины VMware (в байтах).	Целое число	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Размер неразделяемого пространства хранилища виртуальной машины VMware (в байтах).	Целое число	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Время непрерывной работы виртуальной машины VMware (в секундах).	Целое число	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Обнаружение дисковых устройств виртуальной машины VMware.	JSON объект	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		
Статистика чтения с дискового устройства виртуальной машины VMware (байт/операций в секунду).	Целое число ²	url - URL службы VMware uuid - имя хоста виртуальной машины VMware экземпляр - экземпляр дискового устройства режим - bps/ops - байт/операций в секунду		
Статистика записи на дисковое устройство виртуальной машины VMware (байт/операций в секунду).	Целое число ²	url - URL службы VMware uuid - имя хоста виртуальной машины VMware		

	Описание	Возвращаемое значение	Параметры	Комментарии
▲	Обнаружение файловых систем виртуальной машины VMware.	JSON объект	экземпляр - экземпляр дискового устройства режим - bps/ops - байт/операций в секунду	
		vmware.vm.vfs.fs.discovery[<url>,<uuid>]		Необходимо установить VMware Tools на гостевую виртуальную машину.
	Статистика файловой системы виртуальной машины VMware (в байтах/процентах).	Целое число	url - URL службы VMware uuid - имя хоста виртуальной машины VMware fsname - имя файловой системы режим - total/free/used/pfree/pused	Необходимо установить VMware Tools на гостевую виртуальную машину.
		vmware.vm.vfs.fs.size[<url>,<uuid>,<fsname>,<режим>]		

Примечания

¹ Путь к счетчику производительности VMware имеет следующий формат **группа/счетчик[rollup]**, где:

- **группа** - группа счетчика производительности, например *cpi*
- **счетчик** - имя счетчика производительности, например *usagemhz*
- **rollup** - тип возвращаемого значения счетчика производительности, например *average*

Таким образом пример выше может быть следующим счетчиком: *cpi/usagemhz[average]*

Описания групп, имен счетчиков производительности и возвращаемые ими типы значений можно найти в [документации к VMware](#).

² Значения по этим элементам данных получаются от счётчиков производительности VMware и **параметр** *VMwarePerfFrequency* используется для обновления данных по этим элементам данных в Zabbix VMware кэше:

- *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 [мониторинг виртуальных машин](#) для получения более подробной информации о настройке Zabbix для мониторинга сред VMware.

6 Мониторинг файлов журналов

Обзор

Zabbix можно использовать для централизованного мониторинга и анализа файлов журналов с/без поддержки ротации журналов.

Можно использовать оповещения для предупреждения пользователей, когда файл журнала содержит конкретные строки или шаблоны строк.

Для наблюдения за файлом журнала у вас должно быть:

- Работающий Zabbix агент на узле сети
- Настроенный элемент данных для мониторинга журнала

Максимальный размер наблюдаемого файла журнала зависит от [поддержки файлов большого объема](#).

Настройка

Проверка параметров агента

Убедитесь, что в [файле конфигурации агента](#):

- Параметр 'Hostname' совпадает с именем узла сети в веб-интерфейсе
- Указаны сервера в параметре 'ServerActive' для обработки активных проверок

Настройка элемента данных

Настройте [элемент данных](#) для мониторинга журнала.

The screenshot shows the configuration form for a 'Log item'. The fields are as follows:

* Name	Log item	
Type	Zabbix agent (active)	
* Key	log[/var/log/syslog,error]	Select
Type of information	Log	
* Update interval	30s	
* History storage period	3600	
Log time format	ppppddphh:mm:ss	

Все обязательные поля ввода отмечены красной звёздочкой.

Специально для элементов данных наблюдения за журналами вы должны указать:

Тип	Здесь выберите Zabbix агент (активный) .
	Укажите: log[/путь/к/файлу/имя_файла,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<вывод>,<максзадержка>]
Ключ	или logrt[/путь/к/файлу/регулярное_выражение_описывающее_шаблон_имени_файла,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,<вывод>,<максзадержка>]

Zabbix агент фильтрует записи из файла журнала по регулярному выражению, если оно указано.

Если требуется только количество совпадающих строк укажите:

log.count[/путь/к/файлу/имя_файла,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,< максзадержка >]

или

logrt.count[/путь/к/файлу/регулярное_выражение_описывающее_шаблон_имени_файла,<регулярное выражение>,<кодировка>,<макс. кол-во строк>,<режим>,< максзадержка >].

Убедитесь, что у файла имеются права на чтение для пользователя 'zabbix', в противном случае состояние элемента данных будет 'unsupported'.

Для получения более подробных сведенийсмотрите информацию о ключах **log**, **log.count**, **logrt** и **logrt.count** в разделе [поддерживаемых ключей элементов данных Zabbix агентом](#).

Выберите здесь Журнал (лог) для элементов данных **log** и **logrt** или Числовой (целое положительное) для элементов данных **log.count** и **logrt.count**.

Тип информации

Если используется опциональный параметр **ВЫВОД**, вы можете выбрать подходящий тип информации, отличный от "Журнал (лог)".

Обратите внимание, что выбор не журнального типа информации приведет к потере локального штампа времени.

Интервал обновления (в сек)

Этот параметр задает как часто Zabbix агент будет проверять наличие любых изменений в файле журнала. Указав этот параметр равным 1 секунде, вы можете быть уверенными, что получите новые записи как можно скорее.

В этом поле вы можете опционально задать шаблон для анализа штампа времени строки журнала.

Если оставить пустым, штамп времени не будет анализироваться.

Поддерживаемые значения:

- * **у:** Год (0001-9999)
- * **М:** Месяц (01-12)
- * **d:** День (01-31)
- * **h:** Час (00-23)

Формат времени журнала

- * **m:** Минута (00-59)
- * **s:** Секунда (00-59)

Например, рассмотрим следующую строку из файла журнала Zabbix агента: "23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211)."

Она начинается шестью символами обозначающими PID, далее следует дата, время, и остальная часть строки.

Форматом времени журнала для этой строки является "rrrrrr:yyyyMMdd:hhmmss".

Обратите внимание, что символы "р" и ":" являются лишь заменителями и могут быть чем угодно, за исключением "уMdhms".

Важные замечания

- Сервер и агент следят за размером наблюдаемого журнала и временем последней модификации (для logrt) двумя счетчиками. Дополнительно:
 - Также агент использует номера inode (на UNIX/GNU/Linux), индексы файлов (на Microsoft Windows) и MD5 суммы первых 512 байт файла журнала для улучшения выбора в случае когда файлы журнала усекаются и ротируются.
 - На системах UNIX/GNU/Linux предполагается, что файловые системы где хранятся файлы журналов, сообщают числа inode, которые могут быть использованы для слежения за состоянием файлов.
 - На системах Microsoft Windows Zabbix агент определяет тип файловой системы на которой находятся файлы журналов:
 - На файловой системе NTFS 64-битные файловые индексы.

- На файловых системах ReFS (только Microsoft Windows Server 2012) 128-битные файловые ID.
- На файловых системах где файловые индексы меняются (т.е. FAT32, exFAT) используется запасной алгоритм для получения разумного подхода в неопределенных условиях, когда сжатие файла журнала приводит в результате к множеству файлов журналов с одинаковым временем изменения.
- Номера inode, индексы файлов и суммы MD5 собираются Zabbix агентом. Они не передаются Zabbix серверу и теряются в случае остановки Zabbix агента.
- Не меняйте время последней модификации файлов журналов, используя утилиту 'touch', не копируйте файл журнала с последующим восстановлением его имени (это изменит идентификатор иноды файла). В обоих случаях файл будет рассматриваться как другой и будет проанализирован с самого начала, что может привести к дубликатам оповещений.
- Если есть несколько совпадающих файлов журналов для элемента данных `logrt[]` и Zabbix агент следит за наиболее новым из них и этот более новый файл журнал удалается, предупреждающее сообщение будет записано `"there are no files matching "<regexp mask>" in "<directory>"`. Zabbix агент игнорирует файлы журналы с временем изменения меньше чем последнее время модификации полученное агентом во время проверки элемента данных `logrt[]`.
- Агент начинает читать файл журнала с той позиции, на которой он остановился последний раз.
- Количество байт уже проанализированное (счётчик размера) и время последней модификации (счетчик времени) хранятся в базе данных Zabbix и отправляются агенту, для уверенности, что агент начнет читать файл журнала с этой позиции в случаях, когда агент только что был запущен или агент получил элементы данных, которые были ранее деактивированы или не поддерживались. Однако, если агент получает ненулевой размер счётчика от сервера, но элементы данных `logrt[]` или `logrt.count[]` не найдены и не удается найти соответствующие файлы, счётчик размера сбрасывается в 0, чтобы начать анализ сначала, если файлы появятся позже.
- Всякий раз, когда файл журнала становится меньше, чем значение счетчика размера известное агенту, счетчик обнуляется и агент начинает читать файл журнала с самого начала, принимая во внимание счетчик времени.
- Если есть несколько файлов журналов, с одинаковым последним временем модификации файла в соответствующей папке, агент пытается корректно проанализировать все файлы журналы с одинаковым временем модификации и избежать пропущенных данных или проанализировать данные дважды, несмотря на это невозможно охватить все возможные ситуации. Агент не предполагает какую либо определенную схему ротации файлов журналов, либо определяет ее. Когда есть несколько файлов журналов с одинаковым последним временем изменения, агент будет обрабатывать их лексикографически в порядке убывания. Таким образом, для некоторых схем ротации файлы журналы будут проанализированы в их оригинальном порядке. Для других же схем ротации журналов первоначальный порядок файла журнала не будет соблюдать, что может привести к получению найденных по шаблону строк файла журнала в измененном порядке (проблема не случится, если файлы журнала имеют разное время последнего изменения).
- Zabbix агент обрабатывает новые записи файла журнала один раз за *Период обновления* секунд.
- Zabbix агент отправляет не более чем **макс. кол-во строк** записей из файла журнала за секунду. Это ограничение предотвращает перегрузку сети и ресурсов процессора и переопределяет значение по умолчанию предусмотренное параметром **MaxLinesPerSecond** в [файле конфигурации агента](#).

- Для поиска необходимой строки Zabbix обрабатывает в 10 раза больше строк, чем указано в параметре **MaxLinesPerSecond**. Таким образом, например, если элемент данных `log[]` или `logrt[]` имеет *Интервал обновления* 1 секунда, по умолчанию агент будет анализировать не более чем 400 строк файла журнала и будет отправлять не более чем 200 совпадающих записей Zabbix серверу за одну проверку. Увеличением параметра **MaxLinesPerSecond** в файле конфигурации агента или указанием параметра **макс. кол-во строк** в ключе элемента данных, лимит можно увеличить вплоть до 10000 проанализированных записей в журнале и 1000 совпадающих записей для отправки Zabbix серверу за одну проверку. Если *Интервал обновления* указан значением в 2 секунды, лимиты для одной проверки могут быть увеличены в два раза больше, чем для *Интервала обновления* в 1 секунду.
- Кроме того, данные из файлов журналов всегда ограничены 50% размера буфера отправки у агента, даже если в буфере нет значений не связанных с данными из файлов журналов. Таким образом, значения **макс. кол-во строк** будут отправлены за одно соединение (а не в нескольких соединениях), параметр [BufferSize](#) агента должен быть по крайней мере равен макс. кол-во строк x 2.
- При отсутствии данных для элементов данных журналов весь размер буфера используется для значений не связанных с данными из журналов. Когда появляются значения от файлов журналов они заменяют устаревшие данные не связанные с файлами журналов, если требуется, до максимального уровня 50%.
- Для записей в файле журнала длиннее 256КБ, только первые 256КБ сопоставляются с регулярным выражением, остальная часть игнорируется. Однако, если Zabbix агент был остановлен в процессе обработки длинной строчки, внутреннее состояние агента теряется и длинная строчка может быть проанализирована иначе после запуска агента.
- Специальное примечание для разделителей пути “\”: если формат файла представлен как “file\log”, тогда там не должно быть папки “file”, поскольку невозможно однозначно определить, экранируется ли это символ ”\“ или это первый символ в имени файла.
- Регулярные выражения для `logrt[]` поддерживаются только в именах файлов, совпадение регулярного выражения с папкой не поддерживается.
- В UNIX элементы данных `logrt[]` становятся НЕПОДДЕРЖИВАЕМЫМИ, в случае если папка не существует где файл журнала должен был бы находиться.
- В Microsoft Windows, если папка не существует элемент данных не переводится в состояние НЕПОДДЕРЖИВАЕТСЯ (например, если в ключе элемента данных папка указана с ошибкой)
- Отсутствие файла журнала для элемента данных `logrt[]` не переводит его в состояние НЕПОДДЕРЖИВАЕТСЯ.
- Ошибки чтения файлов журналов для элемента данных `logrt[]` записываются в журнал агента как предупреждения, но не переводят элемент данных в состояние НЕПОДДЕРЖИВАЕТСЯ.
- Журнал Zabbix агента может быть очень полезен для поиска причин почему элементы данных `log[]` или `logrt[]` становятся НЕПОДДЕРЖИВАЕМЫМИ. Zabbix может мониторить свой файл журнала, за исключением случая когда он в режиме `DebugLevel=4`.

Извлечение совпадающей части регулярного выражения

Иногда мы можем захотеть извлечь только интересующие значения из требуемого файла вместо того, чтобы получать всю строку, в случае когда найдено совпадение с регулярным выражением.

Начиная с Zabbix 2.2.0, элементы данных файлов журналов расширены возможностью получения извлечения требуемых значений из строк файла. Добавился дополнительный параметр **вывод** у элементов данных `log` и `logrt`.

Использование параметра 'вывод' позволяет обозначить подгруппу совпадения в которой мы можем быть заинтересованы.

И так, например

```
log[/path/to/the/file,"large result buffer allocation.*Entries: ([0-9]+),,\,\1]
```

должно позволить получить количество записей со следующего содержания:

```
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
```

Причина, почему Zabbix вернет только одно число, потому что параметр 'вывод' здесь определен как \1 ссылка только на первую интересующую подгруппу: (**[0-9]+**)

Вместе с возможностью извлечения и получения числа, значение можно использовать в определениях триггеров.

Использование параметра максзадержка

Параметр 'максзадержка' в элементах данных журналов позволяет игнорировать более старые строки с целью получения наиболее новых строк проанализированных в течении "максзадержка" секунд.

Параметр 'maxdelay' > 0, может привести к **игнорированию важных записей в файлах журналов и пропуску оповещений**. Используйте этот параметр осторожно и на свой страх и риск, только в случае необходимости.

По умолчанию элементы данных мониторинга журналов забирают все новые строки появляющиеся в файлах журналов. Однако, имеются приложения, которые в некоторых ситуациях начинают записывать огромное количество сообщений в свои файлы журналов. Например, если база данных или DNS сервер недоступны, то такие приложения могут флудить файлы журналов тысячами практически идентичных сообщений об ошибке до тех пор пока не восстановится нормальный режим работы. По умолчанию, все эти сообщения добросовестно анализируются и совпадающие строки отправляются на сервер, как настроено в элементах данных log и logrt.

Встроенная защита от перегрузов состоит из настраиваемого параметра 'макс. кол-во строк' (защищающий сервер от слишком большого количества приходящих совпадающих строк в журнале) и ограничения в 4*'макс. кол-во строк' (защищает CPU и I/O хоста от перегрузки агентам одной проверкой). Тем не менее имеется 2 проблемы со встроенным механизмом защиты. Первая, на сервер будет отправлено большое количество потенциально не так информативных сообщений, которые займут место в базе данных. Вторая, по причине ограниченного количества строк анализируемых в секунду агент может отставать на часы от самых новых записей в журнале. Вполне вероятно, что вы захотите как можно быстрее быть информированным о текущей ситуации в файлах журналов вместо ковыряния часами старых записей.

Решение этих двух проблем является использование параметра 'максзадержка'. Если параметр 'maxdelay' > 0, во время каждой проверки измеряются количество обработанных байт, количество оставшихся байт и время обработки. Отталкиваясь от этих значений, агент вычисляет оценочную задержку - как много секунд может потребоваться, чтобы проанализировать все оставшиеся записи в файле журнала.

Если задержка не превышает 'максзадержка', тогда агент поступает с анализом файла журнала как обычно.

Если задержка больше чем 'максзадержка', тогда агент **игнорирует часть файла журнала, "перепрыгивая" эту часть** к новой оценочной позиции таким образом, чтобы оставшиеся строки можно было проанализировать за 'максзадержка' секунд.

Обратите внимание, что агент даже не читает проигнорированные строки в буфер, но вычисляет приблизительную позицию для прыжка в файле.

Сам факт пропуска строк в файле журнала записывается в файл журнала агента, примерно 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
```

Количество "to byte" является оценочным, потому что после "прыжка" агент скорректирует позицию в файл к началу строки в журнале, которая может быть в файле чуть дальше или раньше.

В зависимости от того как скорость роста соотносится к скорости анализа файла журнала, вы можете не увидеть "прыжков", а можете увидеть редкие или частые "прыжки", большие или маленькие "прыжки", или даже маленькие "прыжки" каждую проверку. Колебания загрузки системы и сетевые задержки также влияют на вычисления задержки и, следовательно, "прыжки" вперед чтобы не отставать от параметра "максзадержка".

Не рекомендуется указывать 'максзадержка' < 'интервал обновления' (это может привести к частым маленьким "прыжкам").

Заметки по обработке ротации 'copytruncate' файлов журналов

`logrt` с опцией `copytruncate` подразумевает, что разные файлы журналов имеют разные записи (по крайней мере штампы времени в них отличаются), поэтому MD5 суммы начальных блоков (до первых 512 байт) будут отличаться. Два файла с одинаковыми MD5 суммами начальных блоков означают, что один из них оригинал, а второй - копия.

`logrt` с опцией `copytruncate` делает попытку правильной обработки копий файлов журналов без дублирующих сообщений. Тем не менее, такие варианты как создание нескольких копий файлов журналов с одинаковыми штампами времени, ротация файлов журналов чаще чем интервал обновления `logrt[]` элемента данных, частый перезапуск агента не рекомендуются. Агент пытается справиться со всеми этими ситуациями, но хорошие результаты не гарантируются при всех обстоятельствах.

Действия, если произошла ошибка связи между агентом и сервером

Каждая совпадающая строка с элементов данных `log[]` и `logrt[]` и результат проверки каждого элемента данных `log.count[]` и `logrt.count[]` требует свободный слот в выделенной 50% области буфера отправки в агенте. Элементы буфера регулярно отправляются серверу (или прокси) и слоты буфера становятся снова пустыми.

Пока имеются свободные слоты в выделенной области для журналов в буфере отправки в агенте и связь между агентом и сервером (или прокси) нарушена, результаты мониторинга журналов накапливаются в буфере отправки. Такое поведение позволяет смягчить кратковременные нарушения связи.

Во время длительных нарушений связи все слоты журналов становятся занятыми и выполняются следующие действия:

- Проверки элементов данных `log[]` и `logrt[]` останавливаются. Когда связь восстановится и появятся свободные слоты, проверки вернутся к предыдущей позиции. Не совпадающие строки потеряются. Совпадающие строки не будут потеряны, они просто отправятся позже.
- Проверки `log.count[]` и `logrt.count[]` останавливаются, если `maxdelay = 0` (по умолчанию). Поведение похоже на элементы данных `log[]` и `logrt[]`, описанное выше. Обратите внимание, что потеря связи может повлиять на результаты `log.count[]` и `logrt.count[]`: например, одна проверка насчитает 100 совпадающих строк в файле журнала, но по причине отсутствия свободных слотов в буфере проверка будет остановлена. Когда связь восстановится агент насчитает те же 100 совпадающих строк, а также 70 новых совпадающих строк. После чего агент отправит количество = 170, так как они найдены за одну проверку.
- Проверки `log.count[]` и `logrt.count[]` при `maxdelay > 0`: если не было “прыжка” во время проверки, тогда поведение аналогично описанному выше. Если всё же был “прыжок” через строки файла журнала, тогда позиция после “прыжка” сохранится и подсчитанный результат будет отброшен. Таким образом, агент пытается не отставать от увеличивающегося файла журнала, даже в случае проблем со связью.

7.2.2.7 Вычисляемые элементы данных

1 Обзор

При помощи вычисляемых элементов данных вы можете выполнять подсчеты на основании других элементов данных.

Таким образом, вычисляемые элементы данных являются одним из способов создания виртуальных источников данных. Значения будут периодически вычисляться на основе арифметического выражения. Все вычисления выполняются Zabbix сервером - Zabbix агенты и прокси не участвуют в подсчете вычисляемых элементов данных и не имеют никаких данных для этого.

Полученные данные записываются в базу данных Zabbix, как и для любого другого элемента данных - это означает хранение значений истории и динамики изменений для быстрой генерации графиков. Созданные элементы данных можно использовать в выражениях триггеров, сославшись макросами или другими объектами, также как и на любой другой тип элементов данных.

Для использования вычисляемых элементов данных, выберите тип элемента данных **Вычисляемое**.

2 Настраиваемые поля

Ключ уникальный идентификатор элемента данных (в пределах узла сети). Вы можете создать любое имя ключа используя поддерживаемые символы.

Формула расчета необходимо ввести в поле **Формула**. Не существует практически никакой связи между формулой и ключем. В любом случае параметры ключа не используются в формуле.

Корректный синтаксис простой формулы:

функция(<ключ>|<узел сети:ключ>, <параметр1>, <параметр2>, . . .)

Где:

АРГУМЕНТ	ОПРЕДЕЛЕНИЕ
функция	Одна из функций поддерживаемых в выражениях триггеров: last, min, max, avg, count и остальные
ключ	Ключ другого элемента данных, данные которого вы хотите использовать. Его можно задать как ключ или узел сети:ключ .

Обратите внимание: Настоятельно рекомендуется заключать весь ключ в двойные кавычки ("..."), во избежании неправильного разбора из-за пробелов или запятых в ключе.

Также если в ключе имеются параметры заключенные в кавычки, то двойные кавычки должны быть экранированы с помощью обратной косой чертой (\). Смотрите ниже **Пример 5.**

параметр(ы) Параметр(ы) функций, если требуются.



Все элементы данных, на которые ссылается формула вычисляемого элемента данных, должны существовать и собирать данные ([исключения в функциях и неподдерживаемых элементах данных](#)). Также, если вы измените ключ элемента данных на который ссылается элемент данных, то вы должны вручную обновить все формулы в которых используется этот ключ.



[Пользовательские макросы](#) в формуле будут раскрыты, если используется ссылка на параметр функции или константу. Пользовательские макросы НЕ будут раскрыты, если используется ссылка на функцию, имя узла сети, параметры ключей элементов данных или оператор.

Для более сложной формулы можно использовать комбинацию функций, операторов и скобок. Вы можете использовать все функции и [операторы](#), которые поддерживаются в выражениях триггеров. Обратите внимание, что синтаксис немного отличается, однако логика и приоритет операций в точности такой же.

В отличии от выражений триггеров, Zabbix обрабатывает вычисляемые элементы данных в соответствии со временем обновления элемента данных, а не при получении нового значения.



Если результатом вычисления будет число с плавающей точкой, оно будет обрезано до целого числа, в случае если типом информации элемента данных указан Числовой (целое положительное).

Вычисляемый элемент данных может перейти в неподдерживаемое состояние в нескольких случаях:

1. элемент(ы) данных, на которые имеется ссылка
 - не найден
 - деактивирован
 - принадлежит деактивированному узлу сети
 - не поддерживается (смотрите исключения в разделах [Функции и неподдерживаемые элементы данных](#), [Выражения с неподдерживаемыми элементами данных и неизвестными значениями](#) и [Операторы](#))
2. нет данных для вычисления функции
3. деление на ноль
4. использован некорректный синтаксис

Поддержка вычисляемых элементов данных была введена в Zabbix 1.8.1.

Начиная с Zabbix 3.2 вычисляемые элементы данных в некоторых случаях могут использовать неподдерживаемые элементы данных как описано в [Функции и неподдерживаемые элементы данных](#), [Выражения с неподдерживаемыми элементами данных и неизвестными значениями](#) и [Операторы](#).

3 Примеры использования

Пример 1

Вычисление процентного отношения свободного места на диске для '/'.

Используйте функцию **last**:

```
100*last("vfs.fs.size[/,free]"/last("vfs.fs.size[/,total]"))
```

Zabbix будет брать последние значения полного и свободного объема диска и вычислять процентное отношение в соответствии с данной формулой.

Пример 2

Вычисление 10-минутного усреднения количества значений обрабатываемых Zabbix.

Используйте функцию **avg**:

```
avg("Zabbix Server:zabbix[wcache,values]", 600)
```

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

Вычисление процентного отношения входящего трафика.

Более сложное выражение:

```
100*last("net.if.in[eth0,bytes]"/(last("net.if.in[eth0,bytes]"+last("net.if.out[eth0,bytes]"))))
```

Пример 5

В вычисляемом элементе данных возможно использование агрегированных элементов данных.

Возьмите на заметку, каким образом двойные кавычки экранируются в заключенном кавычками ключе:

```
last("grpsum[\"video\", \"net.if.out[eth0,bytes]\", \"last\", \"0\"]") /  
last("grpsum[\"video\", \"nginx_stat.sh[active]\", \"last\", \"0\"]")
```

7.2.2.8 Внутренние проверки

Обзор

Внутренние проверки позволяют наблюдать за внутренним процессами Zabbix. Другими словами, вы можете наблюдать что происходит с Zabbix сервером или Zabbix прокси.

Внутренние проверки вычисляются:

- на Zabbix сервере - если узел сети наблюдается через сервер
- на Zabbix прокси - если узел сети наблюдается через прокси

Внутренние проверки обрабатываются сервером или прокси вне зависимости от состояния обслуживания узла сети (начиная с Zabbix 2.4.0).

Для использования этого элемента данных выберите тип элемента данных **Zabbix внутренний**.

Внутренние проверки обрабатываются Zabbix поллерами.

Поддерживаемые ключи

- Параметры без угловых скобок являются константами - например, 'host' и 'available' в `zabbix[host,<тип>,available]`. Используйте их в ключе элемента данных без изменения (*как есть*).
- Значения элементов данных и параметры элементов данных, которые “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.

▲	Описание	Ключ	Возвращаемое значение	Комментарии
		<code>zabbix[boottime]</code>		
	Время запуска процесса Zabbix сервера в секундах.		Целое число.	
		<code>zabbix[history]</code>		
	Количество значений хранимых в таблице HISTORY		Целое число.	Не используйте с MySQL InnoDB, Oracle или PostgreSQL! (не поддерживается прокси)
		<code>zabbix[history_log]</code>		
	Количество значений хранимых в таблице HISTORY_LOG		Целое число.	Не используйте с MySQL InnoDB, Oracle или PostgreSQL! Этот элемент данных поддерживается начиная с версии 1.8.3. (не поддерживается прокси)
		<code>zabbix[history_str]</code>		
	Количество значений хранимых в таблице HISTORY_STR		Целое число.	Не используйте с MySQL InnoDB, Oracle или PostgreSQL! (не поддерживается прокси)
		<code>zabbix[history_text]</code>		
	Количество значений хранимых в таблице HISTORY_TEXT		Целое число.	Не используйте с MySQL InnoDB, Oracle или PostgreSQL! Этот элемент данных поддерживается начиная с версии 1.8.3. (не поддерживается прокси)
		<code>zabbix[history_uint]</code>		
	Количество значений хранимых в таблице HISTORY_UINT		Целое число.	Не используйте на MySQL InnoDB, Oracle или PostgreSQL! Этот элемент данных поддерживается начиная с версии 1.8.3. (не поддерживается прокси)
		<code>zabbix[host,,items]</code>		

	Ключ		
▲	Описание	Возвращаемое значение	Комментарии
	Количество активированных элементов данных (поддерживаемых и неподдерживаемых) у узла сети.	Целое число.	Этот элемент данных поддерживается начиная с Zabbix 3.0.0.
	zabbix[host,,items_unsupported]		
	Количество активированных неподдерживаемых элементов данных узла сети.	Целое число.	Этот элемент данных поддерживается начиная с Zabbix 3.0.0.
	zabbix[host,,maintenance]		
	Возвращает текущее состояние обслуживания узла сети.	0 - узел сети в нормальном состоянии, 1 - узел сети в обслуживании со сбором данных, 2 - узел сети в обслуживании без сбора данных.	Данный элемент данных всегда обрабатывается Zabbix сервером вне зависимости от настроек узла сети (мониторится 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. Данный элемент данных поддерживается начиная с Zabbix 2.4.0.
	zabbix[host,discovery,interfaces]		
	Детали по всем добавленным интерфейсам к узлу сети в веб-интерфейсе Zabbix.	JSON объект.	Этот элемент данных можно использовать в <u>низкоуровневом обнаружении</u> . Этот элемент данных поддерживается начиная с Zabbix 3.4.0. (не поддерживается прокси)
	zabbix[host,<тип>,available]		Допустимые типы: agent , snmp , ipmi , jmx .
	Доступность определенного типа проверок на узле сети. Значение этого элемента соответствует иконкам доступности в списке узлов сети.	0 - недоступен, 1 - доступен, 2 - неизвестно.	Значение элемента данных вычисляется согласно параметрам конфигурации <u>доступности/недоступности</u> соответствующего узла сети.
	zabbix[hosts]		Этот элемент данных поддерживается начиная с Zabbix 2.0.0.

Ключ			
Описание	Возвращаемое значение	Комментарии	
Количество наблюдаемых узлов сети	Целое число.	Данный элемент данных поддерживается начиная с Zabbix 2.2.0.	
	zabbix[items]		
Количество активированных элементов данных (поддерживаемых и не поддерживаемых)	Целое число.		
	zabbix[items_unsupported]		
Количество не поддерживаемых элементов данных	Целое число.		
	zabbix[java,<параметр>]		
Получение информации связанной с Zabbix Java gateway.	<p>Если <параметром> является ping, возвращается “1”. Можно использовать для проверки доступности Java gateway, используя функцию триггера nodata().</p> <p>Если <параметром> является version, возвращается версия Java gateway. Пример: “2.0.0”.</p>	<p>Второй параметр должен быть пустым, т.к. зарезервирован для использования в будущем.</p> <p>Этот элемент данных поддерживается начиная с Zabbix 2.0.0.</p>	
	zabbix[preprocessing_queue]		
Количество значений, помещенных в очередь предварительной обработки.	Целое число.	Этот элемент данных можно использовать для мониторинга размера очереди предварительной обработки.	
			Этот элемент данных поддерживается начиная с 3.4.0.
	zabbix[process,<тип>,<режим>,<состояние>]		
Время конкретного процесса Zabbix или группы процессов (указываются с помощью <тип> и <режим>), потраченное на <состояние> в процентах. Расчет производится только за последнюю минуту.	<p>Процент времени.</p> <p>Целое с плавающей точкой.</p>	<p>На данный момент поддерживаются следующие типы процессов:</p> <p>alerter - процесс отправки уведомлений (<i>не поддерживается прокси</i>)</p> <p>alert manager - менеджер задач оповещения</p> <p>configuration syncer -</p>	
Если в поле <режим> указан номер процесса Zabbix, который не запущен (например, при 5 запущенных поллерах, в <режиме> указано 6),			

Ключ

Описание	Возвращаемое значение	Комментарии
такой элемент данных переходит в состояние <i>не поддерживается</i> . Минимум и максимум относится к процентному использованию одним процессом. Таким образом, если в группе из 3 поллеров процентное использование составило 2, 18 и 66, тогда min должен вернуть 2, а max должен вернуть 66. Процессы сообщают то, что они делают, используя разделяемую память, и процесс самодиагностики резюмирует эти данные каждую секунду. Изменения состояний (busy/idle) регистрируются при изменении - таким образом, процессы, которые становятся занятыми, регистрируют это состояние и не меняют и не обновляют его пока процесс не станет свободным. Такое поведение гарантирует, что даже полностью зависшие процессы будут корректно зарегистрированы как 100% занятые. В настоящее время, "busy" означает "не спит", но в будущем могут добавиться дополнительные состояния - ожидание блокировки, выполнение запросов в базу данных, и т.д. В Linux и в большинстве других систем, точность составляет 1/100 секунды.		процесс управления кэшем данных конфигурации в оперативной памяти data sender - процесс отправки данных с прокси (<i>не поддерживается сервером</i>) discoverer - процесс обнаружения устройств escalator - процесс эскалации действий (<i>не поддерживается прокси</i>) heartbeat sender - процесс уведомления сервера прокси сервером о состоянии прокси (<i>не поддерживается сервером</i>) history syncer - процесс, который записывает историю в БД housekeeper - процесс удаления старых данных истории http poller - поллер веб-мониторинга icmp pinger - поллер проверок icmp ping ipmi manager - менеджер IPMI поллеров ipmi poller - поллер для проверок по IPMI java poller - поллер для Java проверок poller - обычный поллер для пассивных проверок preprocessing manager - менеджер задач предобработки preprocessing worker - процесс предобработки данных proxy poller - поллер для пассивных прокси (<i>не поддерживается прокси</i>) self-monitoring - процесс сбора внутренней статистики сервера snmp trapper - траппер сбора/обработки SNMP трапов task manager - процесс для удаленного выполнения задач, которые

Ключ



Описание

Возвращаемое значение

Комментарии

запрашиваются другими компонентами (например, возможности закрытия проблемы, подтверждения проблемы, принудительной проверки значения элемента данных, удаленной команды)
timer - процесс обработки обслуживаний
trapper - процесс-улавливатель для активных проверок, трапов, связей между нодами и прокси
unreachable poller - поллер недоступных устройств
vmware collector - коллектор данных VMware, ответственный за сбор данных со служб VMware

Примечание: Вы также можете увидеть все эти типы процессов в файле журнала сервера.

Допустимые режимы:
avg - среднее значение по всем процессам указанного типа (по умолчанию)
count - количество форков указанного типа процесса, <состояние> не должно быть указано
max - максимальное значение
min - минимальное значение
<номер процесса> - номер процесса (от 1 до количества префорк процессов).
Например, если запущено 4 траппера, тогда значение от 1 до 4.

Допустимые состояния:
busy - процесс в занятом состоянии, например, обработка запроса (по умолчанию).
idle - процесс в свободном состоянии, ничего не делающий.

Примеры:

Ключ

Описание	Возвращаемое значение	Комментарии
	⇒ zabbix[process,poller,avg,busy]] → среднее время, потраченное процессами поллеров, которые что-либо делали за последнюю минуту ⇒ zabbix[process,"icmp pinger",max,busy] → максимальное время, потраченное любыми процессами ICMP pinger, которые что-либо делали за последнюю минуту ⇒ zabbix[process,"history syncer",2,busy] → время, потраченное на что-то процессом синхронизации истории номер 2 в течении последней минуты ⇒ zabbix[process,trapper,count] → общее количество запущенных процессов trapper	
		Этот элемент данных поддерживается начиная с Zabbix 1.8.5 .
zabbix[proxy,<имя>,<параметр>]	<имя> - Имя прокси Список поддерживаемых параметров (<параметр>): lastaccess – штамп времени последнего сообщения о доступности от прокси	
		Например, ⇒ zabbix[proxy,"Germany",lastaccess]
Информация доступности по Zabbix прокси.	Целое число.	Можно использовать функцию триггеров fuzzytime() для проверки доступности прокси. Начиная с Zabbix 2.4.0 этот элемент данных всегда обрабатывается Zabbix сервером независимо от настроек узла сети (наблюдается через сервер или прокси).

Ключ			
Описание	Возвращаемое значение	Комментарии	
Количество значений в таблице истории прокси ожидающих отправку на сервер	zabbix[proxy_history]	Целое число.	Данный элемент данных поддерживается начиная с Zabbix 2.2.0 <i>(не поддерживается сервером)</i>
Количество наблюдаемых элементов данных в очереди, которые задерживаются <от> и <до> секунд включительно.	zabbix[queue,<от>,<до>]	<от> - по умолчанию: 6 секунд <до> - по умолчанию: бесконечно	Для этих параметров поддерживаются символы времени (s,m,h,d,w). Параметры от и до поддерживаются начиная с Zabbix 1.8.3 .
Статистика доступности кэша конфигурации	zabbix[rcache,<кэш>,<режим>]	Целое число (для размера): Целое с плавающей точкой (для процентов).	Кэш: buffer Режим: total - полный размер буфера free - размер свободного места в буфере pfree - процент свободного места в буфере used - размер использованного места в буфере
Требуемое быстродействие Zabbix сервера, имеется ввиду новые значения в секунду.	zabbix[requiredperformance]	Целое с плавающей точкой.	Приблизительно соответствует значению “Требуемое быстродействие сервера, новые значения в секунду” со страницы Отчеты → Информация о системе . Поддерживается начиная с Zabbix 1.6.2 .
Количество значений хранимых в таблице TRENDS	zabbix[trends]	Целое число.	Не используйте с MySQL InnoDB, Oracle или PostgreSQL! <i>(не поддерживается прокси)</i>
Количество значений хранимых в таблице TRENDS_UINT	zabbix[trends_uint]	Целое число.	Не используйте с MySQL InnoDB, Oracle или PostgreSQL! Этот элемент данных поддерживается начиная с

Ключ

Описание	Возвращаемое значение	Комментарии
		Zabbix 1.8.3. (не поддерживается прокси)
zabbix[triggers] Количество активированных триггеров в базе данных Zabbix, со всеми активированными элементами данных, которые упомянуты в выражениях триггеров на активированных узлах сети.	Целое число.	(не поддерживается прокси)
zabbix[uptime] Время непрерывной работы процесса Zabbix сервера в секундах.	Целое число.	
zabbix[vcache,buffer,<режим>] Статистика доступности кэша значений.	Целое число (для размера); Целое с плавающей точкой (для процентов).	Режим: total - полный размер буфера free - размер свободного места в буфере pfree - процент свободного места в буфере used - размер использованного места в буфере pused - процент использованного места в буфере Данный элемент данных поддерживается с Zabbix 2.2.0. (не поддерживается прокси)
zabbix[vcache,cache,<параметр>] Статистика эффективности кэша значений Zabbix.	Целое число. С параметром равным mode : 0 - нормальный режим, 1 - режим низкой памяти	Параметр: requests - общее количество запросов hits - количество попаданий в кэш (история берется из кэша) phits - процент попаданий в кэш misses - количество непопаданий в кэш (история берется из базы данных) mode - режим работы кэша значений Данный элемент данных поддерживается с Zabbix 2.2.0 и параметр mode поддерживается начиная с Zabbix 3.0.0. (не

Ключ

Описание	Возвращаемое значение	Комментарии
		поддерживается прокси)
		Вы можете использовать этот ключ с шагом предобработки <i>Изменение в секунду</i> при необходимости получения статистики значений в секунду.
	zabbix[vmware,buffer,<режим>]	
Статистика доступности кэша VMware.	Целое число (для размера); Целое с плавающей точкой (для процентов).	Режим: total - полный размер буфера free - размер свободного места в буфере pfree - процент свободного места в буфере used - размер использованного места в буфере pused - процент использованного места в буфере
		Данный элемент данных поддерживается с Zabbix 2.2.0.
	zabbix[wcache,<кэш>,<режим>]	
Статистика и доступность кэша записи Zabbix.		Требуется обязательно указывать <кэш>.
Кэш Режим		
values		Счетчик.
all (по умолчанию)	Количество значений обрабатываемых Zabbix сервером или Zabbix прокси, исключая не поддерживаемые	Вы можете использовать этот ключ с шагом предобработки <i>Изменение в секунду</i> при необходимости получения статистики значений в секунду.
float	Количество обработанных значений с плавающей точкой.	Счетчик.
uint	Количество обработанных целочисленных значений.	Счетчик.
str	Количество обработанных символьных/строковых значений.	Счетчик.
log	Количество обработанных значений из файлов журналов.	Счетчик.
text	Количество обработанных текстовых значений.	Счетчик.

Ключ

	Описание	Возвращаемое значение	Комментарии
not supported	Количество неподдерживаемых элементов данных.	Целое число.	Счетчик. Режим <i>not supported</i> поддерживается начиная с Zabbix 1.8.6 .
history	pfree (<i>по умолчанию</i>)	Свободное место в буфере истории в процентах.	Число с плавающей точкой. Кэш истории используется для хранения значений элементов данных. Низкое количество свободного места отражает проблемы с производительностью на стороне базы данных.
index	free total used	Размер свободного места в буфере истории. Полный размер буфера. Размер занятого места в буфере истории.	Целое число. Целое число. Целое число.
	pfree (<i>по умолчанию</i>)	Свободное место в буфере индексов истории.	Целое с плавающей точкой. Кэш индексов истории используется для индексирования значений записываемых в кэш истории. Кэш индекса поддерживается начиная с Zabbix 3.0.0 .
trend	free total used	Свободное место в буфере индексов истории. Полный размер буфера индексов истории. Размер занятого места в буфере индексов истории.	Целое число. Целое число. Целое число.
	pfree (<i>по умолчанию</i>)	Свободное место в буфере динамики изменений в процентах.	Число с плавающей точкой. Кэш динамики изменений агрегируется за текущий час для всех элементов данных, которые получают данные. (не поддерживается прокси)
	free total used	Свободное место в буфере динамики изменений. Полный размер буфера динамики изменений. Размер занятого места в буфере динамики изменений.	Целое число. Целое число. Целое число (не поддерживается прокси) (не поддерживается прокси) (не поддерживается прокси)

7.2.2.9 Проверки через SSH

Обзор

SSH проверки выполняются без какого-либо агента. Zabbix агент не требуется для проверок выполняемых по SSH.

Для выполнения SSH проверок Zabbix сервер должен быть [изначально сконфигурирован](#) с поддержкой SSH2.

Минимально поддерживаемой версией библиотеки libssh2 является версия 1.0.0.

Настройка

Аутентификация парольной фразой

Проверки SSH предоставляют два метода аутентификации, пара логин пользователя/пароль и на основе ключа-файла.

Если вы не собираетесь использовать ключ, то никакой дополнительной настройки не требуется, при компиляции из исходных кодов, необходима также привязка библиотеки libssh2 к Zabbix.

Аутентификация на базе ключа

Для использования элементов данных SSH на основе аутентификации по ключу необходимо произвести некоторые изменения в конфигурации сервера.

Откройте файл конфигурации Zabbix сервера ([zabbix_server.conf](#)) из под `root` и найдите следующую строку:

```
# SSHKeyLocation=
```

Раскомментируйте её и укажите полный путь к папке, где размещены публичные и приватные ключи:
`SSHKeyLocation=/home/zabbix/.ssh`

Затем сохраните файл и перезапустите `zabbix_server`.

Где `/home/zabbix` домашняя папка для аккаунта `zabbix` пользователя и `.ssh` папка, куда будут по умолчанию генерированы с помощью команды [ssh-keygen](#) публичные и приватные ключи.

Обычно при установке пакетов `zabbix-server` на разных дистрибутивах ОС создается аккаунт `zabbix` пользователя с домашней папкой в не очень известных местах (как для системных аккаунтов). Например, для CentOS папка `/var/lib/zabbix`, для Debian она `/var/run/zabbix`.

До начала генерирования ключей, рассмотрите вариант перемещения домашней папки в более известное место (интуитивно ожидаемое). Этот вариант будет соответствовать параметру `SSHKeyLocation` конфигурации Zabbix сервера, упомянутого выше.

Эти шаги можно пропустить, если аккаунт `zabbix` добавлен 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 аккаунта `zabbix` пользователя все работающие процессы, которые его используют должны быть остановлены:

```
# service zabbix-agent stop  
# service zabbix-server stop
```

Для изменения размещения 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 (в CentOS, например), поэтому её необходимо создать в новом месте. 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 запустить ранее остановленные процессы:

```
# service zabbix-agent start  
# service zabbix-server start
```

Теперь шаги генерирования публичных и приватных ключей можно выполнить командой:

Возьмите на заметку: публичные и приватные ключи (`id_rsa.pub` и `id_rsa` соответственно) генерируются по умолчанию в папку `/home/zabbix/.ssh`, которая соответствует параметру конфигурации `SSHKeyLocation` Zabbix сервера.

Ключи других типов кроме “rsa” могут быть использованы, если поддерживаются утилитой ssh-keygen и библиотекой libssh2, которая используется Zabbix.

Диалог настройки командной строки

Этот шаг необходимо выполнить только один раз на каждом хосте, который будет наблюдаться с использованием SSH проверок.

При использовании следующей команды, файл **публичного** ключа будет установлен на удаленный хост *10.10.10.10*, для того чтобы потом можно было выполнять SSH проверки при помощи аккаунта *root*:

```
# 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's password:
Теперь попытайтесь зайти на машину с помощью "ssh 'root@10.10.10.10'" и проверьте там:
  .ssh/authorized_keys
чтобы убедиться, что мы не добавили дополнительные ключи, которые нежелательны.
```

Теперь можно проверить вход по SSH с использованием приватного ключа по умолчанию (`/home/zabbix/.ssh/id_rsa`) у аккаунта `zabbix` пользователя:

```
# sudo -u zabbix ssh root@10.10.10.10
```

Если вход успешен, то часть настройки в командной строке завершена и удаленное соединение по SSH можно закрыть.

Настройка элемента данных

Фактически выполняемые команда(ы) необходимо поместить в поле **Выполняемый скрипт** при настройке элемента данных.

Можно выполнять несколько команд одну за другой, размещая каждую на новой строке. В этом случае возвращаемые значения также будут отформатированы как многострочные.

* Name SSH test check (whithout passphrase)
Type SSH agent
* Key ssh.run[clear]
* Host interface 127.0.0.1 : 10051
Authentication method Public key
* User name root
* Public key file id_rsa.pub
* Private key file id_rsa
Key passphrase
* Executed script service mysql-server status
Type of information Numeric (unsigned)
Units
* Update interval 30s

Все обязательные поля ввода отмечены красной звёздочкой.

Поля, которые требуют специфичную информацию SSH элементов данных:

Параметр элемента данных	Описание	Комментарии
Тип	Здесь выберите SSH агент .	<уникальное короткое описание> обязательно и должно быть уникальным у всех элементов данных типа SSH в рамках одного узла сети
Ключ	Уникальный (в пределах узла сети) ключ элемента данных в формате ssh.run[<уникальное короткое описание>,<ip>,<порт>,<кодировка>]	Порт 22 по умолчанию, а не порт указанный в интерфейсе узла сети к к которому этот элемент данных назначен
Метод	Один из “Пароль” или “Публичный	

Параметр элемента данных	Описание	Комментарии
аутентификации	ключ”	
Имя пользователя	Имя пользователя для аутентификации на удаленном хосте. Требуется	
Файл публичного ключа	Имя файла публичного ключа, если Метод аутентификации “Публичный ключ”. Требуется	Пример: <i>id_rsa.pub</i> - имя файла публичного ключа по умолчанию сгенерированного командой ssh-keygen
Файл приватного ключа	Имя файла приватного ключа, если Метод аутентификации “Публичный ключ”. Требуется	Пример: <i>id_rsa</i> - имя файла приватного ключа
Пароль или Парольная фраза	Пароль при аутентификации или Парольная фраза, если была использована фраза для приватного ключа	Оставьте поле <i>Парольная фраза</i> пустым, если фраза не используется Смотрите также известные проблемы по поводу использования парольных фраз
Выполняемый скрипт	Выполняемые команды командной строки при использовании удаленной сессии SSH	Примеры: <i>date +%s</i> <i>service mysql-server status</i> <i>ps auxww grep httpd wc -l</i>



Библиотека libssh2 может обрезать выполняемые скрипты до ~32КБ

7.2.2.10 Проверки через Telnet

Обзор

Telnet проверки выполняются без какого-либо агента. Zabbix агент не нужен для проверок через Telnet.

Настраиваемые поля

Действующая команда(ы) для выполнения должны быть размещены в поле **Выполняемый скрипт** при настройке элемента данных.

Можно выполнять несколько команд одну за другой, размещая каждую на новой строке. В этом случае результирующее значение также будет отформатировано как многострочное.

Поддерживаемые символы, которыми строка может заканчиваться:

- \$
- #
- >
- %



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	Комментарии
telnet.run[<的独特ое короткое описание>,<ip>,<порт>,<кодировка>]	Выполнение команды на удаленном устройстве с использованием telnet соединения	



Если telnet проверка возвращает значение с не-ASCII символами и в не-UTF8 кодировке, то необходимо указать корректный параметр ключа <кодировка>. Смотрите страницу [о кодировке возвращаемых значений](#) для получения более подробных сведений.

7.2.2.11 Внешние проверки

Обзор

Внешняя проверка исполняется Zabbix сервером [выполнением shell скрипта](#) или бинарного файла. Однако, когда узлы сети наблюдаются через Zabbix прокси, внешние проверки выполняются через этот прокси.

Внешние проверки не требуют на наблюдаемом узле сети какого-либо агента.

Синтаксис ключа элемента данных:

скрипт [<параметр1>, <параметр2>, . . .]

Где:

АРГУМЕНТ

ОПРЕДЕЛЕНИЕ

скрипт Имя shell скрипта или бинарного файла.

параметр(ы) Опциональные параметры командной строки.

Если вы не хотите передавать какие-нибудь параметры скрипту, вы можете использовать:

script[] или
script

Zabbix сервер заглянет в папку указанную как размещение внешних скриптов (параметр 'ExternalScripts' в [файле конфигурации Zabbix сервера](#)) и выполнит заданную команду. Команда будет выполнена от имени пользователя под которым запущен Zabbix сервер, так что любые права доступа или переменные среды должны быть обработаны в оболочке скрипта, если необходимо, и права доступа на команду должны разрешать этому пользователю выполнение скрипта. Для выполнения доступны только те команды, которые имеются в наличии в указанной папке.



Не злоупотребляйте внешними проверками! Каждый скрипт требует запуска процесса форка Zabbix сервером, запуск множества скриптов может значительно снизить производительность Zabbix.

Пример использования

Выполнение скрипта **check_oracle.sh** с первым параметром '**-h**'. Второй параметр будет заменен 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 выполнит:

`check_oracle.sh '-h' '192.168.1.4'`

Результат внешней проверки

Результирующим значением проверки является стандартный вывод вместе со стандартным выводом ошибок (возвращается полный вывод с обрезанными пробелами в конце начиная с Zabbix 2.0).



Текстовый (символ, журнал или текстовый типы данных) элемент данных не станет неподдерживаемым в случае вывода стандартной ошибки.

В случае, если выполняемый скрипт не найден, Zabbix сервер не имеет необходимых прав на его запуск, элемент данный становится неподдерживаемым и будет возвращено соответствующее сообщение об ошибке. В случае превышения времени ожидания, элемент данных также помечается как неподдерживаемый, соответствующее сообщение об ошибке будет отображено и отдельный процесс этого скрипта будет ликвидирован.

7.2.2.12 Агрегированные проверки

Обзор

В агрегированных проверках Zabbix сервер собирает совокупную информацию из элементов данных, выполняя прямые запросы в базу данных.

Агрегированные проверки не требуют на наблюдаемом узле сети наличия какого-либо агента.

Синтаксис

Синтаксис ключа агрегированного элемента данных:

функция_группировки["группа узлов сети", "ключ элемента данных", функция_элемента_данных, период_времени]

Поддерживаемые функции группировки (функция_группировки):

Функция группировки	Описание
grpavg	Среднее значение
grpmax	Максимальное значение
grpmin	Минимальное значение
grpsum	Сумма значений

Можно включить несколько групп узлов сети, добавив массив с элементами разделенными запятыми. Если указать родительскую группу узлов сети, будет включена родительская группа и все вложенные группы узлов сети вместе с их элементами данных.

Все элементы данных, на которые имеется ссылка из ключа агрегированного элемента данных, должны существовать и собирать данные. В вычисления добавляются только активированные элементы данных у активированных узлов сети.

Ключ агрегированного элемента данных необходимо обновлять вручную, если будет изменен ключ элемента данных на который агрегированный элемент данных ссылается.

Поддерживаемые функции элементов данных (функция_элемента_данных):

Функция элемента данных	Описание
avg	Среднее значение
count	Количество значений
last	Последнее значение
max	Максимальное значение
min	Минимальное значение
sum	Сумма значений

Параметр **период_времени** задает период времени последних собранных значений. Для удобства вы можете использовать [поддерживаемые символы единиц измерения](#), например '5m' (минут) вместо '300' (секунд) или '1d' (день) вместо '86400' (секунд).

Количество значений (с префиксом #) не поддерживается.

Параметр **период_времени** игнорируется сервером в случае, если третий параметр (функция элемента данных) - *last* и его можно не указывать:

```
groupfunc["Группа узлов сети", "Ключ элемента данных", last]
```

Если результатом агрегации будет число с плавающей точкой, оно будет обрезано до целого числа, в случае если типом информации агрегированного элемента данных указан **Числовой (целое положительное)**.

Агрегированный элемент данных может стать неподдерживаемым, если:

- не найдено ни одного элемента данных, на которые он ссылается (что может произойти, если ключ элемента данных некорректный, элементы данных не существуют или все добавленные группы некорректны)
- отсутствуют данные для вычисления функции

Примеры использования

Примеры ключей агрегированных проверок:

Пример 1

Суммарный размер дисков в группе узлов сети 'MySQL Servers'.

```
grpsum["MySQL Servers", "vfs.fs.size[/,total]", last]
```

Пример 2

Средняя загрузка процессора в группе узлов сети 'MySQL Servers'.

```
grpavg["MySQL Servers", "system.cpu.load[,avg1]", last]
```

Пример 3

5-минутное усреднение количества запросов в секунду для группы узлов сети 'MySQL Servers'.

```
grpavg["MySQL Servers", mysql.qps, avg, 5m]
```

Пример 4

Средняя загрузка CPU на всех узлах сети в нескольких группах узлов сети.

```
grpavg[[ "Servers A", "Servers B", "Servers C"], system.cpu.load, last]
```

7.2.2.13 Траппер элементы данных

Обзор

Траппер элементы данных принимают входящие данные вместо запроса этих данных.

Это полезно для любых данных, которые вы возможно захотите “запихнуть” в Zabbix.

Для использования элемента данных траппера вы должны:

- иметь в Zabbix настроенный траппер элемент данных
- отправлять данные в Zabbix

Настройка

Настройка элемента данных

Для настройки элемента данных траппер:

- Перейдите в: *Настройка* → Узлы сети

- Нажмите на *Элементы данных* в строке узла сети
- Нажмите на *Создать элемент данных*
- Введите в диалоге параметры элемента данных

* Name	Trapper item
Type	Zabbix trapper
* Key	trap
Type of information	Text
* History storage period	3600
Allowed hosts	

Все обязательные поля ввода отмечены красной звёздочкой.

Поля, которые требуют указания специфичной информации для траппер элемента данных:

Тип Здесь выберите **Zabbix траппер**.

Ключ Введите ключ, который будет использован для распознавания элемента данных при отправке данных.

Тип информации Выберите тип информации, который будет соответствовать формату данных, которые будут отправлены.

Список разделенных запятыми IP адресов, опционально в CIDR представлении, или имена узлов сети.

Если задано, входящие подключения будут приниматься только с перечисленных здесь узлов сети.

Если включена поддержка IPv6, тогда '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' обрабатываются одинаково и '::/0' будет разрешать любые IPv4 или IPv6 адреса. '0.0.0.0/0' можно использовать, чтобы разрешить любые IPv4 адреса.

Обратите внимание, что "IPv4-совместимые IPv6 адреса" (0000::/96 префикс) поддерживаются, но являются устаревшими согласно [RFC4291](#).

Например: Server=127.0.0.1,192.168.1.0/24,:1,2001:db8::/32,zabbix.domain

Пробелы и [пользовательские макросы](#) разрешены в этом поле с Zabbix 2.2.0.

Возможно вам придется подождать до 60 секунд после сохранения элемента данных пока сервер подхватит изменения, до того как вы сможете отправлять значения.

Отправка данных

В самом простом случае, вы можете использовать утилиту [zabbix_sender](#) для отправки некоторого 'тестового значения' траппер элементу данных.

```
zabbix_sender -z <server IP address> -p 10051 -s "New host" -k trap -o "test value"
```

Для отправки значения мы используем следующие ключи:

-z - чтобы указать IP адрес Zabbix сервера

-p - чтобы указать номер порта Zabbix сервера (10051 по умолчанию)

-s - чтобы указать узел сети (убедитесь, что используете здесь 'техническое' [имя узла сети](#), вместо 'видимого' имени)

-k - чтобы указать ключ элемента данных, который мы только что добавили

-o - чтобы указать фактически отправляемое значение

Процесс Zabbix траппера не раскрывает макросы, которые используются в ключе элемента данных в попытке проверки наличия соответствующего ключа элемента данных на целевом узле сети.

Отображение

Результат в Мониторинг → Последние данные:

HOST	NAME	LAST CHECK	LAST VALUE	CHANGE
New host	- other - (2 items)			
	Trapper item	2015-08-11 18:50:53	test value	History

Штампы времени

Если значения отправляются с использованием утилиты `zabbix_sender` из файла со штампами времени, тогда эти штампы времени будут скорректированы со временем сервера. Например, если штамп времени элемента данных равен “10:30:50”, текущее время на машине откуда запускается `zabbix_sender` - “10:40:03”, и текущее время на Zabbix сервере “10:40:05”, тогда значение элемента данных будет сохранено в базу данных со штампом времени равным “10:30:52”.

Аналогично, если значение сначала отправляется на Zabbix прокси, который далее отправляет его Zabbix серверу, штамп времени будет скорректирован в соответствии со временем Zabbix прокси, и потом будет скорректирован со временем Zabbix сервера.

7.2.2.14 JMX мониторинг

Обзор

Мониторинг JMX можно использовать для наблюдения за счетчиками JMX в Java приложениях.

В Zabbix 2.0 добавлена встроенная поддержка мониторинга JMX, был выпущен новый Zabbix демон, так называемый “Zabbix Java gateway”.

Когда Zabbix сервер хочет узнать значение конкретного счетчика JMX у узла сети, он опрашивает Zabbix **Java gateway**, который в свою очередь используя [API управление JMX](#), удаленно опрашивает интересующее приложение.

Для получения более подробных сведений, включая где можно взять Zabbix Java gateway и как его настроить,смотрите [этот раздел](#) руководства.

Связь между Java gateway и наблюдаемым JMX приложением не должна быть закрыта брандмауэром.

Включение удаленного JMX мониторинга для Java приложений

Приложению Java не требуется какое-либо дополнительное установленное программное обеспечение, но для поддержки удаленного мониторинга JMX приложение должно быть запущено с указанными ниже параметрами командной строки.

Как минимум, если вы просто хотите начать наблюдение за простым приложением Java на локальном хосте без каких либо защиты, запустите его со следующими опциями:

```
java \
-Dcom.sun.management.jmxremote \
-Dcom.sun.management.jmxremote.port=12345 \
-Dcom.sun.management.jmxremote.authenticate=false \
-Dcom.sun.management.jmxremote.ssl=false \
-jar /usr/share/doc/openjdk-6-jre-headless/demo/jfc/Notepad/Notepad.jar
```

С этими аргументами Java будет слушать входящие соединения JMX на порту 12345, только с локальных хостов, без обязательных 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, есть много других опций в Java, которые вам доступны. Например, следующая иллюстрация запускает приложение с более универсальным набором опций и открывает это приложение для более широкой сети, 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 \
-Djavax.net.ssl.keyStore=$ВАШЕ_ХРАНИЛИЩЕ_КЛЮЧЕЙ \
-Djavax.net.ssl.keyStorePassword=$ВАШ_ПАРОЛЬ_К_ХРАНИЛИЩУ_КЛЮЧЕЙ \
-Djavax.net.ssl.trustStore=$ВАШЕ_ДОВЕРЕННОЕ_ХРАНИЛИЩЕ \
-Djavax.net.ssl.trustStorePassword=$ВАШ_ПАРОЛЬ_К_ДОВЕРЕННОМУ_ХРАНИЛИЩУ \
-Dcom.sun.management.jmxremote.ssl.need.client.auth=true \
-jar /usr/share/doc/openjdk-6-jre-headless/demo/jfc/Notepad/Notepad.jar
```

Значительное количество (если не все) этих настроек можно указать в /etc/java-6-openjdk/management/management.properties (или там, где этот файл расположен на вашем компьютере).

Обратите внимание, если вы желаете использовать SSL, то вы должны изменить startup.sh скрипта Java gateway, добавив в него опции -Djavax.net.ssl.* так, чтобы он знал где искать 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 удаленного JMX мониторинга, самое время настроить интерфейсы и элементы данных в Веб-интерфейсе Zabbix.

Настройка JMX интерфейса

Начнем с создания интерфейса JMX-типа у интересующего узла сети.

Hosts

Host Templates IPMI Macros Host inventory Encryption

* Host name

Visible name

* Groups In groups Other groups

Zabbix servers

Anna group
Annas group
bypass
calendarian
data poolers
Discovered hosts
group 1
group 2
Hypervisors
Linux servers

New group

* At least one interface must exist.

Agent interfaces

IP address	DNS name	Connect to	Port
127.0.0.1		IP	10050

Add

SNMP interfaces

JMX interfaces

IP address	DNS name	Connect to	Port
127.0.0.1		IP	12345

Add

Добавление элемента данных JMX агента

Для каждого интересующего вас счетчика JMX вам необходимо добавить элемент данных с типом **JMX агент** присоединенный к этому интерфейсу.

Ключ на снимке экрана ниже имеет следующий вид

`jmx["java.lang:type=Memory", "HeapMemoryUsage.used"]`.

Item Preprocessing

* Name	Used heap memory																
Type	JMX agent																
* Key	jmx["java.lang:type=Memory","HeapMemoryUsage.used"] <input type="button" value="Select"/>																
* Host interface	127.0.0.1 : 12345																
* JMX endpoint	service:jmx:rmi:///jndi/rmi://[HOST.CONN]:[HOST.PORT]/jmxrmi																
User name	{JMX_USERNAME}																
Password	{JMX_USERNAME}																
Type of information	Numeric (unsigned)																
Units																	
* Update interval	30s																
Custom intervals	<table border="1"> <thead> <tr> <th>Type</th> <th>Interval</th> <th>Period</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>Flexible</td> <td>Scheduling</td> <td>50s</td> <td>1-7.00:00-24:00</td> </tr> <tr> <td colspan="4">Remove</td> </tr> <tr> <td colspan="4">Add</td> </tr> </tbody> </table>	Type	Interval	Period	Action	Flexible	Scheduling	50s	1-7.00:00-24:00	Remove				Add			
Type	Interval	Period	Action														
Flexible	Scheduling	50s	1-7.00:00-24:00														
Remove																	
Add																	
* History storage period	90d																
* Trend storage period	365d																
Show value	As is <input type="button" value="show value mappings"/>																
New application																	
Applications	-None-																
Populates host inventory field	-None-																
Description																	
Enabled	<input checked="" type="checkbox"/>																
	<input type="button" value="Add"/> <input type="button" value="Cancel"/>																

Все обязательные поля ввода отмечены красной звёздочкой.

Поля, требующие специфичной информации для JMX элементов данных:

Тип Укажите здесь **JMX агент**.

Ключ элемента данных `jmx[]` состоит из двух параметров:

имя объекта - имя объекта MBean;

имя атрибута - имя атрибута MBean с опциональными составными именами полей, разделенных точками.

Ключ

Смотрите ниже для получения более подробных сведений о ключах элементов данных JMX.

Начиная с Zabbix 3.4, вы можете обнаруживать MBeans и MBean атрибуты, используя элемент данных `jmx.discovery[]` [низкоуровневого обнаружения](#).

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 это сделано в JMX endpoint по умолчанию.

Это поле поддерживается начиная с 3.4.0. Поддерживаются [макросы](#) {HOST.*} и пользовательские макросы.

Имя пользователя	Укажите имя пользователя, если вы настроили аутентификацию у вашего Java приложения. Поддерживаются пользовательские макросы.
Пароль	Укажите пароль, если вы настроили аутентификацию у вашего Java приложения. Поддерживаются пользовательские макросы.

Если вы хотите наблюдать за Логическим счетчиком, который может быть “true” или “false”, вы должны указать тип информации “Числовой (целое положительное)” и “Логический” тип данных. Сервер будет записывать Логические значения как 1 или 0, соответственно.

Детальная информация о ключах JMX элементов данных

Простые атрибуты

Имя объекта MBean неважно, кроме строки, которую вы определили в вашем Java приложении. Имя атрибута, с другой стороны, может быть более сложным. В случае, если атрибут возвращает простой тип данных (число, строку и т.п.), то не стоит волноваться об этом, ключ будет выглядеть примерно так:

```
jmx[com.example>Type=Hello,weight]
```

В этом примере именем объекта является “com.example>Type=Hello”, именем атрибута будет являться “weight” и, скорее всего, тип возвращаемого значения 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 хэш 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 хэша. Точно также, если атрибут возвращает часть вложенных составных данных, их нужно снова разделить точкой:

```
jmx[com.example>Type=Hello,fruits.apple.weight]
```

Проблема с точками

Пока все хорошо. Но что, если имя атрибута или ключ хэша 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?

До 2.0.4 Zabbix Java gateway был не способен справится с такими ситуациями и пользователи оставались с НЕПОДДЕРЖИВАЕМЫМИ элементами данных. Начиная с 2.0.4 проблема была исправлена, все 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 хэша содержит точку вам необходимо её экранировать:

```
jmx[com.example>Type=Hello,all\fruits.apple.total\weight]
```

Другие проблемы

Символ обратной косой черты тоже должен быть экранирован:

```
jmx[com.example:type=Hello,c:\\documents]
```

Для обработки любых других символов в ключе JMX элемента данных, пожалуйста,смотрите [раздел](#) формата ключа элементов данных.

На самом деле это все, что нужно сделать. Успешного мониторинга JMX!

Непримитивные типы данных

Начиная с Zabbix 4.0.0 имеется возможность работы с пользовательскими MBean, которые возвращают непримитивные типы данных, которые переопределяют метод **toString()**.

Пример пользовательского endpoint с JBoss EAP 6.4

Пользовательские endpoint позволяют работать с различными транспортными протоколами, которые отличаются от протокола по умолчанию RMI.

Для иллюстрации этой возможности в качестве примера давайте попытаемся настроить JBoss EAP 6.4. Во-первых, давайте сделаем некоторые предположения:

- У вас уже имеется установленный Zabbix Java gateway. Если нет, тогда вам нужно сделать это в соответствии с [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 будем считать, что все эти компоненты работают на одном и том же хосте.
- Брандмауэр и SELinux отключены (или настроены соответствующим 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
```

Проверьте, 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.

Hosts

All hosts / jboss Enabled ZBX SNMP JMX IPMI Applications 8 Items 55 Triggers 26 Graphs 11 Discovery rules Web scenarios

Host Templates IPMI Macros Host inventory Encryption

Host name: jboss
Visible name:

Groups: In groups: Linux servers; Other groups: Discovered hosts, Hypervisors, Templates, Templates/Applications, Templates/Databases, Templates/Modules, Templates/Network Devices, Templates/Operating Systems, Templates/Servers Hardware

New group:

Agent interfaces: IP address: 127.0.0.1, DNS name:, Connect to: IP, Port: 10050, Default, Remove

SNMP interfaces: Add

JMX interfaces: IP address: 127.0.0.1, DNS name:, Port: 9999, Default, Remove (highlighted with a red box and arrow)

Как мы знаем эта версия JBoss использует протокол JBoss Remoting вместо RMI, мы можем использовать массовое обновление параметра JMX endpoint в нашем шаблоне JMX в соответствии:
service:jmx:remoting-jmx://{{HOST.CONN}}:{HOST.PORT}

Items

All templates / Template App Generic Java JMX-remoting Applications 8 Items 55 Triggers 26

Type: Original

JMX endpoint: 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$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gateway-3.4.2.jar:na]
    at java.util.concurrent.Executors$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$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gateway-3.4.2.jar:na]
    at java.util.concurrent.Executors$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) has stopped
2017-11-07 13:52:26.167 [main] INFO com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885) has started

```

“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</pre>

```

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/local/sbin/zabbix_java/lib/ \; ; done
```

Таким образом, у Java gateway будут в наличии все необходимые модули для работы с jmx-remoting. Осталось только перезапустить Java gateway, немного подождать и, если вы все сделали правильно, вы увидите, что эти данные JMX мониторинга начинают поступать в Zabbix:

	Last check	Last value	Change
Name			
Classes (3 items)			
cl Loaded Class Count	2017-11-07 14:08:13	7868	+2
cl Total Loaded Class Count	2017-11-07 14:08:09	7865	+2
cl Unloaded Class Count	2017-11-07 14:08:13	0	
Compilation (2 items)			
comp Accumulated time spent in compilation	2017-11-07 14:08:13	49s 758ms	+1s 440ms
comp Name of the current JIT compiler	2017-11-07 14:00:39	HotSpot 64-Bit Tiered Compilers	
Garbage Collector (4 items)			
gc Copy accumulated time spent in collection	2017-11-07 14:08:09	0	
gc Copy number of collections per second	2017-11-07 14:08:09	0	
gc MarkSweepCompact accumulated time spent in collection	2017-11-07 14:08:13	372ms	
gc MarkSweepCompact number of collections per second	2017-11-07 14:08:13	0	
Memory (8 items)			
mem Heap Memory committed	2017-11-07 14:08:13	1.23 GB	
mem Heap Memory max	2017-11-07 14:00:39	1.23 GB	
mem Heap Memory used	2017-11-07 14:08:09	271.07 MB	+4.01 MB
mem Non-Heap Memory committed	2017-11-07 14:08:13	66.38 MB	+364 KB
mem Non-Heap Memory used	2017-11-07 14:08:13	59.5 MB	+128.1 KB
mem Object Pending Finalization Count	2017-11-07 14:08:13	0	
Memory Pool (6 items)			
mp Code Cache committed	2017-11-07 14:08:09	12.31 MB	+128 KB
mp Code Cache max	2017-11-07 14:00:40	240 MB	
mp Code Cache used	2017-11-07 14:08:09	12.23 MB	+145.64 KB
mp Tenured Gen committed	2017-11-07 14:08:13	869.38 MB	
mp Tenured Gen max	2017-11-07 14:00:40	869.38 MB	
mp Tenured Gen used	2017-11-07 14:08:09	32.26 MB	

7.2.2.15 ODBC мониторинг

Обзор

ODBC мониторинг соответствует типу элемента данных *Монитор баз данных* в веб-интерфейсе Zabbix.

ODBC - язык программирования на C, промежуточная прослойка API для доступа к системам управления баз данных (DBMS). Концепт ODBC был разработан Microsoft и в дальнейшем портирован на другие платформы.

Zabbix может выполнять запросы к любой базе данных, которая поддерживается ODBC. Чтобы это сделать, Zabbix не подключается напрямую к базам данных, он использует интерфейс ODBC и драйвера установленные в ODBC. Эта функция позволяет мониторить различные базы данных с различными целями с большей эффективностью - например, проверка специфичных запросов к базе данных, статистика использования и прочее. Zabbix поддерживает unixODBC, которая наиболее часто используется в реализациях ODBC API с открытым исходным кодом.

Установка unixODBC

Предлагаемый вариант установки unixODBC состоит из использования репозитариев пакетов по умолчанию в Linux операционной системы. В наиболее популярные дистрибутивы Linux unixODBC включен в репозитарии пакетов по умолчанию. Если он недоступен, вы можете обратиться к домашней странице UnixODBC: <http://www.unixodbc.org/download.html>.

Установка unixODBC на системы на базе RedHat/Fedora с использованием менеджера пакетов yum:

```
shell> yum -y install unixODBC unixODBC-devel
```

Установка unixODBC на системы на базе SUSE с использованием менеджера пакетов zypper:

```
# zypper in unixODBC-devel
```

Пакет unixODBC-devel требуется для компиляции Zabbix с поддержкой unixODBC.

Установка драйверов unixODBC

Драйвер unixODBC базы данных должен быть установлен для базы данных, которая будет наблюдаваться. unixODBC имеет список поддерживаемых баз данных и драйверов:

<http://www.unixodbc.org/drivers.html>. В некоторых дистрибутивах Linux драйвера баз данных включены в репозитарии пакетов. Драйвера MySQL базы данных на системы на базе RedHat/Fedora можно установить с помощью менеджера пакетов yum:

```
shell> yum install mysql-connector-odbc
```

Установка MySQL драйвера на системы на базе SUSE с использованием менеджера пакетов zypper:

```
# zypper in MyODBC-unicODBC
```

Настройка unixODBC

Настройка ODBC выполняется редактированием файлов **odbcinst.ini** и **odbc.ini**. Для проверки размещения этих файлов введите:

```
shell> odbcinst -j
```

odbcinst.ini используется для перечисления установленных драйверов баз данных ODBC:

```
[mysql]
Description = ODBC for MySQL
Driver      = /usr/lib/libmyodbc5.so
```

Подробная информация:

Атрибут	Описание
<i>mysql</i>	Имя драйвера базы данных.
<i>Description</i>	Описание драйвера базы данных.
<i>Driver</i>	Размещение библиотеки драйвера базы данных.

odbc.ini используется для определения источников данных:

```
[test]
Description = MySQL test database
Driver      = mysql
Server      = 127.0.0.1
User        = root
Password    =
Port        = 3306
Database   = zabbix
```

Подробная информация:

Атрибут	Описание
<i>test</i>	Имя источника данных (DSN).
<i>Description</i>	Описание источника данных.
<i>Driver</i>	Имя драйвера базы данных - как указано в odbcinst.ini
<i>Server</i>	IP/DNS сервера базы данных.
<i>User</i>	Пользователь базы данных для подключения.
<i>Password</i>	Пароль к базе данных.
<i>Port</i>	Порт подключения к базе данных.
<i>Database</i>	Имя базы данных.

Для проверки работает ли соединение ODBC корректно, подключение к базе данных необходимо протестировать. Для этого можно воспользоваться утилитой **isql** (включена в пакет unixODBC):

```
shell> isql test
+-----+
| Connected!
|
| sql-statement
| help [tablename]
```

```
| quit  
|  
+-----+  
SQL>
```

Компиляция Zabbix с поддержкой ODBC

Для включения поддержки ODBC, Zabbix должен быть скомпилирован со следующим флагом:

```
--with-unixodbc[=ARG]    use odbc driver against unixODBC package
```

Смотрите более подробную информацию о установке Zabbix из [исходных кодов](#).

Настройка элемента данных в веб-интерфейсе Zabbix

Настройка [элемента данных](#) для мониторинга базы данных.

The screenshot shows the configuration of a database monitoring item. The fields are as follows:

- Name: MySQL host count
- Type: Database monitor
- Key: db.odbc.select[mysql-simple-check,test]
- User name: zabbix
- Password: (empty)
- SQL query: select count(*) from hosts
- Type of information: Numeric (unsigned)

Все обязательные поля ввода отмечены красной звёздочкой.

Специально для элементов данных мониторинга баз данных вы должны указать:

Тип	Выберите здесь Монитор баз данных. Введите db.odbc.select[уникальное_описание,имя_источника_данных]
Ключ	Уникальное описание будет служить идентификатором элемента данных в триггерах и тому подобном.
Имя пользователя	Имя источника данных (DSN) должно быть указано как в настройках odbc.ini.
Пароль	Введите пароль пользователя для доступа к базе данных (опционально, если пароль указан в odbc.ini)
SQL запрос	Введите необходимый SQL запрос
Тип информации	Очень важно знать какой тип информации будет возвращаться указанным запросом, то есть выберите корректный тип информации здесь. С некорректным типом информации элемент данных станет неподдерживаемым.

Важные замечания

- Zabbix не ограничивает время выполнения запроса. Пользователь вправе выбирать запросы, которые могут быть выполнены в разумное время.

- Значение параметра [Timeout](#) с Zabbix сервера используется как время ожидания подключения ODBC (обратите внимание, в зависимости от драйвера ODBC время ожидания подключения может быть проигнорировано).
- Запрос должен возвращать только одно значение.
- Если запрос возвращает более чем одну колонку, будет прочитана только первая колонка.
- Если запрос возвращает более чем одну строку, будет прочитана только первая строка.
- Команда SQL должна начинаться с `select`.
- Команда SQL не должна включать в себя переводы строк.
- Смотрите также [известные проблемы](#) по ODBC проверкам

Сообщения об ошибках

Сообщения об ошибках ODBC скомпонованы в поля для предоставления подробной информации.

Например:

```
Cannot execute ODBC query: [SQL_ERROR]:[42601][7][ERROR: syntax error at or near ";"];
Error while executing the query]
```

Обратите внимание, что длина сообщения об ошибке ограничена 2048 байтами, поэтому сообщение может быть укорочено. Если есть более одной ODBC диагностической записи, Zabbix пытается их скомпоновать (разделяя символом |) на сколько позволяет максимальная длина сообщения.

7.2.2.15.1 Рекомендуемые настройки UnixODBC для MySQL

Установка

- **Red Hat Enterprise Linux/CentOS:**

```
# yum install mysql-connector-odbc
```

- **Debian/Ubuntu:**

Пожалуйста, обратитесь к [документации MySQL](#), чтобы загрузить необходимый драйвер базы данных для соответствующей платформы.

Для получения дополнительной информации, пожалуйста, обратитесь к: [Установка unixODBC](#).

Настройка

Настройка ODBC выполняется изменением **`odbcinst.ini`** и **`odbc.ini`** файлов. Эти файлы конфигурации можно найти в `/etc` папке. Файл **`odbcinst.ini`** может отсутствовать и в этом случае его необходимо создать вручную.

`odbcinst.ini`

```
[mysql]
Description = General ODBC for MySQL
Driver      = /usr/lib64/libmyodbc5.so
Setup       = /usr/lib64/libodbcmyS.so
FileUsage   = 1
```

Пожалуйста, обратите внимание на следующие примеры параметров конфигурации **`odbc.ini`**.

- Пример подключения по IP адресу:

```
[TEST_MYSQL]
Description = MySQL database 1
Driver = mysql
Port = 3306
Server = 127.0.0.1
```

- Пример подключения по IP адресу и с использованием учетной записи. По умолчанию используется база данных zabbix:

```
[TEST_MYSQL_FILLED_CRED]
Description = MySQL database 2
Driver = mysql
User = root
Port = 3306
Password = zabbix
Database = zabbix
Server = 127.0.0.1
```

- Пример подключения через сокет и с использованием учетной записи. По умолчанию используется база данных zabbix:

```
[TEST_MYSQL_FILLED_CRED_SOCK]
Description = MySQL database 3
Driver = mysql
User = root
Password = zabbix
Socket = /var/run/mysql/mysqld.sock
Database = zabbix
```

Все остальные возможные опции параметров конфигурации можно найти на странице [официальной документации MySQL \[en\]](#).

7.2.2.15.2 Рекомендуемые настройки UnixODBC для PostgreSQL

Установка

- Red Hat Enterprise Linux/CentOS:

```
# yum install postgresql-odbc
```

- Debian/Ubuntu:

Пожалуйста, обратитесь к [документации PostgreSQL](#), чтобы загрузить необходимый драйвер базы данных для соответствующей платформы.

Для получения дополнительной информации, пожалуйста, обратитесь к: [Установка unixODBC](#).

Настройка

Настройка ODBC выполняется изменением **odbcinst.ini** и **odbc.ini** файлов. Эти файлы конфигурации можно найти в */etc* папке. Файл **odbcinst.ini** может отсутствовать и в этом случае его необходимо создать вручную.

Пожалуйста, обратите внимание на следующие примеры:

odbcinst.ini

```
[postgresql]
Description = General ODBC for PostgreSQL
Driver      = /usr/lib64/libodbcpgsql.so
Setup       = /usr/lib64/libodbcpgsqlS.so
FileUsage   = 1
# Since 1.6 if the driver manager was built with thread support you may add another
entry to each driver entry.
```

```
# This entry alters the default thread serialization level.  
Threading = 2
```

odbc.ini

```
[TEST_PSQL]  
Description = PostgreSQL database 1  
Driver = postgresql  
#CommLog = /tmp/sql.log  
Username = zbx_test  
Password = zabbix  
# Name of Server. IP or DNS  
Servername = 127.0.0.1  
# Database name  
Database = zabbix  
# Postmaster listening port  
Port = 5432  
# Database is read only  
# Whether the datasource will allow updates.  
ReadOnly = No  
# PostgreSQL backend protocol  
# Note that when using SSL connections this setting is ignored.  
# 7.4+: Use the 7.4(V3) protocol. This is only compatible with 7.4 and higher backends.  
Protocol = 7.4+  
# Includes the OID in SQLColumns  
ShowOidColumn = No  
# Fakes a unique index on OID  
FakeOidIndex = No  
# Row Versioning  
# Allows applications to detect whether data has been modified by other users  
# while you are attempting to update a row.  
# It also speeds the update process since every single column does not need to be  
specified in the where clause to update a row.  
RowVersioning = No  
# Show SystemTables  
# The driver will treat system tables as regular tables in SQLTables. This is good for  
Access so you can see system tables.  
ShowSystemTables = No  
# If true, the driver automatically uses declare cursor/fetch to handle SELECT  
statements and keeps 100 rows in a cache.  
Fetch = Yes  
# Booleans as Char  
# Booleans are mapped to SQL_CHAR, otherwise to SQL_BIT.  
BoolsAsChar = Yes  
# SSL mode  
SSLmode = Yes  
# Send to backend on connection  
ConnSettings =
```

7.2.2.15.3 Рекомендуемые настройки UnixODBC для Oracle

Установка

Пожалуйста, для получения всех требуемых инструкций обратитесь к [документации Oracle](#).

Для получения дополнительной информации, пожалуйста, обратитесь к: [Установка unixODBC](#).

7.2.2.15.4 Рекомендуемые настройки UnixODBC для MSSQL

Установка

- Red Hat Enterprise Linux/CentOS:

```
# yum -y install freetds unixODBC
```

- Debian/Ubuntu:

Пожалуйста, обратитесь к [документации пользователя FreeTDS](#), чтобы загрузить необходимый драйвер базы данных для соответствующей платформы.

Для получения дополнительной информации, пожалуйста, обратитесь к: [Установка unixODBC](#).

Настройка

Настройка ODBC выполняется изменением **odbcinst.ini** и **odbc.ini** файлов. Эти файлы конфигурации можно найти в `/etc` папке. Файл **odbcinst.ini** может отсутствовать и в этом случае его необходимо создать вручную.

Пожалуйста, обратите внимание на следующие примеры:

odbcinst.ini

```
$ vi /etc/odbcinst.ini
[FreeTDS]
Driver = /usr/lib64/libtdsodbc.so.0
```

odbc.ini

```
$ vi /etc/odbc.ini
[sql1]
Driver = FreeTDS
Server = <SQL сервер 1 IP>
PORT = 1433
TDS_Version = 8.0
```

7.2.2.16 Зависимые элементы данных

Обзор

Имеются ситуации, когда один элемент данных собирает несколько метрик за раз или имеется больший смысл в одновременном сборе связанных метрик, например:

- Утилизация CPU отдельных ядер
- Входящий/исходящий/весь сетевой трафик

Для обеспечения массового сбора метрик и использования синхронности в нескольких связанных элементах данных, Zabbix поддерживает зависимые элементы данных. Зависимые элементы данных используют основной элемент данных, чтобы собрать свои данные одновременно, одним запросом. Новое значение у основного элемента данных автоматически заполняет значения и зависимых элементов данных.

Опции предобработки в Zabbix можно использовать для извлечения части, которая необходима зависимому элементу данных, из данных основного элемента данных.

Предварительная обработка управляется при помощи менеджера предобработки процесса, который добавлен в Zabbix 3.4, вместе с процессами, которые выполняют шаги предобработки. Все значения (с и без предварительной обработкой) от разных сборщиков данных проходят через менеджер предварительной обработки перед добавлением в кэш истории. Для связи между сборщиками данных (поллерами, трапперами и т.д.) и процессами предобработки используется межпроцессорное взаимодействие (IPC) на основе сокета.

Только Zabbix сервер выполняет шаги предварительной обработки и он же обрабатывает зависимые элементы данных.

Элемент данных любого типа, даже зависимый элемент данных, может быть основным элементом данных. Дополнительные уровни зависимых элементов данных можно использовать для извлечения меньших частей значения уже существующего зависимого элемента данных.

Ограничения

- Зависимости разрешены только в пределах одного узла сети (шаблона)
- Прототип элемента данных может зависеть от другого прототипа элемента данных или от обычного элемента данных с этого же узла сети
- Максимальное количество зависимых элементов данных от одного основного элемента данных ограничено 999 элементами данных
- Разрешено максимум 3 уровней зависимости
- Зависимый элемент данных вместе с основным элементом данных из шаблона не будет экспортиться в XML

Настройка элемента данных

Зависимый элемент данных зависит от его основного элемента данных. Поэтому сначала необходимо настроить (или использовать существующий) **основной элемент данных**:

- Перейдите в: *Настройка* → *Узлы сети*
- Нажмите на *Элементы данных* в строке с узлом сети
- Нажмите на *Создать элемент данных*
- В диалоге укажите параметры элемента данных

The screenshot shows the 'Preprocessing' tab of a configuration dialog. The fields are as follows:

* Name	Apache server status
Type	Zabbix agent
* Key	web.page.get[127.0.0.1,/server-status]
* Host interface	127.0.0.1 : 10050
Type of information	Text
* Update interval	30s

Все обязательные поля ввода отмечены красной звёздочкой.

Нажмите на *Добавить* для сохранения основного элемента данных.

Теперь вы можете настроить **зависимый элемент данных**.

The screenshot shows the 'Preprocessing' tab of a configuration dialog for a dependent item. The fields are as follows:

* Name	Apache server uptime
Type	Dependent item
* Key	apache.server.uptime
* Master item	Apache server status: web.page.get[127.0.0.1,/server-status]
Type of information	Text

Все обязательные поля ввода отмечены красной звёздочкой.

Следующие поля требуют особые параметры по зависимым элементам данных:

Тип	Выберите здесь Зависимый элемент данных .
Ключ	Укажите ключ, который будет использоваться для распознавания этого элемента данных.
Основной элемент данных	Выберите основной элемент данных. Значение основного элемента данных будет использовано для заполнения значения зависимого элемента данных.
Тип информации	Выберите тип информации, который соответствует формату сохраняемых данных.

Вы можете использовать значение [предобработки](#), чтобы извлечь требуемую часть значения основного элемента данных.

Preprocessing steps	Name	Parameters
Regular expression	<dt>Server uptime: (.*)</dt>	\1

Без предварительной обработки значением зависимого элемента данных будет тем же значением, что и значение основного элемента данных.

Нажмите на *Добавить*, чтобы сохранить зависимый элемент данных.

В списке элементов данных при помощи быстрого доступа создания зависимого элемента данных можно использовать помощника:

Отображение

В списке элементов данных зависимые элементы данных отображают с префиксом имени основного элемента данных.

Name	Triggers	Key
Apache server status		web.page.get[192.168.3.31/server-status]
Apache server status: Apache server uptime		apache.server.uptime

Если основной элемент данных удаляется, тогда удаляются и все его зависимые элементы данных.

7.2.2.17 HTTP агент

Обзор

Этот тип элемента данных позволяет получать данные с использованием HTTP/HTTPS протокола. Трапы также возможны с использованием Zabbix sender или протокола Zabbix sender.

Проверка HTTP элемента данных выполняется на стороне Zabbix сервера. Однако, когда 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).

Смотрите также [известные проблемы](#) при использовании HTTPS протокола.

Zabbix сервер/прокси должны быть изначально сконфигурированы с поддержкой cURL (libcurl).

Настройка

Для настройки HTTP элемента данных:

- Перейдите в: *Настройка* → *Узлы сети*
- Нажмите на *Элементы данных* в строке с узлом сети
- Нажмите на *Создать элемент данных*
- Введите параметры элемента данных в диалоге

Item Preprocessing

* Name	HTTP agent item												
Type	HTTP agent												
* Key	http_value_search												
* URL	http://localhost:9200/_str/values/_search												
Query fields													
Name	Value												
scroll	10s												
Add													
Request type	POST												
Timeout	3s												
Request body type	Raw data JSON data XML data												
Request body	<pre>{ "query": { "bool": { "must": [{ "match": { "itemid": 28275 } }] } } }</pre>												
Headers	<table border="1"> <tr> <td>Name</td> <td>Value</td> </tr> <tr> <td>name</td> <td>value</td> </tr> <tr> <td colspan="2">Add</td> </tr> </table>	Name	Value	name	value	Add							
Name	Value												
name	value												
Add													
Required status codes	200												
Follow redirects	<input type="checkbox"/>												
Retrieve mode	Body Headers Body and headers												
Convert to JSON	<input type="checkbox"/>												
HTTP proxy	http://user[:password]@[]proxy.example.com[:port]												
HTTP authentication	None												
SSL verify peer	<input type="checkbox"/>												
SSL verify host	<input type="checkbox"/>												
SSL certificate file													
SSL key file													
SSL key password													
* Host interface	127.0.0.1 : 10050												
Type of information	Numeric (unsigned)												
Units													
* Update interval	30s												
Custom intervals	<table border="1"> <thead> <tr> <th>Type</th> <th>Interval</th> <th>Period</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>Flexible</td> <td>50s</td> <td>1-7.00:00-24:00</td> <td>Remove</td> </tr> <tr> <td colspan="4">Add</td> </tr> </tbody> </table>	Type	Interval	Period	Action	Flexible	50s	1-7.00:00-24:00	Remove	Add			
Type	Interval	Period	Action										
Flexible	50s	1-7.00:00-24:00	Remove										
Add													
* History storage period	90d												
* Trend storage period	365d												
Show value	As is												
show value mappings													
Enable trapping	<input checked="" type="checkbox"/>												
Allowed hosts	104.24.103.152												
New application													

Все обязательные поля ввода отмечены красной звёздочкой.

Поля, которые требуют специфичную информацию HTTP элементов данных:

Tip	Выберите здесь HTTP агент .
Ключ	Введите уникальный ключ элемента данных.
URL	<p>URL для подключения и получения данных. Например: https://www.google.com http://www.zabbix.com/download</p> <p>Имена доменов можно указывать Юникод символами. Они автоматически конвертируются методом punycode в ASCII при выполнении шага веб-сценария. Кнопку <i>Анализ</i> можно использовать чтобы отделить из URL optionalные поля</p>

Поля запроса	запроса (например, ?name=Admin&password=mypassword), переместив атрибуты и значения в <i>Поля запроса</i> , чтобы URL кодировка выполнялась автоматически. Ограничено 2048 символами. Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения. Это поле задает CURLOPT_URL [en] опцию cURL. Переменные для URL (см. выше). Задаются в виде пар атрибутов и значений. Значения URL кодируются автоматически. Значения с макросов раскрываются и затем URL кодируются автоматически. Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения. Это поле задает CURLOPT_URL [en] опцию cURL.
Тип запроса	Выберите тип метода запроса: <i>GET</i> , <i>POST</i> , <i>PUT</i> или <i>HEAD</i> Zabbix не будет тратить более указанного времени при обработке URL (максимум 1 минута). В действительности же этот параметр определяет максимальное время содания подключения к URL и максимальное время для выполнения HTTP запроса. Следовательно, Zabbix не будет тратить более 2 x Время ожидания секунд за одну проверку.
Время ожидания	Поддерживаются суффиксы времени , например, 30s, 1m. Поддерживаемые макросы: пользовательские макросы, макросы низкоуровневого обнаружения. Это поле задает CURLOPT_TIMEOUT [en] опцию cURL. Выберите тип запроса тела: Сырые данные - пользовательское тело HTTP запроса, макросы заменяются, но кодирование не выполняется JSON данные - тело HTTP запроса в формате JSON. Макросы можно использовать в виде строки, числа, true или false; макросы, которые используются в виде строки должны быть заключены в двойные кавычки. Значения из макросов раскрываются и затем экранируются автоматически. Если в заголовках "Content-Type" не задан, тогда будет указано значение по умолчанию "Content-Type: application/json" XML данные - тело HTTP запроса в формате XML. Макросы можно использовать в виде нод текста, атрибутов или CDATA секции. Значения из макросов раскрываются и затем экранируются автоматически в нодах текста и в атрибутах. Если в заголовках "Content-Type" не задан, тогда будет указано значение по умолчанию "Content-Type: application/xml" <i>Обратите внимание</i> , что выбор XML данные требует наличия поддержки libxml2.
Тип запроса тела	Введите тело запроса. Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения. Пользовательские HTTP заголовки, которые будут отправлены при выполнении запроса. Задаются в виде пар атрибутов и значений. Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения. Это поле задает CURLOPT_HTTPHEADER [en] опцию cURL.
Запрос тела	Список ожидаемых кодов состояния HTTP. Если Zabbix получает код не из списка, то элемент данных станет неподдерживаемым. Если поле не заполнено,
Заголовки	
Требуемые коды состояния	

<i>Следовать перенаправлениям</i>	<p>то проверка не производится. Например: 200,201,210-299 Поддерживаемые макросы в этом списке: пользовательские макросы, макросы низкоуровневого обнаружения. Это поле задает CURLINFO_RESPONSE_CODE [en] опцию cURL.</p>
<i>Режим получения</i>	<p>Отметьте для следования по HTTP перенаправлениям. Это поле задает CURLOPT_FOLLOWLOCATION [en] опцию cURL.</p>
	<p>Выберите часть ответа, которую необходимо получать: Тело - только тело Заголовки - только заголовки Тело и заголовки - тело и заголовки</p>
	<p>Заголовки сохраняются в виде пар атрибутов и значений под ключем "header". Если встречается 'Content-Type: application/json', тогда тело сохраняется в виде объекта, в противном случае оно сохраняется в виде строки, например:</p>
	<pre>{ "header": { "<key>": "<value>", "<key2>": "<value>" }, "body": <body> }</pre>
	<p>Вы можете указать необходимый HTTP прокси, следуя следующему формату: <i>http://[имя пользователя[:пароль]@]прокси.тускомпани.com[:порт]</i></p>
	<p>По умолчанию будет использоваться порт 1080.</p>
	<p>Если указан, прокси заменит переменные окружения связанные с прокси такие как <code>http_proxy</code>, <code>HTTPS_PROXY</code>. Если не указан, переменные окружения не будут заменены. Введённое значение передается "как есть", проверка правильности не производится.</p>
	<p>Вы также можете указать адрес SOCKS прокси. Если вы укажите ошибочный протокол, подключение провалится и элемент данных станет неподдерживаемым. Если протокол не указан, прокси будет считаться HTTP прокси.</p>
	<p>Примечание: Для HTTP прокси поддерживается только простая аутентификация.</p>
	<p>Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения.</p>
	<p>Это поле задает CURLOPT_PROXY [en] опцию cURL.</p>
	<p>Тип аутентификации:</p>
	<p>Пусто - без использования аутентификации.</p>
	<p>Простая аутентификация - с использованием простой аутентификации.</p>
	<p>NTLM аутентификация - с использованием NTLM (Windows NT LAN Manager) аутентификации.</p>
	<p>Выбрав какой-либо метод аутентификации, будут доступны доступны два дополнительных поля для ввода имени пользователя и пароля, в которых поддерживаются пользовательские макросы и макросы низкоуровневого обнаружения.</p>
	<p>Это поле задает CURLOPT_HTTPAUTH [en] опцию cURL.</p>
	<p>Отметьте для верификации SSL сертификата веб-сервера. Сертификат сервера будет автоматически взят из места центра сертификации (CA) всей системы. Вы можете перезаписать расположение CA файлов, используя параметр конфигурации <code>SSLCALocation</code> в Zabbix сервере или прокси.</p>
	<p>Это поле задает CURLOPT_SSL_VERIFYPEER [en] опцию cURL.</p>
	<p>Отметьте для верификации, что поле <i>Common Name</i> или поле <i>Subject Alternate Name</i> сертификата веб-сервера совпадают.</p>

Файл SSL сертификата

Это поле задает [CURLOPT_SSL_VERIFYHOST](#) [en] опцию cURL.

Имя файла SSL сертификата для аутентификации клиента. Файл сертификата должен быть в формате PEM¹. Если файл сертификата также содержит и приватный ключ, 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 или прокси.

Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения.

Это поле задает [CURLOPT_SSLCERT](#) [en] опцию cURL.

Файл SSL ключа

Имя файла приватного SSL ключа, который используется для аутентификации клиента. Файл приватного ключа должен быть в формате PEM¹. Папка, содержащая этот файл указывается в параметре конфигурации SSLKeyLocation Zabbix сервера или прокси.

Поддерживаемые макросы: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, пользовательские макросы, макросы низкоуровневого обнаружения.

Это поле задает [CURLOPT_SSLKEY](#) [en] опцию cURL.

Пароль к файлу приватного ключа.

Поддерживаемые макросы: пользовательские макросы, макросы низкоуровневого обнаружения.

Это поле задает [CURLOPT_KEYPASSWD](#) [en] опцию cURL.

Пароль к SSL ключу

При наличии этой опции, элемент данных будет работать как [траппер элемент данных](#) и будет принимать данные, которые отправляются на этот элемент данных при помощи Zabbix sender или с использованием протокола Zabbix sender.

Видимо только, если выбрана опция Активировать трапы.

Список разделенных запятыми IP адресов, дополнительно в CIDR представлении, или имена узлов сети.

Если задано, входящие подключения будут приниматься только с перечисленных здесь узлов сети.

Если включена поддержка IPv6, тогда '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' обрабатываются одинаково и '::/0' будет разрешать любые IPv4 или IPv6 адреса. '0.0.0.0/0' можно использовать, чтобы разрешить любые IPv4 адреса.

Обратите внимание, что “IPv4-совместимые IPv6 адреса” (0000::/96 префикс) поддерживаются, но являются устаревшими согласно [RFC4291](#).

Пример: Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.domain

В этом поле разрешены пробелы и [пользовательские макросы](#).

Если поле HTTP прокси оставить пустым, можно воспользоваться другим способом указать HTTP прокси, для этого необходимо задать переменные окружения.

Для HTTP проверок - укажите переменную окружения **http_proxy** для пользователя Zabbix сервера. Например, *http_proxy=http://proxy_ip:proxy_port*.

Для HTTPS проверок - укажите переменную окружения **HTTPS_PROXY**. Например, *HTTPS_PROXY=http://proxy_ip:proxy_port*. Более подробную информацию можно получить, выполнив в shell команду *# man curl*.

[1] Zabbix поддерживает файлы сертификатов и приватных ключей только в PEM формате. В случае, если у вас имеются данные сертификата и приватного 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
```

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/_search?scroll=10s
- Настройте следующее тело POST, чтобы получать загрузку процессора (среднее значение за 1 по каждому ядру)

```
{
  "query": {
    "bool": {
      "must": [
        {
          "match": {
            "itemid": 28275
          }
        }
      ],
      "filter": [
        {
          "range": {
            "clock": {
              "gt": 1517565836,
              "lte": 1517566137
            }
          }
        }
      ]
    }
  }
}
```

- Получено:

```
{
  "_scroll_id": "DnF1ZXJ5VGhlbkZldGNoBQAAAAAAAAAkFllRMlZBWS1UU1pxTmdEeGVwQjRBTfEAAAAAAAJRZZUTJWQVktVF
```

```

NacU5nRHhlcEI0QUxRAAAAAAAAACYWWVEyVkfZLVRTWnFOZ0R4ZXBCNEFMUQAAAAAAAAnFllRMlZBWS1UU1pxT
mdEeGVwQjRBTFEAAAAAAAKBZZUTJWQVktVFNacU5nRHhlcEI0QUxR",
    "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 с использованием [apiinfo.version](#).

- Настройка элемента данных:

Item Preprocessing

* Name	Check Zabbix API version						
Type	HTTP agent ▾						
* Key	check_zabbix_api_info.version						
* URL	http://zabbix-web-apache-mysql/api_jsonrpc.php						
Query fields	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Add</td> <td></td> </tr> </tbody> </table>	Name	Value	Add			
Name	Value						
Add							
Request type	POST ▾						
Timeout	3s						
Request body type	Raw data JSON data XML data						
Request body	<pre>{ "jsonrpc": "2.0", "method": "apiinfo.version", "params": [], "id": 1 }</pre>						
Headers	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Content-Type</td> <td>application/json-rpc</td> </tr> <tr> <td>Add</td> <td></td> </tr> </tbody> </table>	Name	Value	Content-Type	application/json-rpc	Add	
Name	Value						
Content-Type	application/json-rpc						
Add							
Required status codes							
Follow redirects	<input type="checkbox"/>						
Retrieve mode	Body Headers Body and headers						

Обратите внимание, на использование POST метода с JSON данными, с настройкой заголовков в запросе и получением только заголовков:

- Предварительная обработка значений элемента данных при помощи регулярного выражения, чтобы получить HTTP код:

Item Preprocessing

Preprocessing steps	Name	Parameters
	Regular expression ▾	HTTPV1.1 ([0-9]+) \1
	Add	

- Проверка результата в *Последние данные*:

Latest data

Host groups: type here to search Select Name: Check Zabbix API

Hosts: Zabbix server × nginx × Select Show items without data
Application: Select Show details

Apply Reset

Host	Name	Last check	Last value	Change
Zabbix server	- other - (1 item)			
	Check Zabbix API version	2018-05-16 23:50:34	OK (200)	Graph

Пример 4

Получение информации о погоде при помощи подключения к публичному сервису Openweathermap.

- Настройка основного элемента данных для массового сбора данных в одном JSON:

Parent items: Template Weather

* Name: Get weather

Type: HTTP agent

* Key: get_weather.http

* URL: http://api.openweathermap.org/data/2.5/weather

Query fields

Name	Value
units	= metric
lat	= {\$LAT}
lon	= {\$LON}
APPID	= {\$WEATHER_APIKEY}
lang	= {\$WEATHER_LANG}

Add

Request type: GET

Timeout: 3s

Request body type: Raw data

Request body:

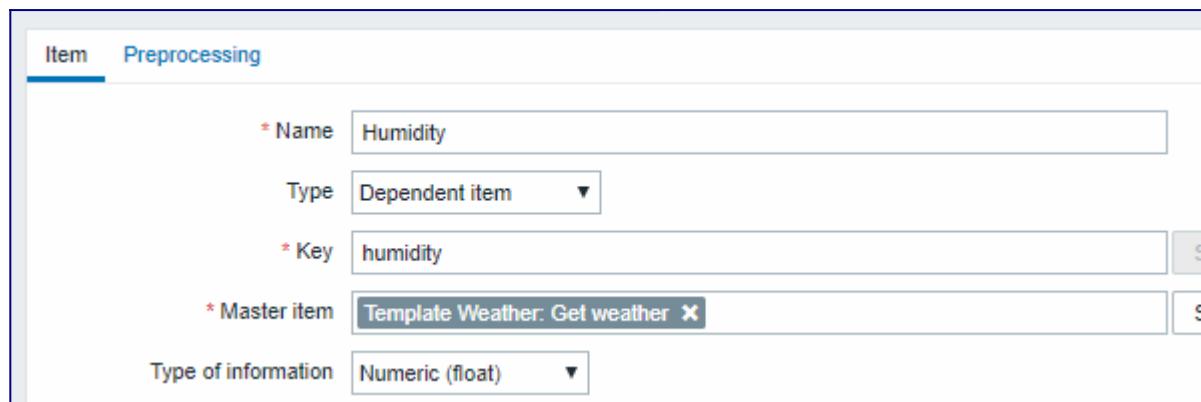
Обратите внимание на использование макросов в полях запроса. Обратитесь к [Openweathermap API](#) для получения подробных сведений о том как заполнять их.

Пример JSON, который вернулся в ответе на HTTP агент:

```
{
  "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 с 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:

Item Preprocessing

Preprocessing steps	Name	Parameters	Action
	JSON Path	\$body.main.humidity	Remove
Add			

- Проверка результата данных о погоде в *Последние данные*:

Host	Name	Inter...	History	Trends	Type	Last check	Last value
weather	Weather (8 items)						
	Get weather get_weather.http	10m	1d		HTTP agent	2018-05-17 01:23:45	{"body": {"coord": {"lon": ...}}
	Get weather HTTP response code get_weather.http_code	7d	0	Depende...		2018-05-17 01:23:45	OK (200)
	Humidity humidity	90d	365d	Depende...		2018-05-17 01:23:45	66 %
	Temperature temp	90d	365d	Depende...		2018-05-17 01:23:45	15.14 C
	Weather weather	90d		Depende...		2018-05-17 01:23:45	Clouds
	Weather condition id weather.condition.id	7d	0	Depende...		2018-05-17 01:23:45	801
	Weather description weather.description	90d		Depende...		2018-05-17 01:23:45	few clouds
	Wind speed wind.speed	90d	365d	Depende...		2018-05-17 01:23:45	1.86 m/s

Пример 5

Подключение в странице состояния Nginx и получения его метрик за один запрос.

- Настройте Nginx, следуя [официальному руководству](#).
- Настройте основной элемент данных для массового сбора данных:

Item Preprocessing

* Name	Get NGINX status page				
Type	HTTP agent				
* Key	get_nginx				
* URL	http://(HOST.CONN)/nginx_status				
Query fields	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> Add	Name	Value	name	value
Name	Value				
name	value				
Request type	GET				
Timeout	3s				
Request body type	<input checked="" type="radio"/> Raw data <input type="radio"/> JSON data <input type="radio"/> XML data				
Request body					

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:

The screenshot shows the 'Item' configuration page. The 'Preprocessing' tab is selected. The 'Name' field contains 'Client requests per second'. The 'Type' dropdown is set to 'Dependent item'. The 'Key' field contains 'nginx_requests_rps'. The 'Master item' dropdown is set to 'Template App Nginx HTTP: Get Nginx stub status'. The 'Type of information' dropdown is set to 'Numeric (unsigned)'. There are two 'Save' buttons at the bottom right.

- Настройка предобработки значений зависимого элемента данных при помощи регулярного выражения:

The screenshot shows the 'Preprocessing' configuration for the 'Client requests per second' item. It lists two steps: 'Regular expression' with the parameter 'server accepts handled requests\\$(\d{0}\.\d{1}) (\d{0}\.\d{1}) (\d{0}\.\d{1})' and 'Change per second'.

- Проверьте полученный результат от модуля stub в *Последние данные*:

Host	Name	Last check	Last value
nginx	Nginx (8 items)		
	Accepted client connections	2018-05-18 17:54:53	568
	Active connections	2018-05-18 17:54:53	1
	Client requests per second	2018-05-18 17:54:53	0 rps
	Get Nginx stub status	2018-05-18 17:54:53	HTTP/1.1 200 OK Se...
	Handled connections per second	2018-05-18 17:54:53	0
	Reading	2018-05-18 17:54:53	0
	Waiting	2018-05-18 17:54:53	0
	Writing	2018-05-18 17:54:53	1

7.2.3 История и динамика изменений

Обзор

История и динамика изменения - два пути хранения в Zabbix собранных данных.

В то время как история хранит каждое собранное значение, динамика изменений хранит усредненную информацию за каждый час, что, следовательно, менее ресурсоемко.

Хранение истории

Вы можете указать как много дней история будет храниться:

- 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).

Самый большой 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 и 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, графики будут использовать значения динамики изменений для отображения старых данных.

Если история установлена в '0', элемент данных будет обновлять только поля инвентаря. Функции триггера не будут вычисляться.

В качестве альтернативного подхода к хранению истории обратитесь к функционалу [экспорта истории](#) в подгружаемых модулях.

Хранение динамики изменений

Динамика изменений является встроенным механизмом обработки данных истории, где для каждого часа хранятся значения минимума, максимума и усредненное значения, а также общее количество значений за этот час.

Вы можете указать как много дней динамика изменений будет храниться:

- 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 с помощью автоматической очистки базы данных (Housekeeper).

Если динамика изменений установлена в '0', Zabbix сервер вообще не будет считать и хранить динамику изменений.

Динамика изменений вычисляется и сохраняется с использованием того же типа данных, как и оригинальные 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, средним значением будет 0, а не 0.5.

Перезапуск сервера также может привести к потере точности вычисления усредненных значений у целочисленных типов данных за текущий час.

7.2.4 Пользовательские параметры

Обзор

Иногда вы можете захотеть выполнять проверку через агент, которая не предопределена в Zabbix. В этом случае вам на помощь придут пользовательские параметры.

Вы можете написать команду, которая вернет необходимые вам данные и добавить эту команду как пользовательский параметр в [файл конфигурации агента](#) (параметр конфигурации 'UserParameter').

Пользовательский параметр имеет следующий синтаксис:

UserParameter=<ключ>, <команда>

Как вы можете видеть, пользовательский параметр также содержит ключ. Ключ потребуется при настройке элемента данных. Введите ключ на ваш выбор, на который будет легко ссылаться (он должен быть уникальным в пределах узла сети). Перезапустите агента.

Затем, когда [настроите элемент данных](#), введите ключ на который ссылается команда из пользовательского параметра, которую вы хотите выполнять.

Команды из пользовательских параметров выполняются Zabbix агентом. Может быть возвращено до 512КБ данных до шагов предварительной обработки элементов данных. Однако, обратите внимание на то, что такое текстовое значение, которое может быть сохранено в базе данных ограничено 64КБ в MySQL (смотрите информацию по остальным базам данных в [таблице](#)).

В операционных системах UNIX используется интерпретатор командной строки **/bin/sh**.

Пользовательские параметры подчиняются времени ожидания проверок на агента; если время ожидания будет превышено, созданный по пользовательским параметрам процесс будет завершен.

Смотрите также:

- [Пошаговую инструкцию](#) по использованию пользовательских параметров.
- [Выполнение команд](#)

Примеры простых пользовательских параметров

Простая команда:

UserParameter=ping, echo 1

Агент будет всегда возвращать '1' для элемента данных с ключем 'ping'.

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 параметры с указанным ключом. В этом случае гибкие пользовательские параметры могут быть основой для создания нескольких элементов данных.

Гибкие пользовательские параметры имеют следующий синтаксис:

UserParameter=ключ [*], команда

Параметр	Описание
Ключ	Уникальный ключ элемента данных. [*] задает, что ключ может принимать параметры из скобок. Параметры указываются при настройке элемента данных.
Команда	Команда, которая выполняется для получения значения ключа. Только для гибких пользовательских параметров: Вы можете использовать ссылки на позиции \$1...\$9 для того, чтобы обратиться к соответствующему параметру в ключе элемента данных. Zabbix разбирает параметры заключенные в [] ключа элемента данных и заменяет \$1,...,\$9 в команде соответственно. \$0 будет заменена оригинальной командой (до раскрытия \$0,...,\$9) для выполнения. Ссылки на позиции интерпретируются Zabbix агентом независимо от того заключены ли они в двойные ("") или в одинарные ('') кавычки.

Параметр

Описание

Для использования ссылок на позиции без изменения, укажите двойной символ доллара - например, awk '{print \$\$2}'. В этом случае \$\$2 фактически превратится в \$2 при выполнении команды.

Ссылки на позиции, начинающиеся со знака \$, ищутся и заменяются Zabbix агентом только в случае гибких пользовательских параметров. В случае простых пользовательских параметров такие ссылки пропускаются и, следовательно, экранирование любых знаков \$ не требуется.

По умолчанию некоторые символы не разрешены в пользовательских параметрах. Смотрите документацию по [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. В качестве параметров 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 подсчета 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.

Текстовый элемент данных (символ, журнал или текст типами информации) не станет неподдерживаемым в случае вывода ошибки стандартным выводом.

Пользовательские параметры, которые возвращают текст (символ, журнал, текстовый типы информации) могут возвращать пробел. В случае ошибочного результата элемент данных станет неподдерживаемым.

7.2.4.1 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 в zabbix_agentd.conf команду:

```
UserParameter=mysql.questions,mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

mysql.questions это уникальный идентификатор. Здесь может быть любой допустимый идентификатор, например, *queries*.

Протестируйте этот параметр, используя Zabbix агент с флагом “-t” (если запускаете под root, однако, стоит отметить, что агент может иметь разные права при работе демоном):

```
zabbix_agentd -t mysql.questions
```

Шаг 3

Перезапустите Zabbix агента.

Агент заново загрузит файл конфигурации.

Протестируйте этот параметр, используя утилиту [zabbix_get](#).

Шаг 4

Добавьте новый элемент данных с Ключем=mysql.questions к наблюдаемому узлу сети. Тип этого элемента данных должен быть Zabbix агент или Zabbix агент (активный).

Помните, очень важно указать корректный тип возвращаемого значения в Zabbix сервере. Иначе Zabbix не будет принимать значения.

7.2.5 Подгружаемые модули

7.2.5.1 Обзор

Подгружаемые модули предлагают производительную опцию для расширения функциональности Zabbix.

Уже имеются возможности для расширения функциональности Zabbix при помощи:

- [пользовательских параметров](#) (метрики агента)
- [внешние проверки](#) (мониторинг без агента)
- `system.run[]` [элемент данных Zabbix агента](#).

Они работают очень хорошо, но имеют главный недостаток, называемый форком (`fork()`). Zabbix должен создавать новый ответвленный процесс каждый раз для сбора пользовательских проверок, что не очень хорошо оказывается на производительности. Обычно это не самая большая проблема, но тем не менее это может быть серьезной проблемой для мониторинга встроенных систем, имеющих большое количество наблюдаемых параметров или тяжелых скриптов со сложной логикой или длительным временем запуска.

Поддержка подгружаемых модулей предлагает пути расширения Zabbix агента, сервера и прокси без ущерба производительности.

Подгружаемый модуль - в своей основе разделяемая библиотека используемая Zabbix демоном и загружаемая при старте демона. Библиотека должна содержать определенные функции такие, чтобы Zabbix процесс мог определить что файл на самом деле модуль и он может загрузить его и работать с ним.

Подгружаемые модули имеют много преимуществ. Отличная производительность и возможность внедрения в любую логику что очень важно, но возможно наиболее важное преимущество - возможность развития, использования и распространения Zabbix модулей. Это поспособствует беспроblemному обслуживанию и поможет вносить новую функциональность легче и независимо от кода ядра Zabbix.

Лицензирование и распространение модулей в бинарной форме регламентируется GPL лицензией (модули линкуются с Zabbix во время выполнения и используют заголовки Zabbix; в настоящее время весь код Zabbix лицензируется под GPL лицензией). Бинарная совместимость не гарантируется Zabbix'ом.

Постоянство API модулей гарантируется в пределах одного цикла Zabbix LTS (Долгосрочная поддержка) [выпуска](#). Постоянство Zabbix API не гарантируется (технически имеется возможность вызова внутренних функций Zabbix из модуля, но гарантии, что такие модули будут работать нет).

7.2.5.2 API модулей

Для того чтобы разделяемая библиотека обрабатывалась как Zabbix модуль, она должна реализовывать и экспортить несколько функций. На данный момент имеется шесть функций в API модулей Zabbix, только одна из которых обязательны, а остальные пять - optionalны.

2.1 Обязательный интерфейс

Единственная обязательная функция - `zbx_module_api_version()`:

```
int      zbx_module_api_version(void);
```

Эта функция должна возвращать API версию реализованную в модуле и, чтобы модуль загрузился, версия должна совпадать с версией API модулей поддерживаемой Zabbix. Версией API модулей поддерживаемой Zabbix является ZBX_MODULE_API_VERSION. Таким образом эта функция должна возвращать эту константу. Старая 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.

2.2 Опциональный интерфейс

Опциональными функциями являются следующие функции - `zbx_module_init()`, `zbx_module_item_list()`, `zbx_module_item_timeout()`, `zbx_module_history_write_cbs()` и `zbx_module_uninit()`:

```
int      zbx_module_init(void);
```

Эта функция должна выполнять необходимую инициализацию для модуля (если таковые имеются). 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 при помощи структуры 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 “время ожидания” задается 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 ZBX_HISTORY_WRITE_CBS структуры, 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 загружен, за исключением **zbx_module_uninit()**, которая вызывается один раз при завершении работы Zabbix, когда модуль выгружен.

2.3 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** - C функция, которая обрабатывает элемент данных (например, “zbx_module_dummy_random”), и **test_param** - список параметров, которые используется когда Zabbix агент запускается с флагом “-р” (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 два указателя параметра, 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 “dummy” модуля ниже для получения деталей как получать информацию от AGENT_REQUEST и как указывать информацию в AGENT_RESULT.

2.4 Предоставление обратного вызова экспорту истории

Экспорт истории через модули более не поддерживается Zabbix прокси начиная с Zabbix 4.0.0.

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” из элементов “history_num” 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 Zabbix сервером процессами синхронизации истории в конце процедуры синхронизации истории после того, как данные записаны в базу данных Zabbix и сохранены в кэш значений.

2.5 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 перечисленные выше структуры данных. Другой полезный заголовок - **include/sysinc.h**, который выполняет включение необходимых системных заголовков, который сам по себе помогает **include/module.h** работать должным образом.

Для того чтобы **include/module.h** и **include/sysinc.h** были включены, необходимо сначала выполнить команду **./configure** (без аргументов) из корня дерева исходных кодов Zabbix. Команда создаст **include/config.h** файл, который основывается на **include/sysinc.h**. (Если вы получили исходные коды Zabbix из хранилища Subversion, скрипт **./configure** не будет существовать, сначала выполните команду **./bootstrap.sh**, чтобы скрипт сгенерировался.)

Зная эту информацию, имейте в виду, что все готово для сборки модуля. Модуль должен включать **sysinc.h** и **module.h**, и скрипт сборки должен понимать, что эти два файла указаны в **include**.

Смотрите пример модуля “dummy” ниже для получения деталей.

Другим полезным заголовком является **include/log.h**, который определяет функцию **zabbix_log()**, которую можно использовать для журналирования и отладки.

7.2.5.3 Параметры конфигурации

Zabbix агент, сервер и прокси поддерживают два [параметра](#) для работы с модулями:

- **LoadModulePath** – полный путь до расположения подгружаемых модулей
- **LoadModule** – модуль(ы) для загрузки при старте. Модули должны находиться в папке указанной в параметре **LoadModulePath**. 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=dummy.so

```

После запуска агента будут загружены модули mariadb.so, apache.so, kernel.so и dummy.so из папки /usr/local/lib/zabbix/agent. Агент не запустится, если модуль отсутствует, либо в случае некорректных прав доступа или разделяемая библиотека не является модулем Zabbix.

7.2.5.4 Настройка веб-интерфейса

Подгружаемые модули поддерживаются Zabbix агентом, сервером и прокси. Следовательно, тип элемента данных в Zabbix веб-интерфейсе 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-NUL функцию обратной связи, тогда экспорт истории будет включен автоматически.

7.2.5.5 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, он 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) 2001-2016 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 "sysinc.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 internal */
/* 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 ********************/
/*
 * 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 ********************/
/*
 * 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 ********************/
ZBX_METRIC      *zbx_module_item_list(void)
{
    return keys;
}

static int      dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    SET_UI64_RESULT(result, 1);

    return SYSINFO_RET_OK;
}

static int      dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char      *param;

    if (1 != request->nparam)
    {
        /* set optional error message */

```

```

        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
        return SYSINFO_RET_FAIL;
    }

    param = get_rparam(request, 0);

    SET_STR_RESULT(result, strdup(param));

    return SYSINFO_RET_OK;
}

/********************* Function: dummy_random ********************/
/*
 * 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->timeout - processing should not take longer than
 *                                this number of seconds
 *              request->params[N-1] - pointers to item key parameters
 *
 *              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.
 */
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 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 экспортирует 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]`

7.2.5.6 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 отношения к модулю из `zabbix_agentd.conf`. В настоящее время этот функционал не поддерживается. Если вам необходимо в вашем модуле использовать некоторые параметры конфигурации вам, возможно, следует осуществлять анализ отдельного файла конфигурации модуля.

7.2.6 Счетчики производительности Windows

Обзор

Вы можете эффективно мониторить счетчики 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, смотрите [специфичные ключи элементов данных для Windows](#).

Чтобы получить полный список счетчиков производительности для мониторинга, вы можете выполнить:

```
tuperf -qx
```

Числовое представление

В зависимости от настроек локализации, именования счетчиков производительности могут быть разными на разных серверах Windows. Такое поведение может внести определенные проблемы при создании шаблона для мониторинга нескольких Windows машин, использующих разные настройки локализации.

В то же время каждый счетчик производительности может быть переведен в цифровую форму, которая является уникальной и независимой от языковых настроек, так что вы можете использовать числовое представление, а не строковое.

Для того чтобы найти цифровые эквиваленты, выполните `regedit`, а затем найдите `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

Запись в реестре содержит информацию наподобии 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 счетчика производительности, такой как '\System\% Processor Time':

```
System -> 2  
% Processor Time -> 6
```

Затем вы можете использовать эти числа для преобразования пути в числа:

\2\6

Пользовательские параметры

Вы можете разворачивать некоторые PerfCounter параметры для мониторинга счетчиков производительности Windows.

Например, вы можете добавить следующее в файл конфигурации Zabbix агента:

```
PerfCounter=UserPerfCounter1, "\Memory\Page Reads/sec", 30  
или  
PerfCounter=UserPerfCounter2, "\4\24", 30
```

С такими параметрами, вы можете просто использовать *UserPerfCounter1* или *UserPerfCounter2* как ключи при создании соответствующих элементов данных.

Не забудьте перезапустить Zabbix агента после внесения изменений в файл конфигурации.

Решение проблем

Периодически Zabbix агент не может получить значения счётчиков производительности в системах на основе Windows 2000, потому что pdh.dll файл устарел. Такое поведение отображается сообщениями об ошибках в файлах журналов Zabbix агента и сервера. В этом случае необходимо обновить pdh.dll на более новую 5.0.2195.2668 версию.

7.2.7 Массовое обновление

Обзор

Иногда вы можете захотеть изменить какие-нибудь атрибуты у нескольких элементов данных одновременно. Вместо открытия каждого элемента данных для редактирования, вы можете воспользоваться функцией массового обновления.

Использование массового обновления

Для массового обновления каких-либо элементов данных, выполните следующее:

- Отметьте элементы данных из списка для обновления
- Выберите *Массовое обновление* из списка ниже
- Отметьте атрибуты для обновления
- Введите новые значения для этих атрибутов и нажмите на *Обновить*

7.2.8 Преобразование значений

Обзор

Для более “человеческого” представления полученных значений, вы можете использовать преобразования значений, которые содержат соответствия между числовыми значениями и строковыми представлениями.

Преобразования значений можно использовать как в веб-интерфейсе Zabbix, так и в оповещениях отправляемых по email/SMS/jabber и т.п.

Например, элемент данных, который имеет значения '0' или '1' может использовать преобразование значений для представления этих значений в более читаемой форме:

- '0' ⇒ 'Недоступен'
- '1' ⇒ 'Доступен'

Или преобразование значений для резервного копирования данных может быть такое:

- 'F' ⇒ 'Полный'
- 'D' ⇒ 'Дифференциальный'
- 'T' ⇒ 'Инкрементный'

Таким образом, при [настройке элементов данных](#) вы можете указать преобразование значений для более “человеческого” отображения этого элемента данных. Чтобы сделать это, вы должны выбрать имя ранее добавленного преобразования значения в поле *Отображение значения*.

Преобразование значений можно использовать с элементами данных, которые имеют Числовой (с плавающей точкой) и Символ типы информации.

Преобразования значений, начиная с Zabbix 3.0, можно экспортировать/импортировать, как отдельно, так и с соответствующими шаблоном или узлом сети.

Настройка

Для того чтобы добавить преобразование значений:

- Перейдите в: Администрирование → Общие
- Выберите Преобразование значений из выпадающего меню
- Нажмите на Создать преобразование значений (или на имя уже существующего преобразования)

* Name	Windows service state	
* Mappings	Value	Mapped to
	0	⇒ Running
	1	⇒ Paused
	2	⇒ Start pending
	3	⇒ Pause pending
	4	⇒ Continue pending
	5	⇒ Stop pending
	6	⇒ Stopped
	7	⇒ Unknown
	255	⇒ No such service
	Add	
	Add	Cancel

Параметры преобразования значений:

Параметр

Описание

Имя Уникальное имя набора преобразования значений.

Соответствия Индивидуальные преобразования - пары числовых (буквенных) значений и их строковых представлений.

Все обязательные поля ввода отмечены красной звёздочкой.

Для добавления отдельного соответствия, нажмите на *Добавить*.

Как это работает

Например, один из предустановленных агентских элементов данных 'Ping to the server (TCP)' использует существующее преобразование значение именуемое 'Service state' для отображения своих значений.

* Name	Service state	
* Mappings	Value	Mapped to
	0	⇒ Down
	1	⇒ Up
	Add	
	Add	Cancel

В [диалоге настройки](#) элемента данных вы можете увидеть ссылку на это преобразование значений в поле *Отображение значения*:



Таким образом в *Мониторинг → Последние данные* преобразование подставляется для отображения 'Up' (вместе с исходным значением в скобках).

NAME	LAST CHECK	LAST VALUE
Zabbix agent (1 item)		
Agent ping	2015-08-11 22:01:07	Up (1)

В разделе *Последние данные* отображаемые значения сокращаются до 20 символов. Если используется преобразование значений, то усечение не применяется к значению соответствия, оно применяется только к сырому значению отдельно (отображается в круглых скобках).

Значение, которое отображается в более читаемой форме, также проще для понимания при получении оповещений.

Без предустановленного преобразования значений вы могли бы получить только следующее:

NAME	LAST CHECK	LAST VALUE
Zabbix agent (1 item)		
Agent ping	2015-08-11 22:09:21	1

Поэтому, в этом случае придется либо догадываться, что означает '1' или искать в документации, чтобы узнать значение этой цифры.

Type	<input type="checkbox"/>	Original
Host interface	<input type="checkbox"/>	Original
JMX endpoint	<input type="checkbox"/>	Original
URL	<input type="checkbox"/>	Original
Request body type	<input type="checkbox"/>	Original
Request body	<input type="checkbox"/>	Original
Headers	<input type="checkbox"/>	Original
SNMP community	<input type="checkbox"/>	Original
Context name	<input type="checkbox"/>	Original
Security name	<input type="checkbox"/>	Original
Security level	<input type="checkbox"/>	Original
Authentication protocol	<input type="checkbox"/>	Original
Authentication passphrase	<input type="checkbox"/>	Original
Privacy protocol	<input type="checkbox"/>	Original
Privacy passphrase	<input type="checkbox"/>	Original
Port	<input type="checkbox"/>	Original
Type of information	<input type="checkbox"/>	Original
Units	<input type="checkbox"/>	Original
Authentication method	<input type="checkbox"/>	Original
User name	<input type="checkbox"/>	Original
Public key file	<input type="checkbox"/>	Original
Private key file	<input type="checkbox"/>	Original
Password	<input type="checkbox"/>	Original
Preprocessing steps	<input type="checkbox"/>	Original
Update interval	<input type="checkbox"/>	Original
History storage period	<input checked="" type="checkbox"/>	<input type="text" value="7d"/>
Trend storage period	<input type="checkbox"/>	Original
Status	<input type="checkbox"/>	Original
Log time format	<input type="checkbox"/>	Original
Show value	<input type="checkbox"/>	Original
Enable trapping	<input type="checkbox"/>	Original
Allowed hosts	<input type="checkbox"/>	Original
Replace applications	<input type="checkbox"/>	Original
Add new or existing applications	<input type="checkbox"/>	Original
Master item	<input type="checkbox"/>	Original
Description	<input type="checkbox"/>	Original

Заменить группы элементов данных удалит элемент данных из существующих групп элементов данных и заменит на те, что перечислены в этом поле.

Добавить новую или существующую группу элементов данных позволяет указать дополнительные группы элементов данных из существующих или создать совершенно новые группы элементов данных для элементов данных.

Оба эти поля имеют функцию автодополнения - начните вводить и в них автоматически появится выпадающий список совпадающих групп элементов данных. Если группа элементов данных новая, то она также появится в выпадающем списке и будет обозначена (*новый*) в конце строки. Просто прокрутите список и выберете нужное.

7.2.9 Группы элементов данных

Обзор

Группы элементов данных используются для группировки элементов данных в логические группы.

Например, группа элементов данных *MySQL Server* может содержать все элементы данных связанные с MySQL сервером: доступность MySQL, дисковое пространство, загрузка процессора, транзакции в секунду, количество медленных запросов и прочее.

Группы элементов данных также используются для группировки веб-сценариев.

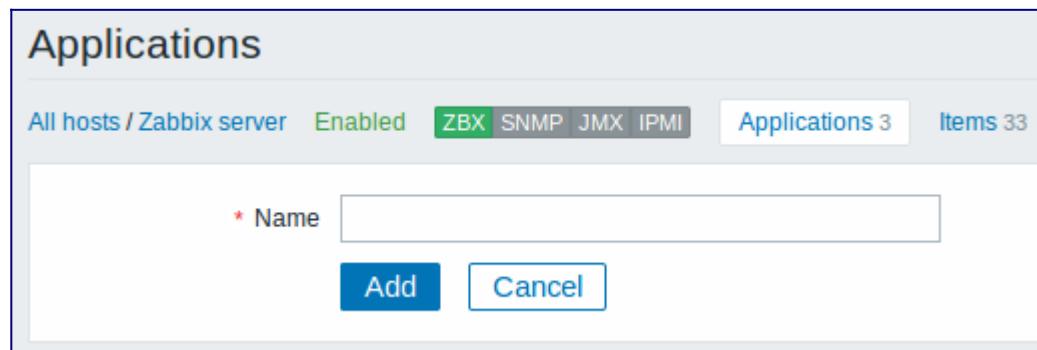
Если вы используете группы элементов данных, то в *Мониторинг → Последние данные* вы увидите элементы данных и веб-сценарии сгруппированные в соответствии со своими группами элементов данных.

Настройка

Для работы с группами элементов данных вы должны сначала создать их и затем присоединить к ним элементы данных или веб-сценарии.

Для создания группы элементов данных, сделайте следующее:

- Перейдите в *Настройка → Узлы сети или Шаблоны*
- Далее нажмите на *Группы элементов данных* у требуемого узла сети или шаблона
- Нажмите на *Создать группу элементов данных*
- Ведите имя группы элементов данных и нажмите на *Добавить*, чтобы сохраните её



Вы можете также создать новую группу элементов данных напрямую из диалога свойств элемента данных.

Элементы данных соединяются с группами элементов данных в диалоге свойств элементов данных. Выберите одну или несколько групп элементов данных, которым элемент данных будет принадлежать.

Веб-сценарии соединяются с группами элементов данных в диалоге добавления веб-сценария. Выберите группу элементов данных, которой сценарий будет принадлежать.

7.2.10 Очередь

Обзор

Очередь отображает элементы данных, которые ожидают 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

Статистика, представленная в очереди, является хорошим показателем производительности сервера Zabbix.

Очередь запрашивается напрямую с Zabbix сервера с использованием JSON протокола. Информация доступна только, если Zabbix сервер запущен.

Чтение очереди

Для чтения очереди, перейдите в *Администрирование* → *Очередь*. В правом верхнем углу в выпадающем списке должен быть выбран *Обзор*.

ITEMS	5 SECONDS	10 SECONDS	30 SECONDS	1 MINUTE	5 MINUTES	MORE THAN 10 MINUTES
Zabbix agent	0	0	0	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	1	0	5	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
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

В целом это изображение “зеленое”. Поэтому мы можем считать, что сервер работает хорошо.

Очередь показывает один элемент данных ожидающий около 10 секунд и один ожидающий около 5 минут. Приятно, было бы здорово знать, что это за элементы данных.

Чтобы сделать это, выберите *Детали* в верхнем правом углу из выпадающего меню. Теперь вы можете видеть список этих задерживающихся элементов данных.

SCHEDULED CHECK	DELAYED BY	HOST	NAME
2015-08-11 22:43:51	12s	Remote proxy: Zabbix server	Zabbix history write cache, % free
2015-08-11 22:43:52	11s	Remote proxy: Zabbix server	Zabbix text write cache, % free
2015-08-11 22:43:53	10s	Remote proxy: Zabbix server	Zabbix trend write cache, % free
2015-08-11 22:43:54	9s	Remote proxy: Zabbix server	Values processed by Zabbix server per second

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 быть обновлены с секунды на секунду. Однако, если вы видите большое количество элементов данных ожидающих обновления длительное время, это может стать серьезной проблемой.

ITEMS	5 SECONDS	10 SECONDS	30 SECONDS	1 MINUTE	5 MINUTES	MORE THAN 10 MINUTES
Zabbix agent	0	13	7	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	5	1	9	0	0	0

Агент недоступен?

Элемент данных очереди

Можно использовать специальный внутренний элемент данных **zabbix[очередь, <от>, <до>]** для наблюдения за состоянием очереди Zabbix. Он будет возвращать количество элементов данных задерживающихся за указанный период времени. Для более подробной информации смотрите [Внутренние элементы данных](#).

7.2.11 Кэш значений

Обзор

Чтобы сделать вычисления выражений триггеров, вычисляемых/агрегированных элементов данных и некоторых макросов, начиная с Zabbix 2.2 в Zabbix сервере поддерживается опция кэша значений.

Этот кэш находящийся в оперативной памяти используется для доступа к данным истории, вместо выполнения прямых запросов SQL в базу данных. Если данные истории не присутствуют в кэше, недостающие значения запрашиваются из базы данных и, соответственно, кэш обновляется.

Для включения функционала кэша значений, поддерживается дополнительный параметр **ValueCacheSize** в [файле конфигурации](#) Zabbix server.

Два внутренних элемента данных поддерживаются для наблюдения за кэшем значений **zabbix[vcache,buffer,<режим>]** и **zabbix[vcache,cache,<параметр>]**. Для получения более детальных сведений смотрите [внутренние элементы данных](#).

7.2.12 Проверить сейчас

Обзор

Проверка нового значения элемента данных в Zabbix циклический процесс, который основывается на заданных интервалах обновления. Хотя у многих элементов данных интервалы обновления достаточно маленькие, имеются и другие (включая правила низкоуровневого обнаружения) у которых интервалы обновления весьма большие, таким образом в реальных ситуациях в жизни может потребоваться более быстрая проверка нового значения - чтобы подхватить изменения в обнаруживаемых ресурсах, например. Чтобы покрыть подобную необходимость, имеется возможность изменения расписания пассивной проверки и получения нового значения немедленно.

Эта функция поддерживается только для **пассивных** проверок. Поддерживаются следующие типы элементов данных:

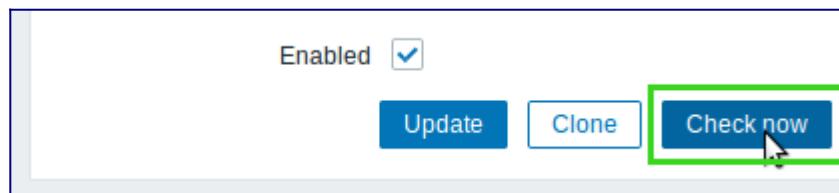
- Zabbix агент (пассивный)
- SNMPv1/v2/v3 агент
- IPMI агент
- Простая проверка
- Zabbix внутренний
- Zabbix агрегированный
- Внешняя проверка
- Монитор баз данных
- JMX агент
- SSH агент
- Telnet
- Вычисляемый
- HTTP агент

Проверка должна существовать в кэше конфигурации, чтобы выполнение было успешным; для получения более подробных сведений смотрите [CacheUpdateFrequency](#). При проверке нового значения кэш конфигурации **не** обновляется, поэтому значения не будут отражать совсем недавние изменения конфигураций элементов данных/правил обнаружения. Таким образом, также отсутствует возможность проверки нового значения по элементу данных/правилу обнаружения, которые созданы только что.

Настройка

Для выполнения немедленной пассивной проверки:

- нажмите на *Проверить сейчас* в диалоге настройки существующего элемента данных (или правила обнаружения):



- нажмите на *Проверить сейчас* для выбранных элементов данных/правил в списке элементов данных/правил обнаружения:

В последнем случае можно выбрать несколько элементов данных/правил и выполнить “проверку сейчас” сразу по всем выбранным объектам.

7.3 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 приведет 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 “поджигается” - или другими словами меняет состояние на ПРОБЛЕМА.

Триггер может принимать следующие состояния:

ЗНАЧЕНИЕ	ОПИСАНИЕ
OK	Нормальное состояние триггера. В более ранних версиях Zabbix оно называлось ЛОЖЬ.
ПРОБЛЕМА	Обычно означает, что что-то случилось. Например, загрузка процессора слишком высокая. В более ранних версиях Zabbix оно называлось ИСТИНА.
Состояние триггера (выражение)	пересчитывается каждый раз, когда Zabbix сервер получает новое значение, которое является частью выражения.

Если в выражении триггера используются функции, связанные со временем (**nodata()**, **date()**, **dayofmonth()**, **dayofweek()**, **time()**, **now()**), то тогда триггер будет пересчитываться каждые 30 секунд процессом Zabbix синхронизации истории. Если используются оба типа функций связанные со временем и не связанные со временем в одном выражении, то пересчёт выполняется при поступлении нового значения и же каждые 30 секунд.

Вы можете [создавать выражения триггеров](#) с различной степенью сложности.

7.3.1 Настройка триггера

Обзор

Для настройки триггера, сделайте следующее:

- Перейдите в: *Настройка* → *Узлы сети*
- Нажмите на *Триггеры* в строке с узлом сети

- Нажмите на *Создать триггер* сверху справа (или на имя триггера для изменения уже существующего триггера)
- Ведите в диалоге параметры триггера

Настройка

Вкладка **Триггер** содержит все существенные атрибуты триггера.

Trigger Dependencies

* Name: Disk I/O is overloaded on {HOST.NAME}

Severity: Not classified | Information | **Warning** | Average | High

* Expression: `{Zabbix server:system.cpu.util[,iowait].avg(5m)}>20 and {Zabbix server:system.uname.str(Linux)}=1`

OK event generation: Expression | Recovery expression | None

PROBLEM event generation mode: Single | Multiple

OK event closes: All problems | All problems if tag values match

Tags: tag | value
Add

Allow manual close:

URL: [empty]

Description: OS spends significant time waiting for I/O (input/output) operations. It could be indicator of performance issues with storage system.

Enabled:

Add | Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	<p>Имя триггера. Имя может содержать поддерживаемые макросы: {HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {ITEM.VALUE}, {ITEM.LASTVALUE} и {\$MACRO}.</p> <p>Можно использовать макросы \$1, \$2...\$9 как ссылки на первую, вторую... девятую константу из выражения.</p> <p><i>Обратите внимание:</i> Макросы \$1-\$9 будут раскрыты корректно, если</p>

Параметр	Описание
<i>Важность</i>	константы на которые ссылаются макросы, в относительно простых, понятных выражениях. Например, имя “Загрузка процессора превышает \$1 на {HOST.NAME}” будет автоматически изменено на “Загрузка процессора превышает 5 на New host”, если выражение - {New host:system.cpu.load[percru,avg1].last()}>5
<i>Выражение проблемы</i>	Определение требуемой <u>важности</u> триггера нажатием кнопок.
<i>Формирование OK событий</i>	<p>Логическое <u>выражение</u>, используемое для определения условий проблемы.</p> <p>Опции формирования ОК событий:</p> <p>Выражение - ОК события формируются на основе того же выражения что и события о проблемах;</p> <p>Выражение восстановления - ОК события формируются, если выражение проблемы вычисляется как ЛОЖЬ и выражение восстановления вычисляется как ПРАВДА;</p> <p>Нет - в этом случае триггер никогда самостоятельно не вернётся в состояние ОК.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p> <p>Логическое <u>выражение</u>, используемое для определения условий, когда проблема решена.</p> <p>Выражение восстановления вычисляется только после того, как выражение проблемы будет вычислено как ЛОЖЬ. Невозможно решить проблему с помощью выражения восстановления, если условие проблемы всё ещё присутствует.</p> <p>Это поле опционально и доступно только, если в <i>Формирование OK событий</i> выбрано 'Выражение восстановления'.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p> <p>Режим формирования событий о проблемах:</p> <p>Одиночный - формируется одно событие, когда триггер переходит в состояние 'Проблема' в первый раз;</p> <p>Множественный - событие формируется на каждое 'Проблема' вычисление триггера.</p> <p>Выберите, если ОК событие закрывает:</p> <p>Все проблемы - все проблемы этого триггера</p> <p>Все проблемы, если значение тега совпадает - только те проблемы триггера, значения тегов событий которых совпадают</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p> <p>Укажите имя тега события, которое будет использоваться для корреляции событий.</p> <p>Это поле отображается, если в свойстве <i>OK событие закрывает</i> выбрано 'Все проблемы, если значения тегов совпадают' и в этом случае это поле обязательно.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p> <p>Укажите пользовательские теги для маркировки событий триггера.</p> <p><u>Теги событий</u> могут использоваться для корреляции событий, в условиях действий и будут также видны в <i>Мониторинг</i> → <i>Проблемы</i> и в <i>Проблемы</i> виджете.</p> <p>Тегами является пара имя тега и значение. Вы можете использовать только имя или пару имя со значением.</p> <p>Пользовательские макросы, контекст пользовательских макросов, макросы низкоуровневого обнаружения и макросы <u>функций</u> {{ITEM.VALUE}}, {{ITEM.LASTVALUE}}, а также макросы низкоуровневого обнаружения, поддерживаются в тегах событий. Макросы низкоуровневого обнаружения можно использовать в контексте макросов.</p>
<i>Выражение восстановления</i>	
<i>Режим формирования ПРОБЛЕМА событий</i>	
<i>OK событие закрывает</i>	
<i>Теги для совпадения</i>	
<i>Теги</i>	

Параметр	Описание
<i>Разрешить закрывать вручную</i>	Если суммарная длина раскрытоого значения превышает, оно будет обрезано до 255 символов. Смотрите все поддерживаемые макросы тегами событий. Поддерживается начиная с Zabbix 3.2.0.
<i>URL</i>	Выберите, чтобы разрешить закрывать вручную события о проблемах сформированные этим триггером. Закрытие вручную возможно при подтверждении событий о проблемах. Поддерживаются начиная с Zabbix 3.2.0.
<i>Описание</i>	Если не пустой, то URL, указанный здесь, доступен в виде ссылки в нескольких местах веб-интерфейса, например при нажатии на имя проблемы в <i>Мониторинг → Проблемы</i> (опция <i>URL</i> в меню <i>Триггер</i>) и в виджете <i>Проблемы</i> на панели. Поддерживаемые макросы: {ITEM.VALUE}, {ITEM.LASTVALUE}, {TRIGGER.ID}, несколько {HOST.*} макросов, пользовательские макросы. Текстовое поле используется, для того чтобы сообщить больше информации об этом триггере. Может содержать инструкции по устранению специфичной проблемы, контактные данные ответственного сотрудника и т.п. <i>Начиная с Zabbix 2.2</i> , описание может содержать тот же набор макросов, что и имя триггера.
<i>Активировано</i>	Уберите отметку деактивирует триггер, если это требуется. Вкладка Зависимости содержит все зависимости триггера.
Нажмите на <i>Добавить</i> для добавления новой зависимости.	
Вы также можете настроить триггер, открыв уже существующий триггер, затем нажать на кнопку <i>Клонировать</i> и сохранить его под другим именем.	

Тестирование выражений

Имеется возможность проверки настроенного выражения триггера относительного того, как результат выражения будет меняться в зависимости от полученного значения.

Следующее выражение с официального шаблона взято в качестве примера:

```
{Template Net Cisco IOS SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.
{#SNMPINDEX}].avg(5m)}>{$TEMP_WARN}
or
{Template Net Cisco IOS SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.
{#SNMPINDEX}].last(0)}={$TEMP_WARN_STATUS}
```

Для тестирования выражения, нажмите на *Конструктор выражения* под полем выражения.

The screenshot shows the 'Trigger' configuration page. The 'Dependencies' tab is selected. In the 'Expression' field, the following code is entered:

```
{Template Net Cisco IOS
SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.
{#SNMPINDEX}].avg(5m)}>{$TEMP_WARN}
or
{Template Net Cisco IOS
SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.
{#SNMPINDEX}].last(0)}={$TEMP_WARN_STATUS}
```

A green box highlights the 'Expression constructor' button at the bottom of the expression input area.

В Конструкторе выражения перечислены все отдельные выражения. Чтобы открыть окно тестирования, нажмите на *Тест* ниже списка выражений.

Target Expression

Or

⌞ A {Template Net Cisco IOS SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.#{SNMPINDEX}].avg(5m)]>{\$TEMP_WARN}

⌞ B {Template Net Cisco IOS SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.#{SNMPINDEX}].last(0)}=={\$TEMP_WARN_STATUS}

Test

В окне тестирования вы можете задать пробные значения (“80, 70, 0, 1” в этом примере) и затем увидеть результат выражения, нажав на кнопку *Тест*.

Test

Test data

Expression	Type	Value
{Template Net Cisco IOS SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.#{SNMPINDEX}].avg(5m)}	Numeric (float)	80
(\$TEMP_WARN)	Numeric (float)	70
{Template Net Cisco IOS SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.#{SNMPINDEX}].last(0)}	Numeric (integer)	0
(\$TEMP_WARN_STATUS)	Numeric (float)	1

Result

Expression	Result
Or	TRUE
⌞ A {\$TEMP_WARN}	TRUE
⌞ B {\$TEMP_WARN_STATUS}	FALSE
A or B	TRUE

Test **Cancel**

Можно увидеть результат отдельных выражений, наряду со полным выражением.

Результат “ПРАВДА” означает, что введённое выражение корректно. В этом конкретном выражении А, “80” больше чем указанное значение {\$TEMP_WARN}, “70” в этом примере. Соответственно, появится результат “ПРАВДА”.

Результат “ЛОЖЬ” означает, что введённое выражение некорректно. В этом конкретном выражении В, {\$TEMP_WARN_STATUS}, равно “1” в этом примере, должно быть равно указанному значению 0, и это неправда. Соответственно, появится результат “ЛОЖЬ”.

Выбранным типом выражения является “ИЛИ”/“ПРАВДА”. Если по крайней мере одно из заданных выражений (А или В в этом случае) будет ПРАВДА, общий результат будет также ПРАВДА. Что означает, что текущее значение превысило пороговое значение и возникнет Проблема.

7.3.2 Выражение триггера

Обзор

Используемые в триггерах выражения являются очень гибкими. Вы можете использовать их для создания сложных логических тестов, учитывая статистику по мониторингу.

Простое полезное выражение может выглядеть примерно так:

{<сервер>:<ключ>.<функция>(<параметр>)}<оператор><константа>

Функции

Функции триггеров позволяют ссылаться на собранные значения, текущее время и другие факторы.

Имеется полный список [поддерживаемых функций](#).

Параметры функций

Большинство числовых функций принимают количество секунд в качестве параметра.

Вы можете использовать префикс #, чтобы указать что этот параметр должен иметь другой смысл:

ВЫЗОВ ФУНКЦИИ	СМЫСЛ
sum(600)	Сумма всех значений за 600 секунд
sum(#5)	Сумма последних 5 значений

Функция **last** использует другой смысл для значений, когда начинается с решетки - она дает выбрать n-ое предыдущее значение, так что с учетом значений 3, 7, 2, 6, 5 (от наиболее нового до наиболее старого), при **last(#2)** вернется 7 и при **last(#5)** вернется 5.

Несколько функций поддерживают дополнительный, второй параметр **сдвиг_времени**. Этот параметр позволяют ссылаться на данные из периода времени в прошлом. Например, для **avg(1h,1d)** будет возвращено среднее значение за час днем ранее.

Вы можете использовать поддерживаемые [суффиксы преобразований](#) в выражениях триггеров, например, '5m' (минут) вместо '300' секунд или '1d' (день) вместо '86400' секунд. '1K' будет состоять из '1024' байт.

Операторы

Следующие операторы поддерживаются для триггеров (**представлены по убыванию приоритета выполнения**):

ПРИОРИТЕТ	ОПЕРАТОР	ОПРЕДЕЛЕНИЕ	Заметки по неизвестным значениям
1	-	Унарный минус	-Неизвестно → Неизвестно
2	not	Логическое НЕ	not Неизвестно → Неизвестно
3	*	Умножение	0 * Неизвестно → Неизвестно (да, Неизвестно, не 0 - чтобы не потерять Неизвестно в арифметических операциях)
	/	Деление	1.2 * Неизвестно → Неизвестно Неизвестно / 0 → ошибка Неизвестно / 1.2 → Неизвестно 0.0 / Неизвестно → Неизвестно
4	+	Арифметический плюс	1.2 + Неизвестно → Неизвестно
	-	Арифметический минус	1.2 - Неизвестно → Неизвестно
5	<	Менее чем. Этот оператор может быть представлен в виде: $A < B \Leftrightarrow (A < B - 0.000001)$	1.2 < Неизвестно → Неизвестно
	<=	Менее чем или равно. Этот оператор может быть представлен в виде:	Неизвестно <= Неизвестно → Неизвестно

$A \leq B \Leftrightarrow (A \leq B + 0.000001)$

Более чем. Этот оператор может быть представлен в виде:

>

$A > B \Leftrightarrow (A > B + 0.000001)$

Более чем или равно. Этот оператор может быть представлен в виде:

>=

$A \geq B \Leftrightarrow (A \geq B - 0.000001)$

Равенство. Этот оператор может быть представлен в виде:

6

=

$A = B \Leftrightarrow (A \geq B - 0.000001) \text{ и } (A \leq B + 0.000001)$

Не равно. Этот оператор может быть представлен в виде:

<>

$A \neq B \Leftrightarrow (A < B - 0.000001) \text{ или } (A > B + 0.000001)$

0 **and** Неизвестно → 0

1 **and** Неизвестно →

Неизвестно

Неизвестно **and** Неизвестно → Неизвестно

1 **or** Неизвестно → 1

0 **or** Неизвестно →

Неизвестно

Неизвестно **or** Неизвестно → Неизвестно

7

and

Логическое И

8

or

Логическое ИЛИ

Операторы **not**, **and** и **or** регистрозависимы 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 на право. Унарные - и **not** не ассоциативны (имеется в виду необходимо использовать **-(-1)** и **not (not 1)** вместо **--1** и **not not 1**).

Результат вычисления:

- Операторы <, <=, >, >=, <> должны давать '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 operandам должно давать '1', если оба из этих operandов сравнения не равны '0'; в противном случае, будет давать '0'; для неизвестных operandов **and** даст '0' только, если один из operandов сравнения равен '0'; в противном случае, он даст 'Неизвестно';
- **or** по известным operandам должно давать '1', если какой-либо из этих operandов сравнения не равен '0'; в противном случае, будет давать '0'; для неизвестных operandов **or** даст '1' только, если один из operandов сравнения не равен '0'; в противном случае, он даст 'Неизвестно';
- Результат логического operandда отрицания **not** для известного operandда равен '0', если значение этого operandда сравнения не равно '0'; '1', если значение его operandда сравнения равно '0'. Для неизвестных operandдов **not** даст 'Неизвестно'.

Кэширование значений

Значения, которые требуются для вычисления триггеров, кэшируются Zabbix сервером. По этой причине такое вычисление триггеров на некоторое время приводит к более высокой загрузке базы данных 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, которые заданы в функциях триггеров, либо пока сервер не будет перезапущен.

Примеры триггеров

Пример 1

Высокая загрузка процессора на www.zabbix.com.

```
{www.zabbix.com:system.cpu.load[all,avg1].last()}>5
```

'www.zabbix.com:system.cpu.load[all,avg1]' представляет короткое имя наблюдаемого параметра. Эта строка указывает, что сервер - 'www.zabbix.com' и наблюдаемый ключ - 'system.cpu.load[all,avg1]'. Используя функцию 'last()', мысылаемся на самое последнее значение. И наконец '>5' означает, что триггер перейдет в состояние ПРОБЛЕМА всякий раз, когда самое новое измерение загрузки процессора на сервере www.zabbix.com будет превышать 5.

Пример 2

www.zabbix.com перегружен

```
{www.zabbix.com:system.cpu.load[all,avg1].last()}>5 or  
{www.zabbix.com:system.cpu.load[all,avg1].min(10m)}>2
```

Это выражение будет истинным, когда либо текущая загрузка процессора станет более 5, либо загрузка процессора больше значения 2 за последние 10 минут.

Пример 3

/etc/passwd был изменен

Используется функция diff:

```
{www.zabbix.com:vfs.file.cksum[/etc/passwd].diff()}=1
```

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 скачивает большой файл из Интернет

Используется функция min:

```
{www.zabbix.com:net.if.in[eth0,bytes].min(5m)}>100K
```

Это выражение будет истинным, когда количество полученных байт на eth0 превышает 100 КБ за последних 5 минут.

Пример 5

Оба узла кластера SMTP серверов недоступны

Примечание, в выражении используются два разных узла сети:

```
{smtp1.zabbix.com:net.tcp.service[smtp].last()}=0 and  
{smtp2.zabbix.com:net.tcp.service[smtp].last()}=0
```

Это выражение будет истинным, когда оба SMTP сервера недоступны на обоих smtp1.zabbix.com и smtp2.zabbix.com.

Пример 6

Zabbix агент нуждается в обновлении

Используется функция str():

```
{zabbix.zabbix.com:agent.version.str("beta8")}=1
```

Это выражение будет истинным, когда версия Zabbix агента содержит в себе 'beta8' (возможно 1.0beta8).

Пример 7

Сервер недоступен

```
{zabbix.zabbix.com:icmpping.count(30m,0)}>5
```

Это выражение будет истинным, если узел сети “zabbix.zabbix.com“ недоступен более 5 раз за последние 30 минут.

Пример 8

Нет данных за последние 3 минуты

Используется функцию nodata():

```
{zabbix.zabbix.com:tick.nodata(3m)}=1
```

Для того, чтобы этот триггер заработал, элемент данных ‘tick’ должен быть задан как элемент данных типа Zabbix [траппер](#). Узел сети должен периодически отправлять данные этому элементу данных, используя zabbix_sender. Если не было получено данных за последние 180 секунд, значением триггера станет ПРОБЛЕМА.

Обратите внимание, что 'nodata' можно использовать с любым типом элементов данных.

Пример 9

Активность CPU в ночное время

Используется функция time():

```
{zabbix:system.cpu.load[all,avg1].min(5m)}>2 and  
{zabbix:system.cpu.load[all,avg1].time()}>000000 and  
{zabbix:system.cpu.load[all,avg1].time()}<060000
```

Триггер может изменить свое состояние в истинное только в ночное время (00:00-06:00).

Пример 10

Проверка синхронизации времени на клиенте со временем на Zabbix сервере

Используется функция fuzzytime():

```
{MySQL_DB:system.localtime.fuzzytime(10)}=0
```

Триггер изменит состояние на проблему тогда, когда локальное время на сервере MySQL_DB и Zabbix сервере различаются более чем на 10 секунд.

Пример 11

Сравнение средней загрузки сегодня со средним значением загрузки за это же время вчера (использование второго параметра **Сдвиг_времени**).

```
{server:system.cpu.load.avg(1h)}/{server:system.cpu.load.avg(1h,1d)}>2
```

Триггер изменит свое состояние на проблему, если средняя загрузка за последний час будет в два раза больше чем за аналогичный период времени вчера.

Пример 12

Использование значение другого элемента данных в качестве порогового значения триггера:

```
{Template PfSense:hrStorageFree[#{SNMPVALUE}].last()}<{Template PfSense:hrStorageSize[#{SNMPVALUE}].last()}*0.1
```

Триггер изменит свое состояние на проблему, если свободное пространство на диски упадет ниже 10 процентов.

Пример 13

Использование [результата вычисления](#) для получения количества триггеров больше порога:

```
({server1:system.cpu.load[all,avg1].last()}>5) +  
({server2:system.cpu.load[all,avg1].last()}>5) +  
({server3:system.cpu.load[all,avg1].last()}>5)>=2
```

Триггер изменит свое состояние на проблему, если по крайней мере два триггера из выражения будут больше 5.

Гистерезис

Порой нам необходим интервал между состояниями OK и Проблема, а не просто порог. Например, мы бы хотели задать триггер, который переходит в состояние Проблема, когда температура в серверной комнате становится больше 20C и мы бы хотели чтобы он оставался в этом состоянии пока температура не опустится ниже 15C.

Чтобы это сделать, сначала мы зададим выражение триггера для события о проблеме. Затем выберем 'Выражение восстановления' в *Формирование OK события* и укажем выражение восстановления для OK события.

Обратите внимание, что выражение восстановления будет вычислено только при первом решении события о проблеме. Невозможно решить проблему при помощи выражения восстановления, если условие проблемы всё еще присутствует.

Пример 1

Температура в серверной комнате слишком высокая.

Выражение проблемы:

```
{server:temp.last()}>20
```

Выражение восстановления:

```
{server:temp.last()}<=15
```

Пример 2

Очень мало свободного места на диске

Выражение проблемы: если меньше 10ГБ за последние 5 минут

```
{server:vfs.fs.size[/,free].max(5m)}<10G
```

Выражение восстановления: если больше 40ГБ за последние 10 минут

```
{server:vfs.fs.size[/,free].min(10m)}>40G
```

Выражения с неподдерживаемыми элементами данных и неизвестными значениями

Версии до Zabbix 3.2 очень строго относились к неподдерживаемым элементам данных в выражениях триггеров. Любой неподдерживаемый элемент данных в выражении незамедлительно менял значение триггера на Неизвестно.

Начиная с Zabbix 3.2 существует более гибкий подход к неподдерживаемым элементам данных, допуская неизвестные значения при вычислении выражений:

- У некоторых функций их значения не зависят от того поддерживается ли элемент данных или нет. Такие функции теперь вычисляются даже, если ссылаются на неподдерживаемые элементы данных. Смотрите список в разделе [функции и неподдерживаемые элементы данных](#).
- Логические выражения с ИЛИ и И могут быть вычислены для известных значений в двух случаях независимо от неизвестных операндов:
 - “1 or Неподдерживаемый_элемент_данных1.некая_функция() or Неподдерживаемый_элемент_данных2. некая_функция() or ...” может быть вычислена как '1' (Правда),
 - “0 and Неподдерживаемый_элемент_данных1. некая_функция() and Неподдерживаемый_элемент_данных2. некая_функция() and ...” может быть вычислена как '0' (Ложь).
Zabbix пытается вычислить логические выражения принимая неподдерживаемые элементы данных как Неизвестные значения. В двух случаях, упомянутых выше, будет приниматься известное значение; в остальных случаях значением триггера будет Неизвестно.
- Если вычисление триггера по поддерживаемому элементу данных приведет к ошибке, значением функции будет Неизвестно и оно будет частью дальнейшего вычисления выражения.

Обратите внимание на то, что неизвестные значения могут “исчезать” только в логических выражениях описанных выше. В арифметических выражениях неизвестные Неизвестному результату (за исключением деления на 0).

Если выражение триггера с несколькими неподдерживаемыми элементами данных вычисляется как Неизвестное, сообщение об ошибке в веб-интерфейсе ссылается на последний вычисленный неподдерживаемый элемент данных.

7.3.3 Зависимости триггеров

Обзор

Временами доступность одного узла сети зависит от другого. Сервер, который находится за каким-нибудь роутером будет недоступным, если сам роутер станет недоступным. При наличии настроенных триггеров на обоих устройствах, вы можете получить оповещения сразу для обоих недоступных узлов сети - в то время как виноват только роутер.

В этом случае некоторая зависимость между узлами сети может оказаться очень полезной. С наличием зависимости, набор оповещений о зависимых проблемах, может быть не отправлен и будет отправлено только одно оповещение о корне проблемы.

Хотя Zabbix не поддерживает зависимости между узлами сети напрямую, с другой стороны они могут быть заданы с помощью другого более гибкого метода - зависимостями между триггерами. Триггер может иметь один или более триггеров зависимых от него.

Таким образом в нашем простом примере мы откроем форму настройки триггера и укажем этому триггеру зависимость от триггера роутера. При такой зависимости зависящий триггер сервера от триггера роутера не изменит своё состояние пока триггер роутера находится в состоянии 'ПРОБЛЕМА' - и следовательно зависимые действия не будут предприняты и оповещения не будут отправлены.

Если и сервер, и роутер станут недоступными и зависимость будет настроена, Zabbix не выполнит действия для зависимого триггера.

Действия на зависимые триггера не будут выполняться, если триггер от которого они зависят:

- меняет своё состояние с 'ПРОБЛЕМА' на 'НЕИЗВЕСТНО'
- закрыт вручную, корреляцией или при помощи функций связанных со временем
- решен значением элемента данных, не участвующего в зависимом триггере
- деактивирован, имеет деактивированный элемент данных или узел элемента данных деактивирован

Обратите внимание, что "второй" (зависимый) триггер во всех вышеупомянутых случаях не будет обновлен немедленно.

Также:

- Зависимость триггера может быть добавлена от любого триггера узла сети к триггеру любого другого узла сети, пока это не приведет к циклической зависимости.
- Зависимость триггера может быть добавлена от шаблона к шаблону. Если триггер из шаблона А зависит от триггера из шаблона В, то шаблон А может быть соединен с узлом сети (или с другим шаблоном) только вместе с шаблоном В, но шаблон В может быть соединен с узлом сети (или с другим шаблоном) в одиночку.
- Зависимость триггера может быть добавлена от шаблонного триггера к триггеру узла сети. В этом случае, соединение шаблона с узлом сети создаст триггер у узла сети, который будет зависеть от такого же триггера, что и шаблонный триггер. Это позволяет, например, иметь шаблон, в котором некоторые триггеры зависят от триггеров роутера (узла сети). Все узлы сети соединенные с этим шаблоном будут зависеть от этого конкретного роутера.
- Зависимость триггера узла сети от шаблонного триггера не может быть добавлена.
- Зависимость триггера может быть добавлена от прототипа триггера к другому прототипу триггера (для одного и того же правила низкоуровневого обнаружения) либо к реальному триггеру. Прототип триггера не может быть зависим от прототипа триггера другого правила обнаружения или от триггера созданного из прототипа триггера. Прототип триггера узла сети не может быть зависим от шаблонного триггера.

Настройка

Для того чтобы задать зависимость, откройте [диалог настройки](#) триггера. Далее нажмите на 'Добавить в 'Новая зависимость'' и выберите один или несколько триггеров от которых будет зависеть наш триггер.

The screenshot shows the 'Dependencies' tab of a trigger configuration. A dependency named 'New host: Zabbix agent on {HOST.NAME} is unreachable for 5 minutes' has been added. There is a button labeled 'Add' to add more dependencies.

Нажмите *Обновить*. Теперь триггер имеет в списке индикацию на свою зависимость.

The screenshot shows the 'Dependencies' tab after updating. It lists the dependency '{HOST.NAME} is unreachable' with the note 'Depends on: New host: Zabbix agent on {HOST.NAME} is unreachable for 5 minutes'.

Примеры нескольких зависимостей

Например, узел сети Хост находится за маршрутизатором Роутер2, а Роутер2 находится за Роутер1.

Zabbix - Роутер1 - Роутер2 - Хост

Если Роутер1 недоступен, то очевидно, что и Хост и Роутер2 недоступны. И мы не хотим получать три оповещения с информацией о недоступности Хост, Роутер1 и Роутер2.

Для этого случая, мы зададим эти зависимости:

триггер 'Хост недоступен' зависит от триггера 'Роутер2 недоступен'
триггер 'Роутер2 недоступен' зависит от триггера 'Роутер1 недоступен'

Перед изменением состояния триггера 'Хост недоступен', Zabbix проверит зависимости у соответствующих триггеров. Если они будут найдены, и один из триггеров в находится в состоянии ПРОБЛЕМА, то состояние триггера не будет изменено и, следовательно, действия не будут выполнены и оповещения не будут отправлены.

Zabbix выполняет эту проверку рекурсивно. Если Роутер1 или Роутер2 недоступен, то триггер у Хост не будет изменен.

7.3.4 Важность триггеров

Важность триггера определяет насколько триггер важен. Zabbix поддерживает следующие важности триггеров:

ВАЖНОСТЬ	ОПРЕДЕЛЕНИЕ	ЦВЕТ
Не классифицировано	Неизвестная важность.	Серый
Информация	В информационных целях.	Светло-синий
Предупреждение	Предупреждающий.	Желтый
Средняя	Средняя проблема.	Оранжевый
Высокая	Произошло что-то важное.	Светло-красный
Чрезвычайная	Чрезвычайный. Финансовые потери и т.п.	Красный

Важности 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 высокой важности, email – для остального.
- ограничения действий по условиям важностей триггеров

Имеется возможность [настроить имена важностей триггеров и их цвета](#).

7.3.5 Пользовательские важности триггеров

Имена и цвета важностей триггеров связанные с важностью элементов веб-интерфейса можно настроить в *Администрирование* → *Общие* → *Важности триггеров*. Цвета являются общими для всех тем веб-интерфейса.

Перевод пользовательских имен важностей

Если используются переводы веб-интерфейса Zabbix, по умолчанию пользовательские имена важностей будут заменены переведенными именами.

Имена важностей триггеров по умолчанию доступны для перевода для всех локализаций. Если имя важности изменено, пользовательское имя будет использовано во всех локализациях и дополнительно потребуется ручной перевод.

Процедура перевода имени пользовательской важности:

- задайте требуемое имя пользовательской важности, например 'Важное'
- измените <папка_с_веб-интерфейсом>/locale/<требуемая_локаль>/LC_MESSAGES/frontend.po
- добавьте 2 строки:

```
msgid "Important"
msgstr "<переведенная строка>"
```

и сохраните файл

- создайте .mo файлы, как описано в <папка_с_веб-интерфейсом>/locale/README

Здесь **msgid** должно совпадать с новым именем пользовательской важности и **msgstr** должно являться переводом этой важности для указанного языка.

Эта процедура должна быть выполнена после каждого изменения имени важности.

7.3.6 Массовое обновление

Обзор

Используя массовое обновление, вы можете изменить любой параметр для нескольких триггеров за один раз, что спасет вас от необходимости открывать каждый отдельный триггер для изменения.

Использование массового обновления

Для массового обновления каких-либо триггеров, выполните следующее:

- Отметьте триггеры из списка для обновления
- Выберите *Массовое изменение* ниже списка
- Отметьте атрибуты для обновления
- Введите новые значения для этих атрибутов и нажмите на *Обновить*.

Severity <input checked="" type="checkbox"/>	Not classified	Information	Warning	Average	High	Disaster	
Replace dependencies <input checked="" type="checkbox"/>	Name Zabbix server 1: Disk I/O is overloaded on {HOST.NAME} Add						
Replace tags <input checked="" type="checkbox"/>	<input type="text" value="tag"/> <input type="text" value="value"/>			Remove			
Allow manual close <input checked="" type="checkbox"/>	No	Yes					
Update Cancel							

Замена зависимостей заменит существующие зависимости (если они есть) на новые, указанные в массовом обновлении.

7.3.7 Прогнозирующие функции триггеров

Обзор

Иногда имеются признаки надвигающейся проблемы. Эти признаки можно заметить, поэтому 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.

7.3.7.1 Функции

Необходимо знать две вещи, а именно то, как описать состояние проблемы и сколько времени нужно, чтобы предпринять меры. Далее есть два способа создать триггер, сигнализирующий о возможной нежелательной ситуации. Первый: триггер должен “загореться”, когда система после “пора действовать”, как ожидается, будет в состоянии проблемы. Второй: триггер должен “загореться”, если система перейдет в состояние проблемы за время меньше, чем “пора действовать”. Необходимо воспользоваться функциями триггеров, а именно **forecast** и **timeleft** соответственно. Надо заметить, что лежащий в основе статистический анализ, в основном идентичен у этих двух функций. Вы можете настроить триггер в зависимости от того, какой путь более предпочтителен для вас, и результаты будут идентичны.

7.3.7.2 Параметры

Обе функции используют почти одинаковый набор параметров. Обратитесь к списку [поддерживаемых функций](#) за справкой.

2.1 Интервал времени

Прежде всего вам необходимо указать период истории, который Zabbix должен проанализировать для составления прогноза. Сделайте это привычным способом при помощи параметров **sec** или **#num** и необязательного **сдвиг_времени** по аналогии с функциями **avg**, **count**, **delta**, **max**, **min** и **sum**.

2.2 Горизонт предсказания

(Только **forecast**)

Параметр **время** определяет, насколько далеко в будущее Zabbix будет экстраполировать зависимости, которые ему удастся найти в исторических данных. Независимо о того, используете ли вы **СДВИГ_времени** или нет, время всегда отсчитывается от текущего момента.

2.3 Достигаемый порог

(Только **timeleft**)

Параметр **ПОРОГ** определяет значение, которого должен достичь анализируемый элемент данных, нет никакой разницы, сверху или снизу. После того, как мы определили $f(t)$ (смотри ниже), мы должны решить уравнение $f(t) = \text{ПОРОГ}$ и вернуть ближайший к текущему моменту корень, который находится справа от текущего момента, или вернуть 999999999999.9999, если таковых корней нет.

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 **ПОРОГА**, если таковое имеется. Наилучшим решением является использование прогнозирований наряду с обычными диагностиками проблем, а не заменой одним на другое.¹⁾

2.4 Функции аппроксимации

По умолчанию **аппроксимация** является линейной (*linear*) функцией. Если наблюдаемая система более сложная, вы можете выбрать один из следующих вариантов.

аппроксимация

$x = f(t)$

линейная (*linear*)

$x = a + b*t$

полином (*polynomialN*)²⁾

$x = a_0 + a_1*t + a_2*t^2 + \dots + a_n*t^n$

экспоненциальная (*exponential*)

$x = a * \exp(b*t)$

логарифмическая (*logarithmic*)

$x = a + b * \log(t)$

степенная (*power*)

$x = a*t^b$

2.5 Режимы

(Только **forecast**)

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 аппроксимирующей функции 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 **аппроксимации** (вроде *polynomial*) просто лишь одно значение из будущего может ввести в заблуждение.

режим

результат **forecast**

значение (*value*) $f(\text{сейчас} + \text{время})$

максимум (*max*) $\max_{\text{сейчас} \leq t \leq \text{сейчас} + \text{время}} f(t)$

минимум (*min*) $\min_{\text{сейчас} \leq t \leq \text{сейчас} + \text{время}} f(t)$

дельта (*delta*) $\max - \min$

среднее (*avg*) среднее значение $f(t)$ ($\text{сейчас} \leq t \leq \text{сейчас} + \text{время}$) в соответствии с определением

7.3.7.3 Подробности

Для того, чтобы избежать вычислений с большими числами, мы рассматриваем штамп времени первого значения в указанном периоде плюс 1 наносекунда как новую точку остатка времени (текущие штампы времени порядка 10^9 , в квадрате уже 10^{18} , а точность дробных значений около 10^{-16}). 1 нс прибавляется для того, чтобы все значения времени были положительными, поскольку построение логарифмической и степенной аппроксимаций подразумевает вычисление $\log(t)$. Этот сдвиг времени не влияет на линейную функцию, полином и экспоненциальную функции (за исключением более легких и более точных вычислений), но изменяет форму логарифмической и степенной функций.

7.3.7.4 Возможные ошибки

Функции возвращают -1 в следующих ситуациях:

- указанных период не содержит данных;
- математическая операция не задана³⁾;
- сложности вычислений (к сожалению, для некоторых наборов входных данных диапазона и точности формата чисел с плавающей точкой двойной точности становится недостаточно)⁴⁾.

Никаких предупреждений или ошибок не выдаётся, если выбранная аппроксимирующая функция плохо описывает предоставленные данные или просто данных недостаточно для точного прогноза.

7.3.7.5 Примеры и обработка ошибок

Для получения предупреждения о том, что у вашего узла сети скоро закончится свободное дисковое пространство, вы можете использовать следующее выражение триггера:

```
{host:vfs.fs.size[/,free].timeleft(1h,,0)}<1h
```

Однако, может появиться код ошибки -1 и перевести ваш триггер в состояние проблемы. Вообще говоря, это не плохо, потому что вы получите предупреждение о том, что ваши прогнозирования не работает должным образом, и вам стоит обратить на них внимание, чтобы разобраться почему. Но иногда это плохо, потому что -1 может просто означать, что за последний час не было получено никаких данных о свободном дисковом пространстве данного узла сети. Если вы получаете много сообщений о ложных тревогах, подумайте об использовании более сложного выражения⁵⁾ триггера:

```
{host:vfs.fs.size[/,free].timeleft(1h,,0)}<1h and  
{host:vfs.fs.size[/,free].timeleft(1h,,0)}<>-1
```

Ситуация немного сложнее с **forecast**. Начнём с того, что -1 может перевести, а может и не перевести триггер в состояние проблемы в зависимости от вашего выражения триггера, которое может быть вроде

```
{host:item.forecast(...)}<...
```

или наподобие

```
{host:item.forecast(...)}>...
```

Более того, -1 может быть вполне корректным результатом прогнозирования, если прогнозируемый элемент данных может принимать отрицательные значения. Но вероятность возникновения такой ситуации в реальных условиях пренебрежительно мала (смотрите, [как](#) работает оператор =). Поэтому добавьте

```
... or {host:item.forecast(...)}=-1
```

или

```
... and {host:item.forecast(...)}<>-1
```

если вы соответсвенно хотите или, наоборот, не хотите рассматривать -1 как проблему.

Смотрите также

1. [Прогнозирующие функции триггеров \(pdf\)](#) on zabbix.org [en]

[1\)](#)

Например, простой триггер наподобие

```
{host:item.timeleft(1h,,X)} < 1h
```

может перейти в состояние проблемы, когда элемент данных приближается к X, и неожиданно “самоустраниться”, как только значение X достигнуто. Если проблема выражается в том, что значение элемента данных меньше X, используйте:

```
{host:item.last()} < X or {host:item.timeleft(1h,,X)} < 1h
```

Если проблема заключается в том, что значение элемента данных больше X, используйте:

```
{host:item.last()} > X or {host:item.timeleft(1h,,X)} < 1h
```

[2\)](#)

Степень многочлена может быть от 1 до 6, при этом функция *polynomial1* равнозначна *linear*. Однако, использовать полиномы высоких степеней следует [с осторожностью](#). Если период вычисления содержит меньше точек, чем требуется для определения коэффициентов полинома, степень полинома понизится (например, запрашивается *polynomial5*, но есть только 4 точки, поэтому для аппроксимации будет использована *polynomial3*).

[3\)](#)

Например, построение экспоненциальной и степенной функций требует логарифмических вычислений значений элемента данных. Если данные содержат нулевые или отрицательные числа, вы получите сообщение об ошибке, поскольку вычисление логарифма возможно только при положительных значениях.

[4\)](#)

При аппроксимации *linear*, *exponential*, *logarithmic* и *power* выражений все необходимые вычисления можно написать в явном виде. При *polynomial* можно вычислить только *value* без дополнительных шагов. Вычисление *avg* включает в себя вычисление первообразной полинома (аналитически). Вычисление *max*, *min* и *delta* включает в себя вычисление производной полинома (аналитически) и поиск его корней (численно). Решение $f(t) = 0$ требует нахождения корней полинома (численно).

[5\)](#)

Но в этом случае -1 может восстановить триггер из состояния проблемы. Для полной защищённости используйте:

```
{host:vfs.fs.size[/,free].timeleft(1h,,0)}<1h and ({TRIGGER.VALUE}=0 and {host:vfs.fs.size[/,free].timeleft(1h,,0)}<>-1 or {TRIGGER.VALUE}=1)
```

7.4 События

Обзор

События в Zabbix генерируются несколькими источниками:

- события на триггеры - всякий раз, когда триггер меняет свое состояние(OK → ПРОБЛЕМА → OK)
- события на обнаружение - при обнаружении узлов сети или сервисов
- события на авторегистрацию - когда активные агенты автоматически регистрируются сервером
- внутренние события - когда элементы данных/правила низкоуровневого обнаружения становятся не поддерживаемыми или триггер переходит в состояние неизвестно

Внутренние события поддерживаются начиная с Zabbix 2.2.

События имеют штамп времени и могут быть основой для действий, таких как отправка оповещений по почте и т.п.

Для просмотра деталей событий в веб-интерфейсе, перейдите в *Мониторинг* → *События*. Здесь вы можете нажать на дату и время события и просмотреть его детали.

Более подробная информация доступна в:

- [события на триггеры](#)
- [другие источники событий](#)

7.4.1 Генерация событий на триггеры

Обзор

Изменение состояния триггера является наиболее частым и наиболее важным источником событий. Каждый раз, когда триггер меняет свое состояние, генерируется событие. Событие содержит подробную информацию о изменении состояния триггера - когда это случилось, и какое сейчас новое состояние.

Триггерами создаются два типа событий - Проблема и OK.

События проблемы

Событие проблемы создается:

- когда выражение триггера вычисляется как ПРАВДА, если триггер в состоянии OK;
- каждый раз выражение триггера вычисляется как ПРАВДА, если у триггера активировано множественная генерация событий проблема.

События OK

Событие OK закрывает связанные событие(ия) о проблеме и может быть создано тремя компонентами:

- триггеры - основываясь на настройках 'Генерация OK событий' и 'OK событие закрывает';
- корреляция событий
- менеджер задач – когда событие [закрывается вручную](#)

Триггеры

У триггеров имеется настройка 'Формирование OK событий', которая определяет, как генерируются OK события:

- *Выражение* - событие OK генерируется триггеру в состоянии проблема, когда выражение этого триггера вычисляется как ЛОЖЬ. Это самая простая настройка, активированная по умолчанию.
- *Выражение восстановления* - событие OK генерируется триггеру в состоянии проблема, когда выражение этого триггера вычисляется как ЛОЖЬ и выражение восстановления вычисляется

как ПРАВДА. Эту опцию можно использовать, если критерий восстановления триггера отличается от критерия проблемы.

- *Hem* - событие OK никогда не генерируется. Эту опцию можно использовать в сочетании с множественной генерацией проблем для простой отправки оповещений, когда что-то происходит.

Дополнительно, у триггеров имеется настройка 'OK событие закрывает', которая определяет какие события о проблемах закрываются:

- *Все проблемы* - событие OK закрывает все открытые проблемы, которые созданы 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 корреляция событий на уровне триггеров.

Корреляция событий

Корреляция событий (также называемая глобальной корреляцией событий) является способом настройки пользовательских правил закрытия (в результате чего генерируются OK события).

Правила определяют каким образом новые события о проблемах соотносятся с существующими событиями о проблемах 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 добавить условие на основе старого события, если используется операция 'закрыть старые события', или все существующие проблемы могут быть закрыты
3. избегайте использования распространенных имен тегов, которые используются различными настройками корреляции

Менеджер задач

Если у триггера активирована настройка 'Разрешить закрывать вручную', тогда появляется возможность закрывать события о проблемах, которые созданы этим триггером, вручную. Это можно сделать в веб-интерфейсе при [обновлении проблемы](#). Событие не закрывается напрямую – вместо 'закрытия события' создается задача, которая будет обработана менеджером задач в самое короткое время. Диспетчер задач создаст соответствующее OK событие и событие о проблеме закроется.

7.4.2 Другие источники событий

События на обнаружения

Zabbix периодически сканирует диапазоны IP адресов заданные в правилах сетевого обнаружения. Частота этой проверки настраивается индивидуально для каждого правила. 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 может создаваться, 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 от агента.

Для получения более подробной информации, смотрите страницу об [автоматической регистрации активных агентов](#).

Внутренние события

Внутренние события возникают, когда:

- элемент данных меняет свое состояние с “нормального” на “неподдерживается”
- элемент данных меняет свое состояние с “неподдерживается” на “нормальное”
- правило низкоуровневого обнаружения меняет свое состояние с “нормального” на “неподдерживается”
- правило низкоуровневого обнаружения меняет свое состояние с “неподдерживается” на “нормальный”
- триггер меняет свое состояние с “нормального” на “неизвестное”
- триггер меняет свое состояние с “неизвестного” на “нормальное”

Внутренние события поддерживаются начиная с Zabbix 2.2. Целью введения внутренних событий - дать знать пользователям, когда происходят некоторые внутренние события, например, элемент данных становится неподдерживаемым и перестает собирать данные.

7.4.3 Закрытие проблем вручную

Обзор

Хотя в целом события о проблемах решаются автоматически, когда состояние триггера переходит из 'Проблема' в 'OK', могут быть случаи когда трудно определить при помощи выражения триггера что проблема решена. В таких случаях проблема должна быть решена 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 “закрыта вручную”, Zabbix генерирует новую внутреннюю задачу для Zabbix сервера. Затем процесс *task manager* выполняет эту задачу и генерирует OK событие, следовательно закрывает событие о проблеме.

Вручную закрытая проблема не означает, что основной триггер никогда больше не перейдет в состояние 'Проблема'. При поступлении новых данных от любого элемента данных, которые включены в выражение триггера, произойдет повторное вычисление всего выражения, что может привести снова к проблеме. Также триггер вычисляется повторно, когда в его выражении используются функции связанные со временем. Полный список функций связанных со временем можно найти на [странице с Триггерами](#).

Настройка

Необходимо выполнить два шага для закрытия проблем вручную.

Настройка триггера

В настройке триггера включите опцию *Разрешить закрывать вручную*.

Allow manual close	<input checked="" type="checkbox"/>
--------------------	-------------------------------------

Экран обновления проблемы

Если возникает проблема у триггера с флагом *Закрыть вручную*, вы можете перейти на экран [обновления проблемы](#) этого триггера и закрыть проблему вручную.

Чтобы закрыть проблему, проверьте наличие опции *Закрыть проблему* в диалоге и нажмите на *Обновить*.

Update problem

Message	<input type="text" value="Fixed, closing."/>	
History		
Scope	<input checked="" type="radio"/> Only selected problem <input type="radio"/> Selected and all other problems of related triggers 1 event	
Change severity	<input type="checkbox"/> Not classified <input type="checkbox"/> Information <input type="checkbox"/> Warning <input type="checkbox"/> Average <input type="checkbox"/> High	
Acknowledge	<input type="checkbox"/>	
Close problem	<input checked="" type="checkbox"/>	
<small>* At least one update operation or message must exist.</small>		
	<input style="background-color: #0072bc; color: white; border: 1px solid #0072bc; padding: 2px 10px; border-radius: 3px;" type="button" value="Update"/>	<input style="border: 1px solid #ccc; padding: 2px 10px; border-radius: 3px;" type="button" value="Cancel"/>

Все обязательные поля ввода отмечены красной звёздочкой.

Запрос выполняется Zabbix сервером. Обычно такая процедура занимает несколько секунд до закрытия проблемы. В процессе этой процедуры в *Мониторинг → Проблемы* будет отображаться **ЗАКРЫТИЕ** как состояние проблемы.

Проверка

Можно проверить что проблема была закрыты вручную:

- в деталях события, доступных через *Мониторинг → Проблемы*;
- используя макрос {EVENT.UPDATE.HISTORY} в сообщениях оповещения, который предоставит такую информацию.

7.5 Корреляция событий

Обзор

Корреляция событий позволяет сопоставлять события о проблемах с их решениями очень точным и гибким способом.

Корреляцию событий можно определять:

- на уровне триггера - один триггер можно использовать для связки отдельных проблем с их решением
- глобально - проблемы можно соотнести с их решением из другого триггера/метода опроса, используя глобальные правила корреляции

7.5.1 Корреляция событий на основе триггеров

Обзор

Корреляция событий на основе триггеров позволяет сопоставлять отдельные проблемы, о которых сообщает один триггер.

В то время как OK события закрывают в Zabbix все события о проблеме, 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 триггеры имеют активированную опцию *Формирование множественных Проблема событий*. 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 проблеме. 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 выявлены одно за другим и закрыты отдельно на основании метки при помощи тега событий.

Как это работает

В мониторинге журналов вы можете встретиться со строками как эти:

Строка1: Приложение 1 остановлено

Строка2: Приложение 2 остановлено
Строка3: Приложение 1 перезапущено
Строка4: Приложение 2 перезапущено

Идея корреляции событий состоит в том, чтобы была возможность сопоставить событие о проблеме со Строки1 с решением со Строки3 и событие о проблеме со Строки2 с решением со Строки4 и закрыть эти проблемы одну за другой:

Строка1: Приложение 1 остановлено
Строка3: Приложение 1 перезапущено #проблема со Строки 1 закрыта

Строка2: Приложение 2 остановлено
Строка4: Приложение 2 перезапущено #проблема со Строки 2 закрыта

Чтобы сделать подобное вам необходимо отметить эти связанные события как, например, "Приложение 1" и "Приложение 2". Это можно сделать применив регулярное выражение к строке из файла журнала, чтобы извлечь значение тега. Затем, когда события будут созданы, они будут промаркированы как "Приложение 1" и "Приложение 2" соответственно и проблема может быть сопоставлена решению.

Настройка

Для настройки корреляции на уровне триггера:

- перейдите в [диалог настройки](#) триггера

The screenshot shows the 'Dependencies' tab of a Nagios configuration interface. Key settings include:

- Name:** Service {{ITEM.VALUE}}.regsub(".* service ([a-zA-Z]*).*\$", "1") stopped
- Severity:** High
- Expression:** {Template Services:log[/var/log/messages].regexp("stopped")}=1
- OK event generation:** Recovery expression
- Recovery expression:** {Template Services:log[/var/log/messages].regexp("started")}=1
- PROBLEM event generation mode:** Multiple
- OK event closes:** All problems if tag values match
- Tag for matching:** Service
- Tags:** Service, {{ITEM.VALUE}}.regsub(".* service ([a-zA-Z]*).*\$", "1"), Datacenter, value

Все обязательные поля ввода отмечены красной звёздочкой.

- выберите 'Режим формирования Проблема событий' значение **Множественный**
- в 'OK событие закрывает' выберите *Все проблемы, если значения тегов совпадают*
- введите имя тега для сопоставления событий
- настройте [теги](#) для извлечения значений тега со строк журнала

При успешной настройке у вас будет возможность видеть события о проблемах промаркованные приложением и сопоставленные с их решением в *Мониторинг → Проблемы*.

The screenshot shows a table titled 'Problems' with the following data:

Time	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags
08:38:18	High	08:38:18	RESOLVED		Zabbix server	Service Apache stopped	0	No		Service: Apache Webserver

Посколько возможна некорректная настройка, когда похожие теги событий могут быть созданы по **нерешаемым** проблемам, пожалуйста, рассмотрите указанные ниже подобные случаи!

- При наличии двух приложений, которые записывают сообщения об ошибках и восстановлениях в один и тот же файл журнала, пользователь может решить использование двух *Приложение* тегов в одном триггере с различными значениями тегов, используя разные регулярные выражения в значениях тегов для извлечения имен, так сказать приложение А и приложение В из {ITEM.VALUE} макроса (например, если форматы сообщений различаются). Однако, такой подход может не сработать как задумано, если регулярным выражениям будут отсутствовать соответствия. Несоответствующие регулярные выражения могут выдать пустые значения тегов и одного пустого значения тега как по событиям о проблеме, так и по ОК событиям, будет достаточно чтобы их соотнести. Таким образом сообщение о восстановлении с приложения А может случайно закрыть сообщение об ошибке с приложения В.
- Реальные теги и значения тегов становятся видимы только когда срабатывает триггер. Если используемое регулярное выражение ошибочно, оно автоматически заменяется на *НЕИЗВЕСТНО* строку. Если изначальное событие о проблеме со значением тега *НЕИЗВЕСТНО* отсутствует, тогда могут появиться последующие ОК события с тем же значением тега *НЕИЗВЕСТНО*, которые могут закрыть события о проблемах, которые они не должны закрывать.
- Если пользователь использует {ITEM.VALUE} макрос без функций макроса в качестве значения тега, тогда будет применяться ограничение по длине строки в 255-символ. Когда в журнале имеются длинные сообщения и первые 255 символов не конкретизируют проблему, может привести к идентичным тегам событий по нерешаемым проблемам.

Теги событий

Обзор

Существует возможность задать в Zabbix пользовательские теги событий. Теги событий задаются на уровне триггера. После того как теги добавлены, соответствующие новые события маркируются данными тегами.

Наличие пользовательских тегов событий даёт большую гибкость. Наиболее важно, что события можно скоррелировать, основываясь на тегах событий. В качестве другого применения, действия можно назначать на основе тегов событий.

Теги событий реализованы парой *имя тега и значение*. Вы можете использовать только имя или пару имени со значением:

MySQL, Сервис:MySQL, Сервисы, Сервисы:Клиент, Приложения, Приложение:Java,
Приоритет:Высокий

Сценарии применения

Некоторые сценарии использования этой функции заключаются в следующем:

1. Распознавание проблем в файле журнала и раздельное их закрытие

- Добавьте теги в триггер к журналу, которые будут распознавать события, используя значение извлеченное при помощи макроса {{ITEM.VALUE<N>.regsub()}}; * В настройках триггера укажите режим формирования множественных проблема событий; * В настройках триггера используйте [:ru/manual/config/event_correlation|корреляцию событий]]: выберите опцию, которая закроет ОК события только в случае совпадения событий и укажите тег поиска совпадений; * Просматривайте события о проблемах созданные с тегом и закрытые по отдельности. - Использование тегов для фильтрации оповещений * Добавьте теги на уровне триггера, чтобы промаркировать события при помощи различных тегов; * Используйте фильтрацию по тегу в условиях действия чтобы получать оповещения только по тем событиям, которые совпадают с данными тега. - Просмотр информации по тегу в веб-интерфейсе * Добавьте теги на уровне триггера, чтобы промаркировать события при помощи различных тегов; * Просматривайте эту информацию в //Мониторинг// -> //Проблемы//. - Использование информации извлеченной из значения элемента данных значением тега * Используйте макрос {{ITEM.VALUE<N>.regsub()}} в значении тега; * Просматривайте значения тегов в //Мониторинг// -> //Проблемы// как извлеченные данные из значения элемента данных. - Лучшее распознавание проблем в оповещениях * Задайте теги на уровне триггера; * Используйте макрос {EVENT.TAGS} в оповещениях на проблемы; * В итоге более простое распознавание о каких приложениях/сервисах идет речь в оповещении. - Упрощение задач настройки при использовании тегов на уровне шаблона * Задайте теги на уровне шаблонного триггера; * Просматривайте эти теги у всех триггеров созданных из шаблонных триггеров. - Создание триггеров с тегами из низкоуровневого обнаружения (LLD) * Задайте теги прототипам триггеров; * Используйте LLD макросы в имени тега или значении; * Просматривайте эти теги у всех триггеров созданных из прототипов триггеров. === Настройка === Теги событий задаются при настройке триггера. Теги событий можно задать для триггеров, шаблонных триггеров и прототипов триггеров. {{:manual:config:triggers:event_tags.png}} === Поддержка макросов === В тегах событий можно использовать следующие макросы: * Макросы {{ITEM.VALUE}}, {{ITEM.LASTVALUE}}, {{HOST.HOST}}, {{HOST.NAME}}, {{HOST.CONN}}, {{HOST.DNS}}, {{HOST.IP}}, {{HOST.PORT}} и {{HOST.ID}} можно использовать, чтобы заполнять имя тега или значение тега. * [:ru/manual/appendix/macros/supported_by_location|Макросы] {{INVENTORY.*}} можно использовать в качестве ссылки на значения инвентарных данных с одного или нескольких узлов сети в выражении триггера (поддерживается начиная с 4.0.0). * В именах тегов и значениях тегов поддерживаются [:ru/manual/config/macros/usermacros|пользовательские макросы]], а также контекст пользовательских макросов. Контекст пользовательских макросов может включать в себя макросы низкоуровневого обнаружения. * Можно использовать макросы низкоуровневого обнаружения в имени тега и значении тега в прототипах триггеров. Можно использовать макросы {EVENT.TAGS} и {EVENT.RECOVERY.TAGS} в оповещениях на основе триггеров и они будут раскрыты в разделенный запятыми список тегов события или тегов события восстановления. == Извлечение подстроки == Поддерживается извлечение подстроки для заполнения имени тега или значения тега при использовании новой [:ru/manual/config/macros/macro_functions|функции макросов]] - применение регулярного выражения к значению полученному при помощи {{ITEM.VALUE}}, {{ITEM.LASTVALUE}} макросов, а также макросов низкоуровневого обнаружения. Например: {{ITEM.VALUE}.regsub(шаблон, вывод)} {{ITEM.VALUE}.iregsub(шаблон, вывод)} {{#LLDMAKPOC}.regsub(шаблон, вывод)} {{#LLDMAKPOC}.iregsub(шаблон, вывод)}

вывод)} Имя тега и значение будут обрезаны до 255 символов, если их длина превысит 255 символов после раскрытия макросов. Смотрите также: Использование функций макросов в [[:ru/manual/config/macros/lld_macros#использование_функций_макросов| макросах низкоуровневого обнаружения]] для тегирования событий. === Просмотр тегов события === Теги событий, если заданы, можно просмотреть с новыми событиями в: * //Мониторинг// -> //Проблемы// * //Мониторинг// -> //Проблемы// -> //Детали события// * //Мониторинг// -> //Панель// -> виджет //Проблемы// (во всплывающем окне, которое открывается при наведении курсора мыши на имя проблемы) {{:manual:config:triggers:event_tags_view.png?600|}} Отображаются только первые три записи тегов. Если имеется более трех записей тегов, такой факт будет обозначен тремя точками. Если вы проведете курсором мыши над этими тремя точками, во всплывающем окне отобразятся все записи тегов. Обратите внимание, что на порядок в котором отображаются теги влияет фильтрация тегов и опция //Приоритет отображения тегов// в фильтре //Мониторинг// -> //Проблемы// или в виджете панели //Проблемы/. === Влияние на производительность и хранение === Ожидается, что использование тегов событий может иметь следующие последствия: * Обработка событий будет происходить медленнее по причине создания новых записей в таблице тегов событий. * Операции связанные с манипуляциями с триггерами унаследованными из шаблонов будут медленнее так как имеется необходимость создания по одной записи на каждый тег в таблице тегов триггеров. По этой причине создание, обновление и удаление триггеров будет более медленным. * Синхронизация кэша конфигурации будет медленнее по причине наличия тегов триггеров. * Хранение тегов событий потребует дополнительное пространство на диске, которое может быть сравнимо с размером существующих таблиц триггеров и событий. Конкретные требования к хранению очень сильно зависят от количества тегов созданных по каждому триggerу и событию.

7.5.2 Глобальная корреляция событий

Обзор

Глобальная корреляция событий позволяют 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. Созданием интеллектуальные правил корреляции можно обезопасить себя от тысяч повторяющихся оповещений и сфокусироваться на реальных причинах проблемы!

Глобальная корреляция событий - мощный механизм, который позволяет вам освободить себя от логики проблемы и решения основанных на одном триггере. До сих пор одно событие о проблеме создавалось одним триггером 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 меток 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 меткой триггера на основе опроса будет *Состояние: Доступен*. Затем, в глобальном правиле корреляции вы можете соотнести эти триггеры и назначить соответствующую операцию на такую 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 можем получить только одно сообщение о проблемах по каждому сетевому порту? Не требуется сообщать по ним всем. 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 OK события). Если проблема закрыта при помощи глобальной корреляции, об этом сообщается в *Инфо* колонке в *Мониторинг → Проблемы*.

Настройка глобальных правил корреляции доступна пользователям только с Zabbix Супер Администратор уровнем.

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 предполагается (в худшем случае все события о проблемах могут быть закрыты).

Для безопасной настройки глобальной корреляции обратите внимание на следующие важные советы:

- Уменьшайте масштаб корреляции. Всегда указывайте уникальный тег для нового события, который соотносится со старыми событиями и используйте *Тег нового события* условие корреляции;
- Добавляйте условие, которое основывается на старом событии, при использовании *Закрыть старое событие* операции (в противном случае все существующие проблемы могут быть закрыты);
- Избегайте использования общих имен тегов, которые в итоге могут быть использованы различными конфигурациями корреляции;
- Сведите количество правил корреляции к действительно необходимым вам.

Смотрите также: [известные проблемы](#).

Настройка

Для настройки глобальных правил корреляции событий:

- Перейдите в *Настройка → Корреляция событий*
- Нажмите на *Создать корреляцию* справа (или на имя корреляции, чтобы изменить уже существующее правило)
- В диалоге введите параметры правила корреляции

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Описание

Имя	Уникальное имя правила корреляции. Для вычисления условий доступны следующие опции: И - должны быть выполнены все условия Или - достаточно выполнения одного условия И/Или - И с разными типами условий и ИЛИ с одинаковым типом условий
Тип вычисления	Пользовательское выражение - формула вычисления, введенная пользователем, для оценки условий действия. Она должна включать в себя все условия (представленные в виде прописных букв А, В, С, ...) и может включать пробелы, символы табуляции, скобки (), and (с учетом регистра), or (с учетом регистра), not (регистrozависимое).
Условия	Список условий, выбранных из <i>Новое условие</i> поля. Выберите условия при которых события будут соотноситься и нажмите на <i>Добавить</i> . Доступны следующие условия: Тег старого события - совпадение нового события со старым событием(ями), которые соответствуют тегу старого события Тег нового события - совпадение нового события, которое соответствует тегу события, старому событию(ям) Группа узлов сети нового события - совпадение нового события, которое принадлежит соответствующей группе узлов сети, старому событию(ям) Пара тегов событий - совпадение нового события старому событию(ям), если значения указанных тегов совпадают в обоих событиях. Имена тегов не должны совпадать. Эта опция может быть полезна для сопоставления значений в реальном времени, которые могут быть неизвестны в процессе настройки (смотрите также Пример 1) Значение тега старого события - совпадение нового события старому событию(ям), которое: = - имеет соответствующее значение тега старого события <> - не имеет соответствующее значение тега старого события содержит - имеет соответствующую строку в значении тега старого события
Новое условие	

Параметр

Описание

не содержит - не имеет соответствующую строку в значении тега старого события
Значение тега нового события - совпадение нового события старому событию(ям), если новое событие:
= - имеет соответствующее значение тега нового события
<> - не имеет соответствующее значение тега нового события
содержит - имеет соответствующую строку в значении тега нового события
не содержит - не имеет соответствующую строку в значении тега нового события

Описание Описание правила корреляции.

Активировано Если вы выберите этот параметр, правило корреляции будет активировано.

- В диалоге выберите операцию правила корреляции в диалоге

The screenshot shows the 'Correlation' dialog with the 'Operations' tab selected. On the left, there's a list of operations under 'Operations': 'Close old events' (with an 'Action' button labeled 'Remove') and 'New operation' (with a dropdown menu showing 'Close new event' and an 'Add' button). The 'Details' section is visible above the operations list.

Параметр

Описание

Операции Список операций, выбранных из *Новая операция* поля.

Выберите выполняемую операцию при соответствии события и нажмите на *Добавить*. Доступны следующие операции:

Новая операция **Закрыть старые события** - закрыть старые события при появлении нового события.
Всегда добавляйте условие на основе старого события при использовании *Закрыть старые события* операции или будут закрыты все существующие проблемы.
Закрыть новое событие - закрыть новое событие при его появлении

Посколько возможна некорректная настройка, когда похожие теги событий могут быть созданы по **нерешаемым** проблемам, пожалуйста, рассмотрите указанные ниже подобные случаи!

- Реальные теги и значения тегов становятся видимы только когда срабатывает триггер. Если используемое регулярное выражение ошибочно, оно автоматически заменяется на ***НЕИЗВЕСТНО*** строку. Если изначальное событие о проблеме со значением тега ***НЕИЗВЕСТНО*** отсутствует, тогда могут появиться последующие OK события с тем же значением тега ***НЕИЗВЕСТНО***, которые могут закрыть события о проблемах, которые они не должны закрывать.
- Если пользователь использует **{ITEM.VALUE}** макрос без функций макроса в качестве значения тега, тогда будет применяться ограничение по длине строки в 255-символ. Когда в журнале имеются длинные сообщения и первые 255 символов не конкретизируют проблему, может привести к идентичным тегам событий по нерешаемым проблемам.

Примеры

Пример 1

Остановка повторяющихся событий о проблемах с одного и того же сетевого порта.

Correlation **Operations**

* Name	Correlate network port problems			
Type of calculation	And	A and B		
* Conditions	Label	Name	Action	
	A	Old event tag Port = new event tag Port	Remove	
	B	Old event tag Host = new event tag Host	Remove	
New condition	Event tag pair	old event tag	=	new event tag
	Add			
Description	Keep only one problem per port. No need to report all of them.			
Enabled	<input checked="" type="checkbox"/>			
	Add	Cancel		

Такое глобальное правило корреляции сопоставит проблемы, если у триггера существуют Узел сети и Порт значения тегов и они такие же как у оригинального события, так и у нового события.

Correlation **Operations**

* Operations	Details	Action
	Close new event	Remove

Эта операция закроет все события о проблеме по тому же самому сетевому порту, оставив открытой только оригинальную проблему.

7.6 Визуализация

7.6.1 Графики

Обзор

При наличии большого потока данных поступающего в Zabbix данных, намного проще для пользователей, если они смогут видеть визуальное представление о происходящем, а не только числа.

Именно в этом случае приходят графики. Графики позволяют понять поток данных просто взглянув на него, соотнести проблемы, обнаружить когда что-то началось или выступить с докладом о том, 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, используя [ситуационные графики](#)

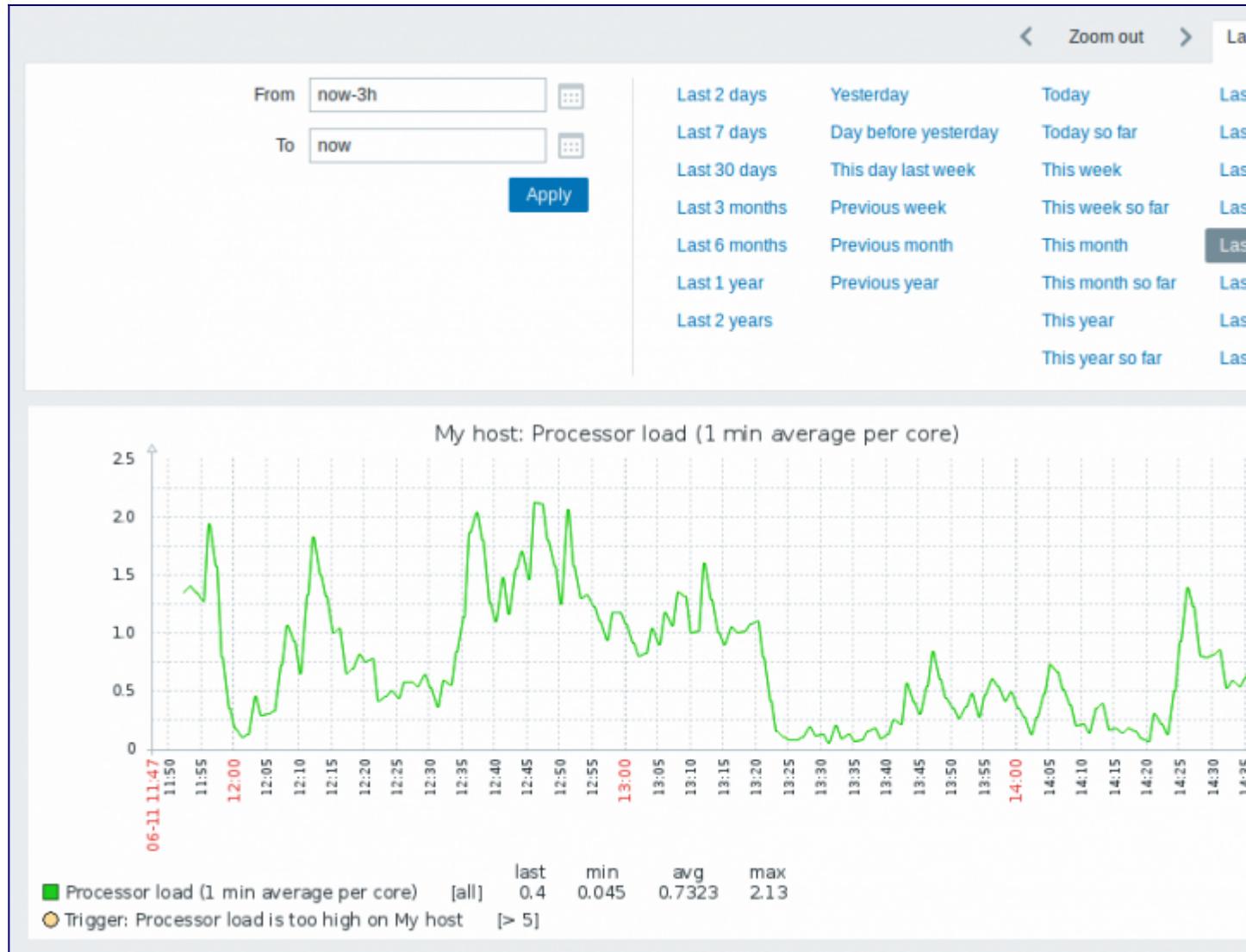
7.6.1.1 Простые графики

Обзор

Простые графики служат для визуализации данных собранных элементами данных.

Со стороны пользователя не требуются никакие усилия по настройке простых графиков для их просмотра. Они свободно доступны через Zabbix.

Просто перейдите в *Мониторинг* → *Последние данные* и нажмите на ссылку Графика напротив соответствующего элемента данных и график отобразится.



Простые графики доступны для всех числовых элементов данных. Для текстовых элементов данных доступна ссылка на Историю в *Мониторинг* → *Последние данные*.

Выбор периода времени

Обратите внимание на выбор периода времени над графиком. Он позволяет выбрать часто используемые периоды одним щелчком мыши.

Обратите внимание, что такие опции как *Сегодня*, *Эта неделя*, *Этот месяц*, *Этот год* отображают весь период целиком, включая часы/дни которые ещё не наступили (из будущего). *Сегодня до сих пор*, напротив, отображает только прошедшие часы.

Когда период выбран, его можно перемещать назад и вперед во времени нажимая на кнопки стрелок . Кнопка *Отдалить* позволяет увеличить период в два раза или на 50% с каждой стороны. Выполнить увеличение также возможно, если дважды нажать в графиках. Можно свернуть весь селектор выбора времени, нажав на подпись к вкладке, которая содержит строку с выбранным периодом.

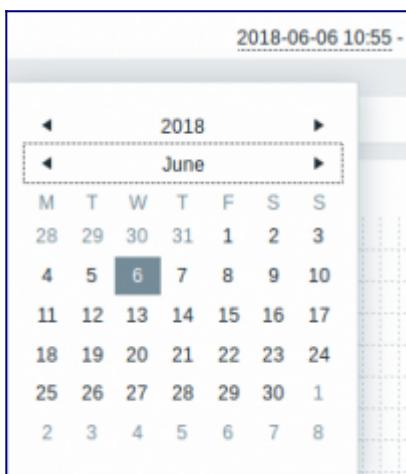
Поля *От/До* отображают выбранный период в виде:

- абсолютного времени в формате Г - М - Д Ч : м : с
- синтаксиса относительного времени, например: now - 1d

Дата в относительном формате может содержать только одну математическую операцию (- или +). В относительном времени поддерживаются следующие аббревиатуры:

- now
- s (секунды)
- m (минуты)
- h (часы)
- d (дни)
- w (недели)
- M (месяцы)
- у (года)

Имеется возможность выбрать конкретную дату начала/конца нажатием на иконке календаря после полей *От/До*. В этом случае откроется всплывающее окно выбора даты.



Внутри выбора даты имеется возможность перемещения между блоками года/месяца/даты используя Tab и Shift+Tab. Стрелки клавиатуры или кнопки стрелок позволяют выбрать желаемое значение. Нажатие Enter (или нажатие на желаемом значении) активирует выбор.

Другой путь контроля отображаемого времени - выделение области на графике левой кнопкой мыши. График будет приближен в выбранную область сразу как вы отпустите левую кнопку мыши.

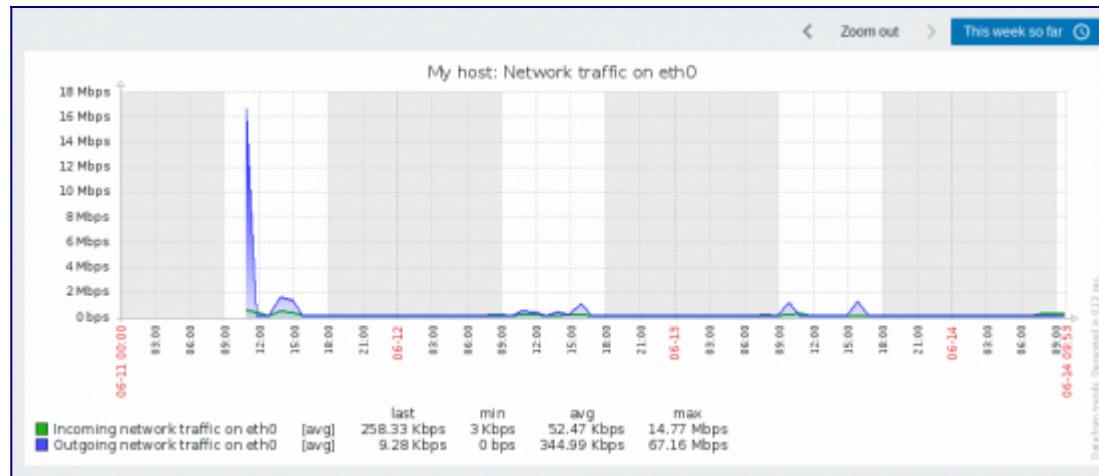
В случае, если значение времени не указано или поле пустое, тогда значение времени будет задано значением равным “00:00:00”. Такое поведение не применяется к выбору сегодняшней даты: в этом случае время будет задано текущим значением времени.

Недавние данные vs длительные периоды

Для самых последних данных **одиночная** линия рисуется с учетом каждого полученного значения. Одиночная линия рисуется до тех пор, пока доступен хотя бы один горизонтальный пиксель для одного значения.

Для данных показывающих длительный период рисуется **три линии** - темно зеленая линия отображает среднее значение, тогда как светло розовая и светло зеленая линии отображают максимальное и минимальное значения в данный момент времени. Пространство между максимумами и минимумами заполнено желтым цветом.

Рабочее время (рабочие дни) отображаются на графиках белым фоном, а нерабочее время отображается серым (при 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 отображает период более 3 месяцев.

Формирование из истории/динамики изменений

Графики могут быть отрисованы на основе [истории или динамики изменений](#) элемента данных.

У пользователей, у которых в веб-интерфейсе активирован [режим отладки](#), снизу справа от графика отображается вертикальная подпись серым цветом, которая указывает откуда пришли данные.

Два фактора, которые влияют что будет использовано история или динамика изменений:

- длительность истории элемента данных. Например, история элемента данных хранится 14 дней. В этом случае, любые данные старше четырнадцати дней будут взяты из динамики изменений.
- перегрузка данных в графике. Если отображаемое количество секунд по горизонтальным пикселям графика превышает 3600/16, то тогда будут отображены данные из динамики изменений (даже если история элемента данных еще имеется за этот период).
- если динамика изменений отключена, то для построения графика будет использоваться история элемента данных - если история доступна за требуемый период. Такое поведение поддерживается начиная с Zabbix 2.2.1 (до этого, отключенная динамика изменений означала пустой график за выбранный период, даже если история была доступна).

Переключение к сырым данным

Выпадающее меню в верхнем правом углу выше графика позволяет переключаться с простого графика на список Значения/500 последних значений. Эта функция может быть полезна для просмотра числовых значений, создающих график.

Значения представленные здесь “сырые”, то есть без единиц измерения и без постобработки значений. Однако, преобразования значений здесь также применяются.

Известные проблемы

Смотрите [известные проблемы](#) по графикам.

7.6.1.2 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 или в пределах одного шаблона.

Настройка пользовательских графиков

Для создания пользовательского графика, сделайте следующее:

- Перейдите в *Настройка* → *Узлы сети* (или *Шаблоны*)
- Нажмите на *Графики* в строке с желаемым узлом сети или шаблоном
- На странице графиков нажмите на *Создать график*
- Измените атрибуты графика

NAME	FUNCTION	DRAW STYLE	Y AXIS SIDE	COLOUR	ACTION
1: New host: Outgoing network traffic on eth0	avg	Filled region	Right	00C800	Remove
2: New host: Incoming network traffic on eth0	avg	Bold line	Right	C80000	Remove

Все обязательные поля ввода отмечены красной звёздочкой.

Атрибуты графика:

Параметр Описание

Имя

7.6.1.3 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 из 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, начиная с версии Zabbix 2.4 имеется возможность создать ситуационные графики по нескольким элементам данных на лету.

Настройка

Для создания ситуационного графика, выполните следующее:

- Перейдите в *Мониторинг* → *Последние данные*
- Используйте фильтр, чтобы отображались те элементы данных, которые вам необходимы
- Выберите те элементы данных, которые желаете увидеть на графике
- Нажмите на кнопки *Показать стекируемыхый график* или *Показать график*

The screenshot shows the 'Latest data' interface in Zabbix. At the top, there are search fields for 'Host groups' (with placeholder 'type here to search') and 'Hosts' (with 'My host' and 'Zabbix server' selected). A 'Name' field contains 'Download speed for scenario "Zabbix frontend"'. Below these are checkboxes for 'Show items without data' (checked) and 'Show details' (unchecked). Buttons for 'Apply' and 'Reset' are at the bottom of this section. The main area displays a table of items:

Host	Name	Last check	Last value	Change	Action
My host	Zabbix frontend (1 item)				
<input checked="" type="checkbox"/>	Download speed for scenario "Zabbix frontend".	2018-06-13 07:42:52	502.11 Kbps	-58.14 Kbps	Graph
Zabbix server	Zabbix frontend (1 item)				
<input checked="" type="checkbox"/>	Download speed for scenario "Zabbix frontend".	2018-06-13 07:43:01	593.04 Kbps	-54.14 Kbps	Graph

At the bottom left, there are buttons for 'Display stacked graph' and 'Display graph'. The 'Display graph' button is highlighted with a blue border.

График будет создан мгновенно:

2 items

View as Graph

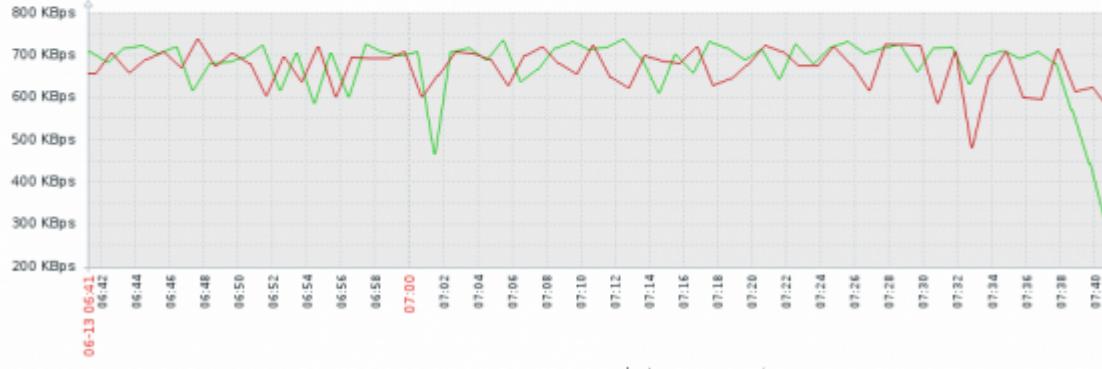
From now-1h

To now

Apply

Last 2 days Yesterday Today Last 5 minutes
Last 7 days Day before yesterday Today so far Last 15 minutes
Last 30 days This day last week This week Last 30 minutes
Last 3 months Previous week This week so far **Last 1 hour**
Last 6 months Previous month This month Last 3 hours
Last 1 year Previous year This month so far Last 6 hours
Last 2 years This year Last 12 hours
This year so far Last 1 day

Item values



My host: Download speed for scenario "Zabbix frontend". [avg] last 290.26 KBps min 290.26 KBps avg 676.92 KBps max 736 KBps
Zabbix server: Download speed for scenario "Zabbix frontend". [avg] 573.09 KBps 481.02 KBps 670.19 KBps 737.11 KBps

0.4 hours history. Generated in 0.14 sec.

Обратите внимание, что во избежании отображения слишком большого количества линий на графике, показываются только средние значения по каждому элементу данных (линии минимального и максимального значения не отображаются). Также на графике не отображаются триггеры и информация с триггеров.

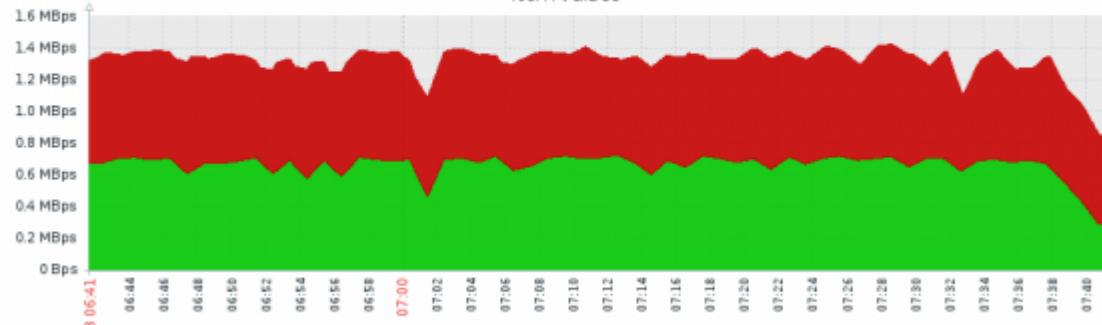
В созданном окне графика у вас имеется [выбор периода времени](#) и возможность переключения из “нормального” линейного графика на стекируемых (и обратно).

2 items

View as Graph

Graph type **Normal** Stacked

Item values



Zabbix server: Download speed for scenario "Zabbix frontend". [avg] last 573.09 KBps min 481.02 KBps avg 670.19 KBps max 737.11 KBps
My host: Download speed for scenario "Zabbix frontend". [avg] 290.26 KBps 290.26 KBps avg 676.43 KBps max 736 KBps

0.4 hours history. Generated in 0.48 sec.

7.6.2 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).

Перейдем к [настройке карты сети](#).

7.6.2.1 Настройка карты сети

Обзор

Настройка карты сети в Zabbix требует, что вы сначала создадите карту определив её общие параметры и затем начнёте заполнять карту элементами и связями между этими элементами.

Вы можете заполнять карту элементами: узлами сети, группами узлов сети, триггерами, изображениями или другими картами сети.

Для отображения элементов карты используются иконки. Вы можете указать информацию, которая будет отображена с иконками и установить, какие недавние недавние проблемы будут отображаться особым образом. Вы можете связать иконки и задать информацию, которая будет отображаться у связей.

Вы можете добавить пользовательские URL'ы, которые будут доступны при нажатии на иконках. Таким образом вы можете связать иконку узла сети со свойствами узла сети или иконку карты сети с другой картой.

Карты сети, которые уже готовы, можно просмотреть в *Мониторинг* → [Карты сети](#). На странице мониторинга вы можете нажать на иконки и воспользоваться ссылками на какие-нибудь скрипты или URL'ы.

Начиная с Zabbix 3.4, карты сети основаны на векторной графике (SVG).

Публичные и приватные карты

Все пользователи Zabbix (включая пользователей не являющихся администраторами) могут создавать карты сети. У карт сетей имеется владелец - пользователь, который создал их.

- *Публичные* карты сети видимы всем пользователям, однако, эти пользователи должны иметь как минимум права на чтение по крайней мере одного элемента карты сети, чтобы её увидеть. Публичные карты можно редактировать в случае, если пользователь/группа пользователей имеет права чтения-записи к этой карте и по крайней мере права чтения всех элементов соответствующей карты, включая триггеры в связях.
- *Приватные* карты сети видны только своим владельцам и пользователям/группам пользователей с [общим доступом](#) к этой карте сети, которым поделился владелец. Обычные пользователи (не Супер Администраторы) могут предоставлять общий доступ только тем группам и пользователям, которым они принадлежат сами. Пользователи уровня Администратора могут видеть приватные карты сети независимо от того, являются ли они владельцами или принадлежат списку пользователей с общим доступом. Приватные карты может редактировать владелец карты и пользователь/группа пользователей с правами чтения-

записи этой карты сети и по крайней мере с правами чтения всех элементов соответствующей карты, включая триггеры в связях.

Элементы карты, к которым у пользователя нет прав чтения, отображаются серой иконкой, а вся текстовая информация по элементу скрывается. Однако, подпись к триггеру видна даже, если у пользователя отсутствуют права к этому триггеру.

Чтобы добавить элемент на карту, у пользователя должны быть как минимум права на чтение этого элемента.

Создание карты сети

Для создания карты, сделайте следующее:

- Перейдите в *Мониторинг* → *Карты сети*
- Перейдите к просмотру всех карт
- Нажмите на *Создать карту сети*

Чтобы создать новую карту сети, вы также можете использовать кнопки *Клонировать* и *Полное клонирование* в диалоге конфигурации существующей карты. При нажатии на *Клонировать* будут сохранены все атрибуты макета оригинальной карты, но без элементов. При *Полном клонировании* сохраняются как общие атрибуты макета, так и все элементы оригинальной карты сети.

Вкладка **Карта сети** содержит общие атрибуты карт сетей:

The screenshot shows the 'Map' configuration dialog in Zabbix. The 'Map' tab is selected. Key configuration options include:

- Owner:** Admin [Zabbix Administrator] (with a 'Select' button)
- Name:** Local network
- Width:** 680
- Height:** 600
- Background image:** No Image
- Automatic icon mapping:** <manual> (with a 'show icon mappings' link)
- Icon highlight:** checked
- Mark elements on trigger status change:** checked
- Display problems:** Expand single problem (selected), Number of problems, Number of problems and expand most critical one
- Advanced labels:** checked
- Host group label type:** Label
- Host label type:** Label
- Trigger label type:** Status only
- Map label type:** Label
- Image label type:** Nothing
- Map element label location:** Bottom
- Problem display:** All
- Minimum severity:** Not classified (selected), Information, Warning, Average, High, Disaster
- Show suppressed problems:** unchecked
- URLs:** A table with one entry: Name: Latest data, URL: https://localhost/zabbix/latest.php, Element: Host. An 'Add' button is available below the table.

Все обязательные поля ввода отмечены красной звёздочкой.

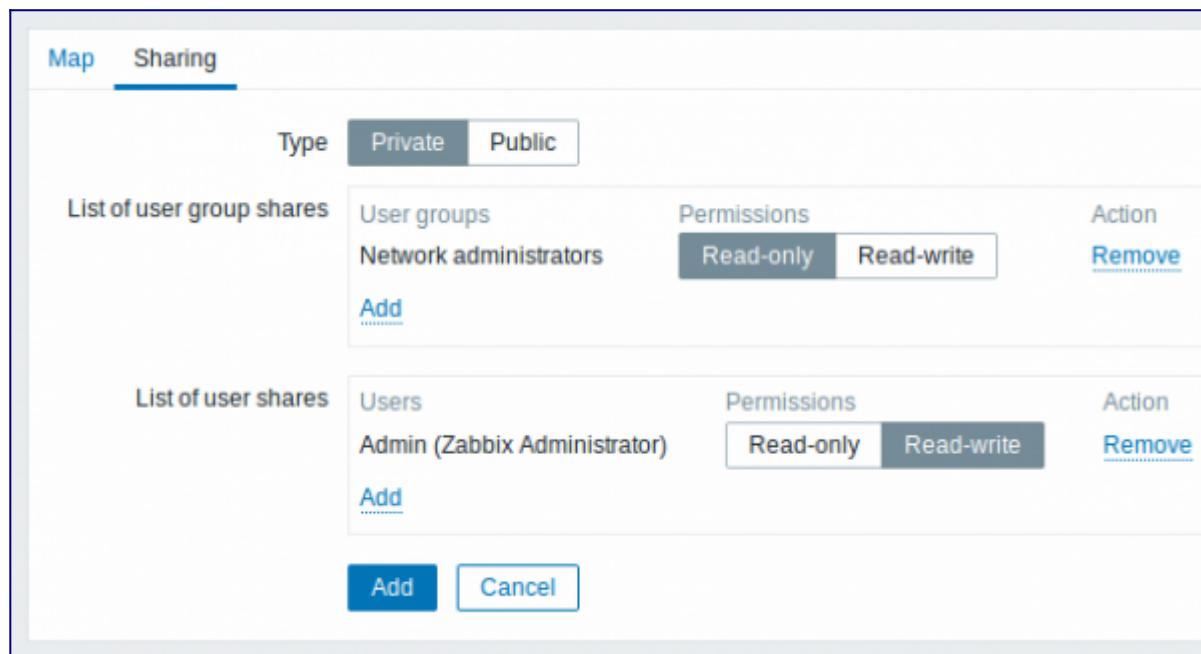
Общие атрибуты карты сети:

Параметр	Описание
Владелец	Имя владельца карты.
Имя	Уникальное имя карты.
Ширина	Ширина карты в пикселях.
Высота	Высота карты в пикселях.
Фоновое изображение	Использование фонового изображения: Нет изображения - без фонового изображения (белый фон) Изображение - выбранное изображение, которое используется как фоновое изображение. Масштабирование не производится. Вы можете использовать географическую карту или любое другое изображение для улучшения вашей карты.
Автоматическое соответствие иконок	Вы можете указать использование автоматического соответствия иконок, заданные в Администрирование → Общие → Соответствие иконок . Соответствие иконок позволяет отображать некоторые иконки в соответствии с полями инвентарных данных узлов сети.
Подсветка иконок	Если вы отметите эту опцию, элементы карты получат подсветку. Элементы с активными триггерами будут иметь фон в виде круга, такого же цвета, что и триггер с наивысшей важностью. Кроме того, будет отображаться зеленая толстая линия вокруг круга, если все проблемы подтверждены. Если элемент в состоянии “деактивирован” или “в обслуживании”, будет использоваться фон в виде квадрата, серый и оранжевый соответственно. Смотрите также: Просмотр карт сети
Помечать элементы при изменении состояния триггера	Недавнее изменение состояния триггера (недавняя проблема или её решение) будет подсвечиваться маркерами (указывающие внутрь красные треугольники) по трем сторонам иконки элемента, которые не заняты подписью. Маркеры отображаются 30 минут.
Отображение проблем	Выбор способа отображения проблемы с элементом карты: Разворачивание одиночной проблемы - если у элемента карты имеется одна проблема, отображать имя проблемы (триггера) Количество проблем - отображается общее количество проблем Количество проблем и раскрывать наиболее критичную - отображается имя проблемы (триггера) по самому критичному триггеру, а также общее количество проблем. Определение 'самого критичного' основывается на: <i>Для элемента триггера на карте</i> - важность триггера и его позиция в списке триггеров; <i>Для остальным элементов карты</i> - важность триггера и ID триггера. Если эта опция выбрана, у вас будет возможность задать тип подписи для разных типов элементов.
Расширенные подписи	Тип подписи, который будет использоваться для элементов карты: Подпись - подпись к элементу карты IP адрес - IP адрес Имя элемента - имя элемента (например, имя узла сети) Только состояние - только состояние (OK или ПРОБЛЕМА) Ничего - подписи не будут отображаться
Размещение подписи к элементам карты	Расположение подписи по отношению к элементу карты: По нижнему краю - ниже иконки По левой стороне - слева По правой стороне - справа По верхнему краю - выше элемента карты
Отображение проблем	Отображение количества проблем как: Все - будет отображено полное количество проблем

Параметр	Описание
Минимальная важность триггеров	Неподтвержденные отдельно - количество неподтвержденных проблем будет отображено отдельным числом из общего количества проблем Только неподтвержденные - будет отображено только количество неподтвержденных проблем
Отображение подавленных проблем	Проблемы с важностью ниже заданной в данной опции не будут отображены на карте. Например, с выбранной важностью <i>Предупреждение</i> , изменения состояний триггеров с <i>Информация</i> и <i>Не классифицировано</i> не будут отражены на карте. Параметр поддерживается начиная с Zabbix 2.2.
URL'ы	Отметьте опцию отображения проблем, которые в противном случае были бы подавлены (не отображены) по причине обслуживания узла сети. Можно указать URL'ы для каждого типа элемента (с подписью). Они будут отображаться как ссылки, если пользователь нажмет на элементе в режиме просмотра карты. В URL'ах карт можно использовать <u>макросы</u> : {MAP.ID}, {HOSTGROUP.ID}, {HOST.ID}, {TRIGGER.ID}

Общий доступ

Вкладка **Общий доступ** содержит как тип карты сети, так и опции общего доступа (группы пользователей, пользователи) для приватных карт сети:



Параметр	Описание
Type	Выберите тип карты: Приватная - карта видима только выбранным группам пользователей и пользователям Публичная - карта видима всем
Список групп пользователей участников	Выберите группы пользователей, которые будут иметь доступ к этой карте сети. Вы можете разрешить доступ только на чтение или чтение-запись.
Список пользователей участников	Выберите пользователей, которые будут иметь доступ к этой карте сети. Вы можете разрешить доступ только на чтение или чтение-запись.
Когда вы нажмете Добавить для сохранения карты сети, вы создадите пустую карту сети с именем, размерами и определенными настройками. Теперь вам необходимо добавить какие-нибудь элементы.	

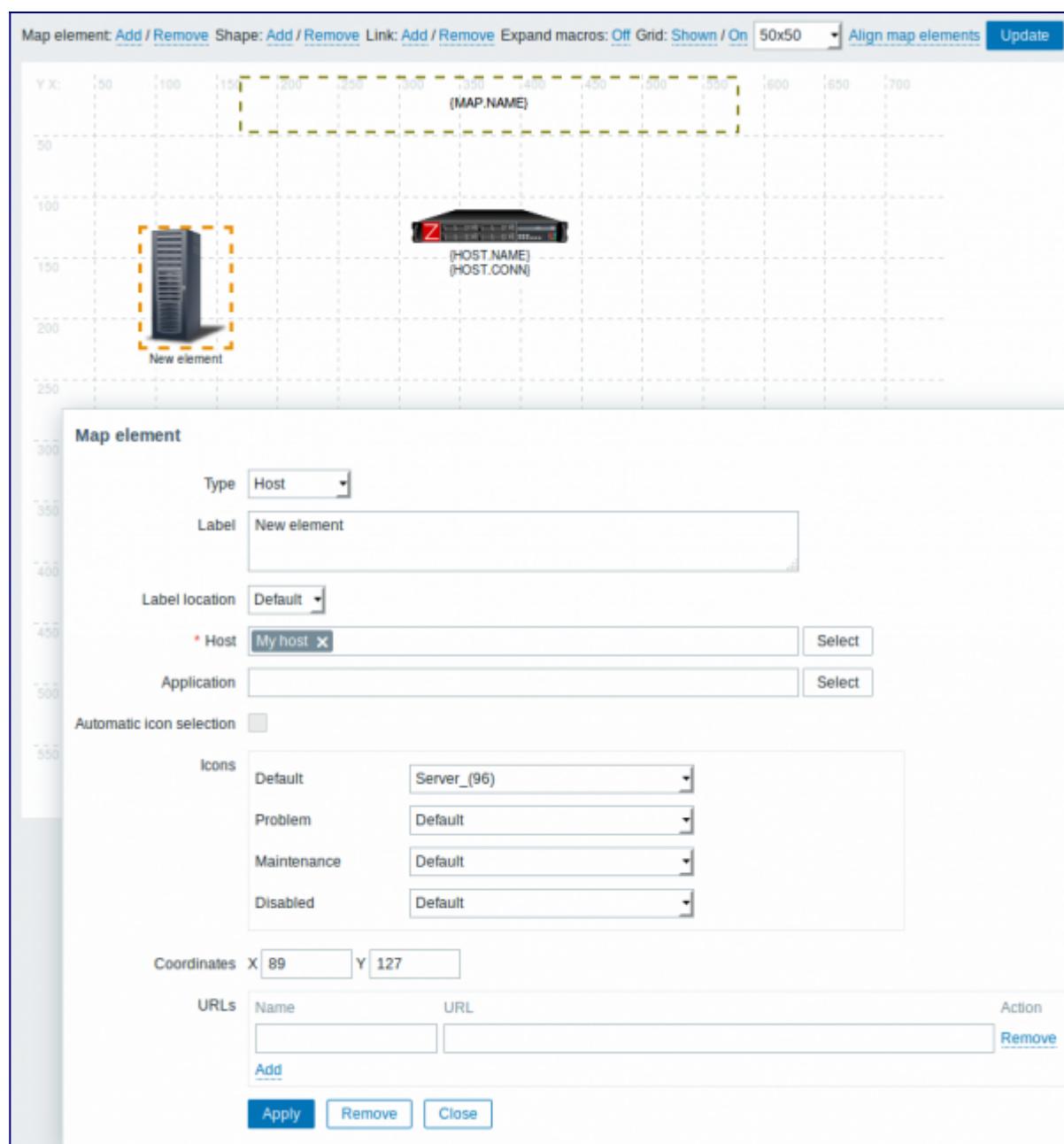
Чтобы это сделать, нажмите на Конструктор в списке карт сетей, чтобы открыть область редактирования.

Добавление элементов

Для добавления элемента, нажмите на ссылку *Добавить* следующую за подписью 'Элемент карты сети'. Новый элемент появится в левом верхнем углу карты. Возьмите и переместите куда бы вам ни захотелось.

Обратите внимание с "Вкл" опцией Сетки, элементы будут всегда выравниваться по сетке (вы можете выбрать различные размеры сетки из выпадающего меню, а также скрыть/показать сетку). Если вы хотите разместить элементы где бы то ни было без выравнивания, переключите опцию в "Выкл". (Элементы, расположенные случайным образом, можно в дальнейшем снова выровнять по сетке с помощью кнопки *Выровнять элементы карты*.)

Теперь у вас есть несколько размещенных элементов, вы можете захотеть начать различать их, дав имена и т.п. Нажав на элемент, отобразится диалог и вы сможете задать тип элемента, дать имя, выбрать другую иконку и так далее.



Атрибуты элементов карт сетей:

Параметр	Описание
<i>Тип</i>	<p>Тип элемента:</p> <p>Узел сети - иконка, отображающая состояние всех триггеров у выбранного узла сети</p> <p>Карта сети - иконка, отображающая состояние всех элементов на карте сети</p> <p>Триггер - иконка, отображающая состояние одного и более триггеров</p> <p>Группа узлов сети - иконка, отображающая состояние всех триггеров у на всех узлах сети в выбранной группе узлов сети</p> <p>Изображение - иконка, не связанная с каким-то ресурсом</p>
<i>Подпись</i>	<p>Подпись к иконке, любая строка.</p> <p>В подписях можно использовать макросы и многострочные строки.</p> <p>Расположение подписи по отношению к иконке:</p> <p>По умолчанию - расположение подписи по умолчанию карты сети</p> <p>По нижнему краю - ниже иконки</p> <p>По левой стороне - слева</p> <p>По правой стороне - справа</p> <p>По верхнему краю - выше иконки</p>
<i>Узел сети</i>	<p>Введите узел сети, если тип элемента 'Узел сети'. Это поле с функцией автодополнения, таким образом после начала ввода имени узла сети, будет автоматически предложен список совпадающих узлов сети в выпадающем меню. Прокрутите список, чтобы выбрать необходимый. Нажмите на 'x', чтобы удалить выбранный.</p>
<i>Карта сети</i>	<p>Выберите карту сети, если тип элемента 'Карта сети'.</p> <p>Если тип элемента 'Триггер', выберите один или более триггеров в поле <i>Новые триггера</i> снизу и нажмите на <i>Добавить</i>.</p> <p>Порядок выбранных триггеров можно изменить, но только в пределах одной важности триггеров. Несколько выбранных триггеров также влияют на раскрытие макросов {HOST.*} как в режиме конструктора, так и в режиме просмотра.</p> <p>1 В режиме конструктора первые отображаемые макросы {HOST.*} будут раскрыты в зависимости от первого триггера в списке (на основании важности триггера).</p> <p>2 Режим просмотра зависит от параметра Отображения проблем в общих атрибутах карты сети.</p> <ul style="list-style-type: none"> * Если выбран режим <i>Развёртывание одиночной проблемы</i>, тогда первые отображаемые макросы {HOST.*} будут раскрыты в зависимости от последней обнаруженной проблемы триггера (вне зависимости от важности) или от первого триггера в списке (в случае отсутствия обнаруженных проблем); * Если выбран режим <i>Количество проблем и раскрывать наиболее критичную</i>, тогда первые отображаемые макросы {HOST.*} будут раскрыты в зависимости от важности триггера. <p>Выберите группу узлов сети, если тип элемента 'Группа узлов сети'. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню. Прокрутите список, чтобы выбрать необходимую. Нажмите на 'x', чтобы удалить выбранную.</p> <p>Вы можете выбрать группу элементов данных, позволяющую отображать только те проблемные триггеры, которые принадлежат указанной группе элементов данных.</p> <p>Это поле доступно для типов элементов данных узла сети и группы узлов сети и поддерживается начиная с Zabbix 2.4.0.</p>
<i>Группа элементов данных</i>	<p>Иконка (по умолчанию)</p> <p>Используемая иконка.</p>
<i>Автоматический</i>	В этом случае соответствие иконок будет определять, какую иконку отображать.

Параметр	Описание
выбор иконки	
Иконки	Вы можете выбирать различные иконки для элемента для отображения этих случаях: по умолчанию, проблема, обслуживание, деактивирован.
Координата X	Координата X элемента карты сети.
Координата Y	Координата Y элемента карты сети.
URL'ы	URL'ы можно указать для каждого отдельного элемента. Они будут отображаться как ссылки, если пользователь нажмёт на элементе в разделе мониторинга. Если элемент имеет собственные URL'ы, а также URL'ы уровня карты сети для его типа элемента, то ссылки будут объединены в одно меню. Макросы , которые можно использовать в URL'ах карт сети: {MAP.ID}, {HOSTGROUP.ID}, {HOST.ID}, {TRIGGER.ID}
Добавленные элементы не сохраняются автоматически. Если вы покинете эту страницу, все изменения могут быть потеряны.	

Поэтому будет хорошей идеей нажать на кнопке **Обновить** в правом верхнем углу. При нажатии, все изменения сохраняются независимо от того что вы выберите в дальнейшем во всплывающем окне.

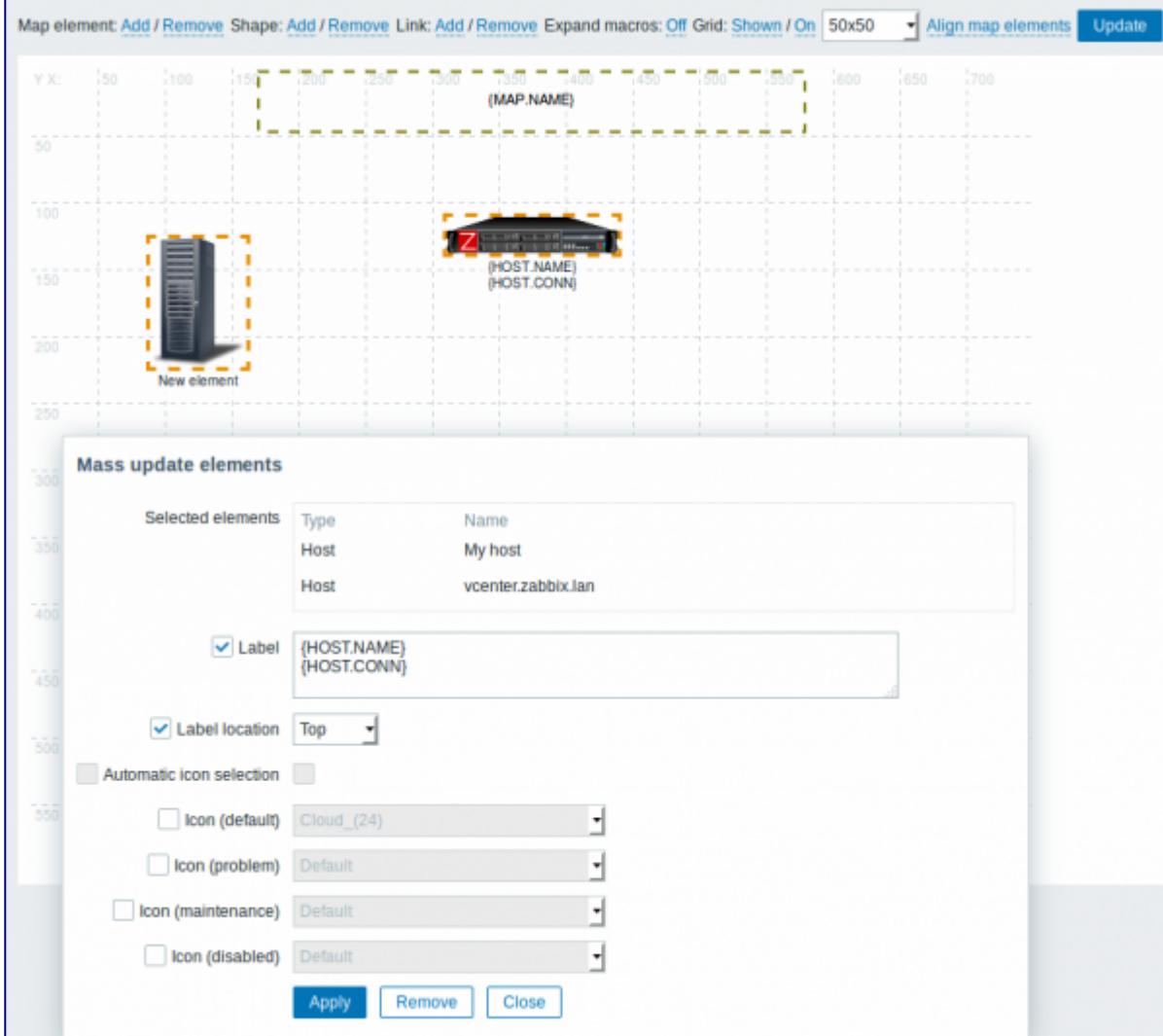
Выбранные опции сетки также сохраняются для каждой карты сети.

Выбор элементов

Для выбора нескольких элементов, выберите сначала один элемент, затем нажмите и удерживайте *Ctrl*, чтобы выбрать другие.

Вы также можете выбрать несколько элементов, переместив прямоугольник в области редактирования и тем самым выбрав все элементы находящиеся в нем (опция доступна начиная с Zabbix 2.0).

Как только вы выберите 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, {HOSTNAME} для подписи элемента).



Связи элементов

После того, как вы расставите элементы на карте, самое время соединить их. Для соединения двух элементов вы должны сначала выбрать их. Когда элементы будут выбраны, нажмите на *Добавить* справа от подписи *Связь*.

При наличии созданной связи, диалог одного элемента теперь содержит дополнительный раздел *Связь*. Нажмите на *Изменить* для редактирования атрибутов связи.

Map element: Add / Remove Shape: Add / Remove Link: Add / Remove Expand macros: Off Grid: Shown / On 50x50 Align map elements Update

The screenshot shows a network map with a host icon labeled "New element" connected to a Zabbix server icon labeled "[HOST.NAME] (HOST.CONN)". A green line with a label "100Mbps" connects them. The background map shows various network components and connections.

Map element

Type: Host

Label: New element

Label location: Default

* Host: My host

Application:

Automatic icon selection:

Icons:

Default	Server_(96)
Problem	Default
Maintenance	Default
Disabled	Default

Coordinates: X: 89 Y: 127

URLs:

Name	URL	Action
<input type="text"/>	<input type="text"/>	<input type="button" value="Remove"/>
<input type="button" value="Add"/>		
<input type="button" value="Apply"/>	<input type="button" value="Remove"/>	<input type="button" value="Close"/>

Links:

Element name	Link indicators	Action
vcenter.zabbix.lan		<input type="button" value="Edit"/>

Label: 100Mbps

Connect to: vcenter.zabbix.lan

Type (OK): Bold line

Colour (OK): #00CC00

Link indicators:

Trigger	Type	Colour	Action
<input type="button" value="Add"/>			

Атрибуты связей:

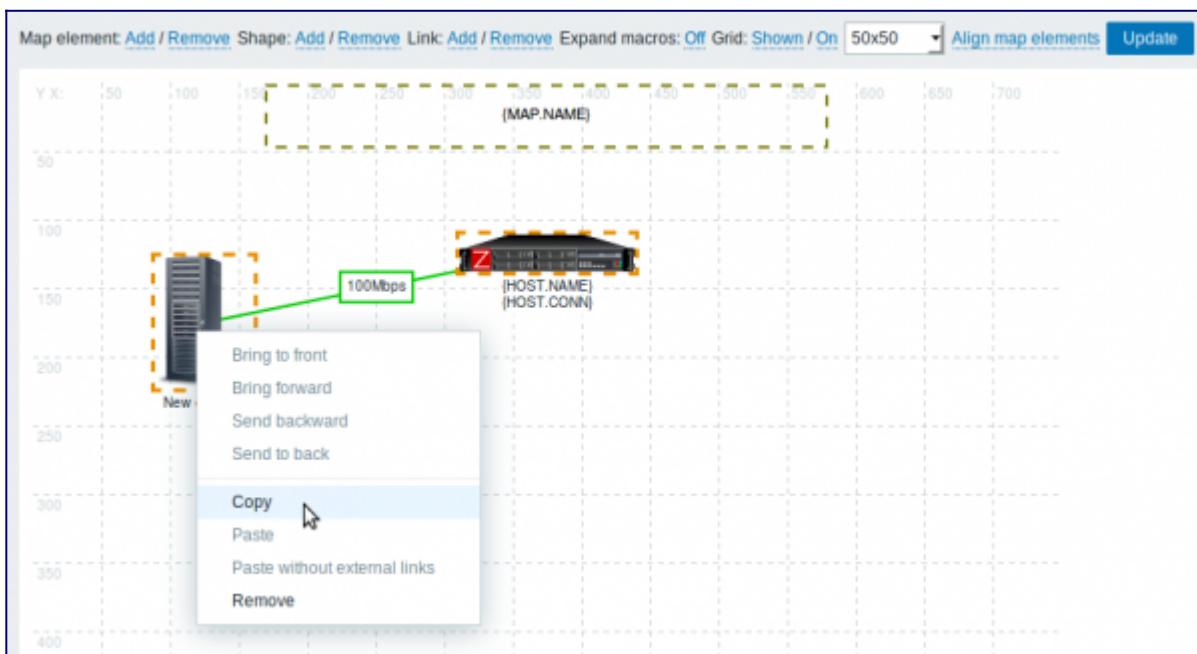
Параметр	Описание
Подпись	Подпись, которая отображается поверх связи. В этом поле поддерживается макрос <code>{узел_сети:ключ.функция(параметр)}</code> , но только функции триггеров avg, last, min и max с параметром в виде секунд.
Подключаться	Элемент, к которому присоединяется связь.

Параметр	Описание
через	Стиль связи по умолчанию: Линия - одиночная линия Жирная линия - жирная линия Точечная линия - точки Пунктирная линия - пунктирная линия
Tun (OK)	
Цвет (OK)	Цвет связи по умолчанию.
<u>Индикаторы связи</u>	Список триггеров соединенных со связью. В случае, если триггер в состоянии ПРОБЛЕМА, его стиль применяется к связи.

Перемещение и копирование-вставка элементов

Несколько элементов можно **переместить** в другое место на карте, нажав на одном из выбранных элементов и удерживая кнопку мыши нажатой переместить курсор в желаемое место.

Один или много элементов можно **скопировать** выбрав эти элементы, затем нажать на выбранном элементе правой кнопкой мыши и из меню выбрать *Копировать*.



Чтобы вставить скопированные элементы, нажмите в области карты правой кнопкой мыши и из меню выберите *Вставить*. Опция *Вставить без внешних связей* вставит элементы и оставит только те связи, которые имеются между выбранными элементами.

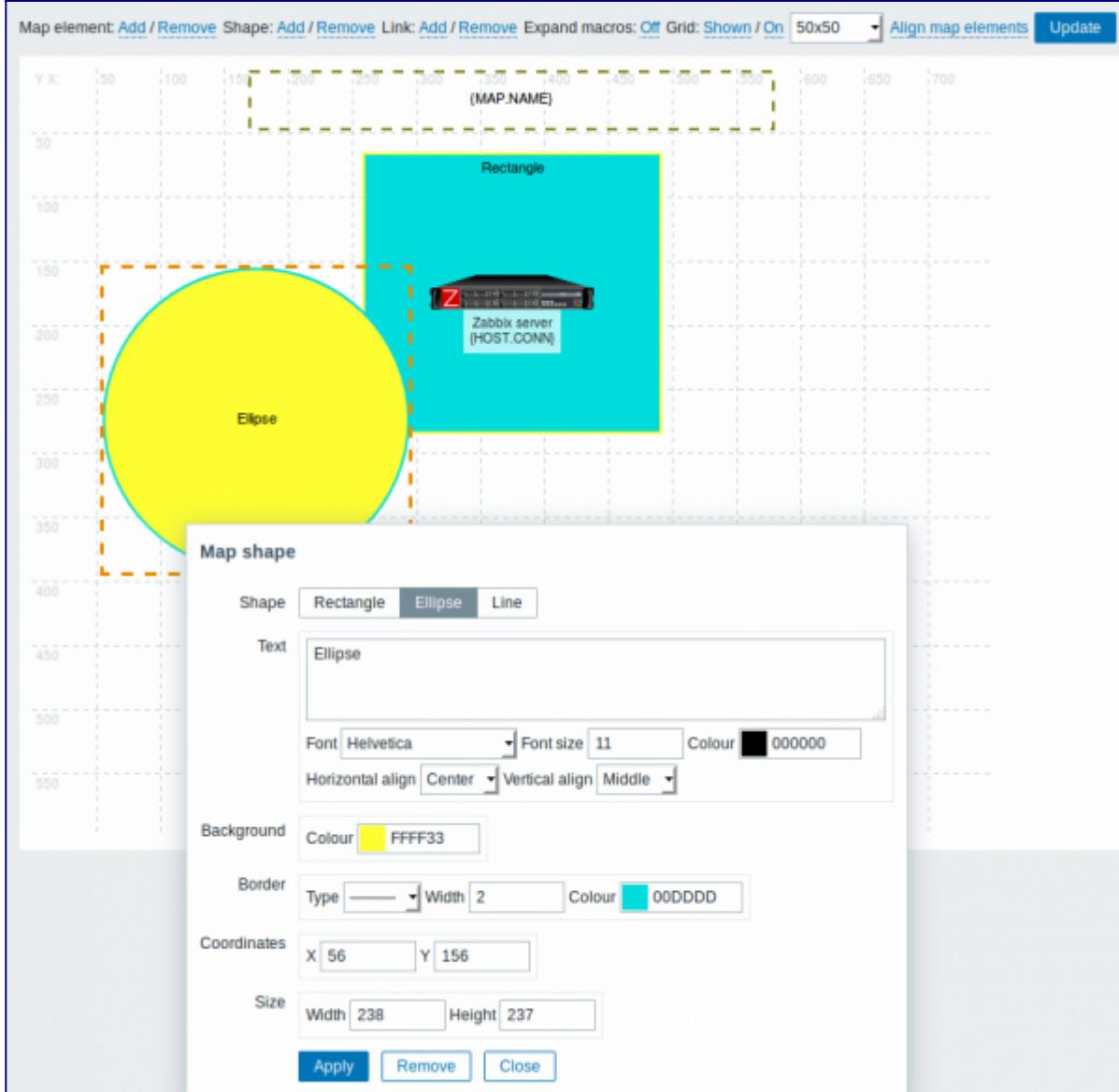
Копирование-вставка работает только в одном окне браузера. Быстрые сочетания клавиш не поддерживаются.

Добавление фигур

В дополнение к элементам карты имеется также возможность добавления некоторых фигур. Фигуры не являются элементами карты; они являются лишь визуальным представлением. Например, фигуру прямоугольника можно использовать фоном для группировки некоторых узлов сети. Можно добавить фигуры прямоугольника и эллипса.

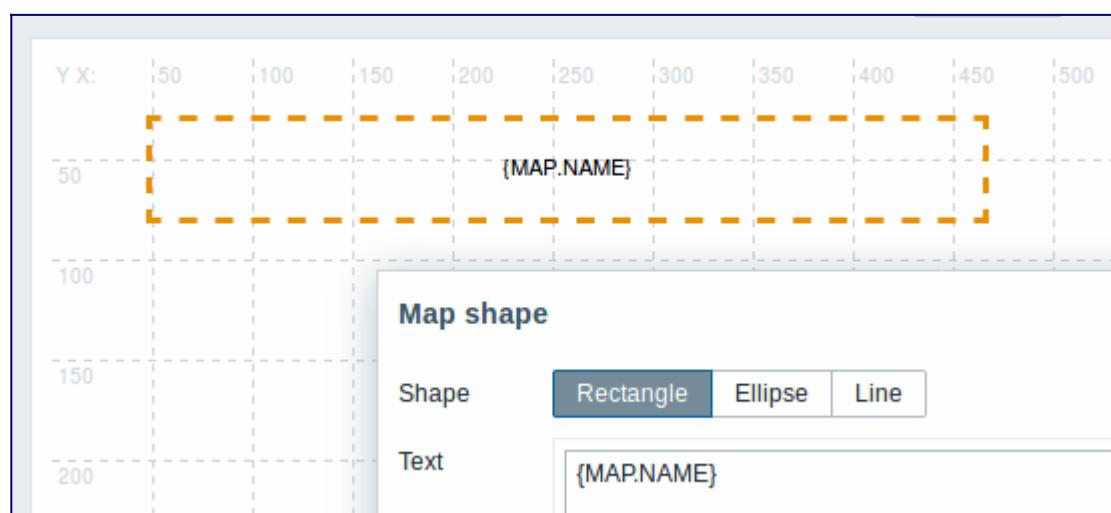
Чтобы добавить фигуру, нажмите на *Добавить* после надписи *Фигура*. Новая фигура появится в левом верхнем углу карты сети. Переместите её в любое место куда пожелаете.

Новая фигура добавляется с цветами по умолчанию. При нажатии на фигуру отобразится диалог и вы сможете изменить внешний вид фигуры, добавить текст и так далее.



Чтобы выбрать фигуры, выберите одну и затем удерживая клавишу *Ctrl* выберите остальные. Когда выбрано несколько фигур, можно будет массово обновить общие свойства, аналогично элементам карты сети.

К фигурам можно добавлять текст. Чтобы отображать только текст, фигуру можно сделать невидимой, удалив рамку фигуры (выберите 'Нет' в поле Рамка). Например, обратите внимание на то, как макрос {MAP.NAME}, отображаемый на снимке экрана выше, в действительности является фигурой прямоугольника с текстом, что можно увидеть нажав на макрос:



{MAP.NAME} раскрывается в заданное имя карты при просмотре карты сети.

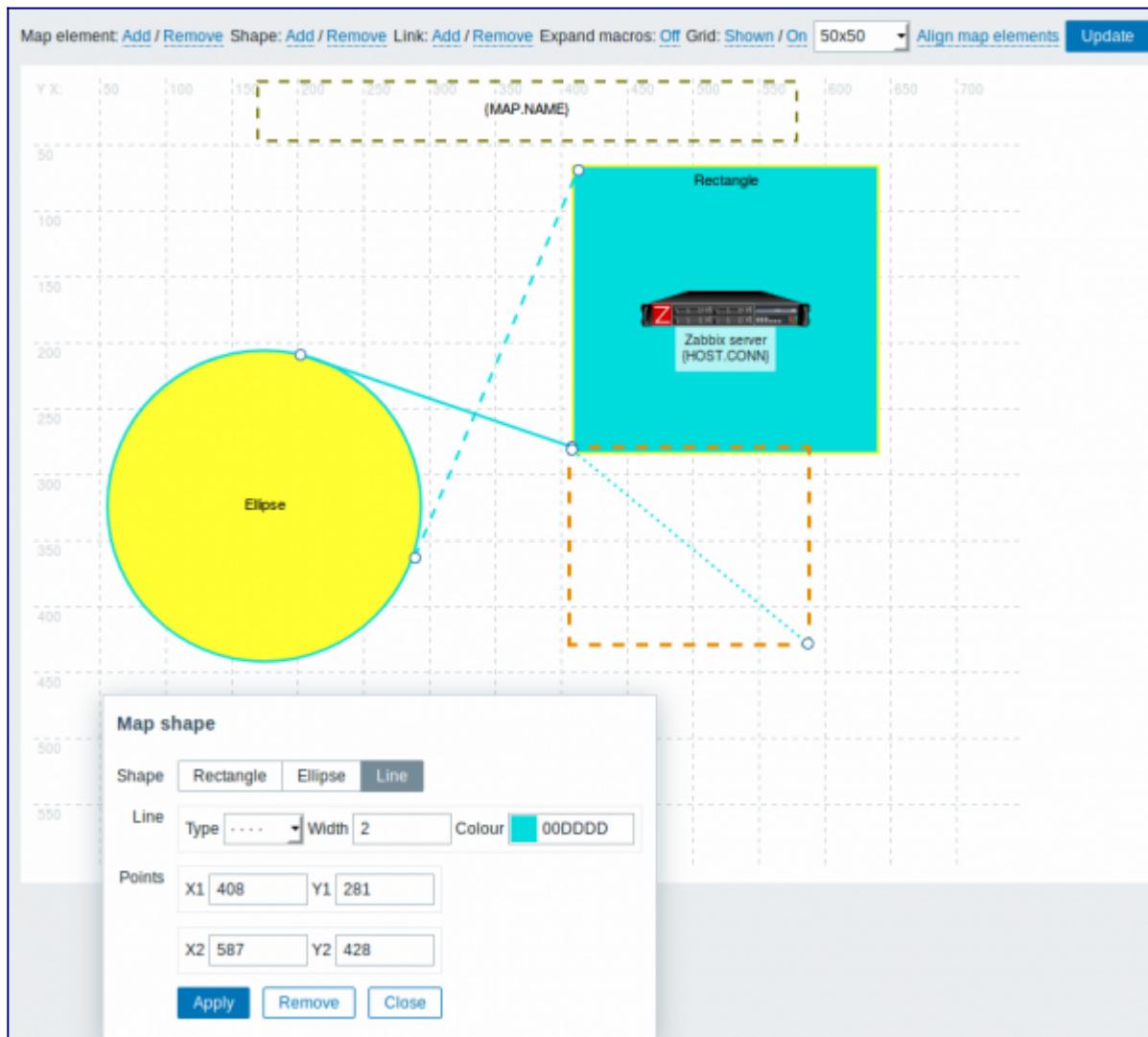
Если в тексте используются гиперссылки, они станут нажимаемыми при просмотре карты сети.

Перенос строк текста всегда “включен” в пределах фигур. Однако, в пределах эллипса строки переносятся так, как будто эллипс это прямоугольник. Перенос слов не реализован, поэтому длинные слова (слова, которые не вписываются в фигуру) не переносятся, но замаскированы (на странице конструктора) или обрезаны (другие страницы с картами).

Добавление линий

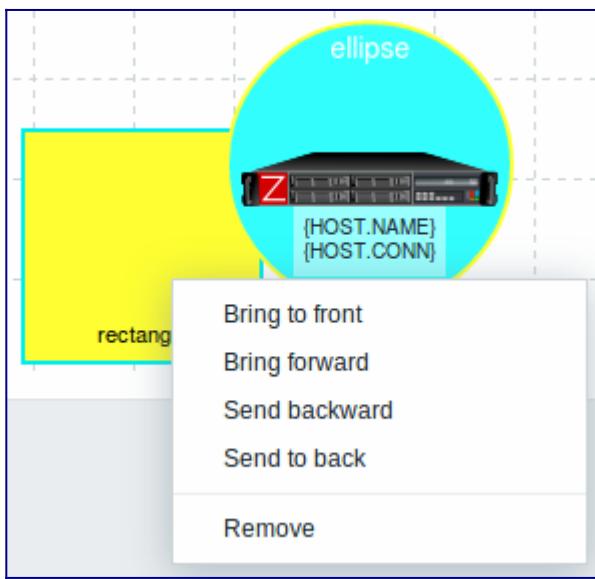
В дополнение к фигурам имеется также возможность добавления некоторых линий. Линии можно использовать для создания связей между элементами или фигурами на карте сети.

Чтобы добавить линию, нажмите на *Добавить* после надписи *Фигура*. Новая фигура появится в левом верхнем углу карты сети. Выберите её и в диалоге редактирования нажмите на *Линия*, чтобы изменить фигуру на линию. Затем настройте свойства линии, такие как тип, ширина, цвет и остальные.



Очередность фигур и линий

Чтобы поместить одну фигуру спереди от других (или наоборот), нажмите на фигуре правой кнопкой мыши и выберите необходимый пункт во всплывающем меню к фигуре.



7.6.2.2 Элементы группы узлов сети

Обзор

Этот раздел объясняет как добавить тип элемента “Группа узлов сети” при настройке [карты сети](#).

Настройка



Все обязательные поля ввода отмечены красной звёздочкой.

Эта таблица состоит из параметров, характерных для типа элемента *Группа узлов сети*:

Параметр

Описание

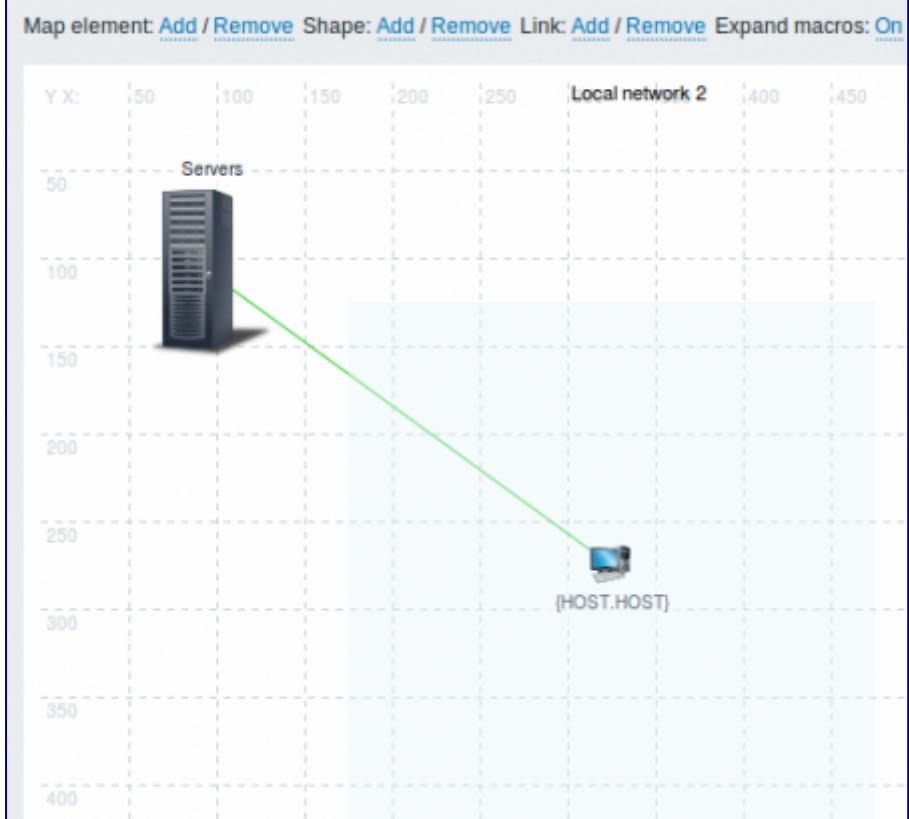
<i>Тип</i>	Выберите Тип элемента: Группа узлов сети - иконка, которая представляет состояние всех триггеров всех узлов сети, которые принадлежат выбранной группе
<i>Отображение</i>	Опции отображения: Группа узлов сети - выбор этой опции приведет к тому, что одна иконка будет отображать соответствующую информацию о конкретной группе узлов сети Элементы группы узлов сети - выбор этой опции приведет к тому, что несколько иконок будут отображать соответствующую информацию по каждому отдельному

Параметр	Описание
<i>Тип области</i>	элементу (узлу сети) из конкретной группы узлов сети Эта опция доступна, если выбран параметр “Элементы группы узлов сети”: Вписать в карту сети - все элементы группы узлов сети располагаются равномерно по всей карте сети Пользовательский размер - настройка вручную области карты сети для отображения всех элементов группы узлов сети Эта опция доступна, если выбраны параметр “Элементы группы узлов сети” и параметр “Тип области” : Ширина - числовое значение, которое необходимо ввести, чтобы указать ширину области карты Высота - числовое значение, которое необходимо ввести, чтобы указать высоту области карты
<i>Размер области</i>	Сетка – единственная доступная опция отображения всех элементов группы узлов сети
<i>Алгоритм размещения</i>	Подпись к иконке, любая строка. В подписях можно использовать макросы и многострочные строки. Если типом элемента карты сети является “Группа узлов сети”, тогда ввод определенных макросов влияет на отображение на карте сети соответствующей информации по каждомуциальному отдельному узлу сети. Например, если используется макрос {HOST.IP}, при редактировании карты сети отобразится только сам макрос {HOST.IP}, тогда как просмотр карты сети будет включать и отображать уникальный IP адрес каждого узла сети
<i>Подпись</i>	

Просмотр элементов групп узлов сети

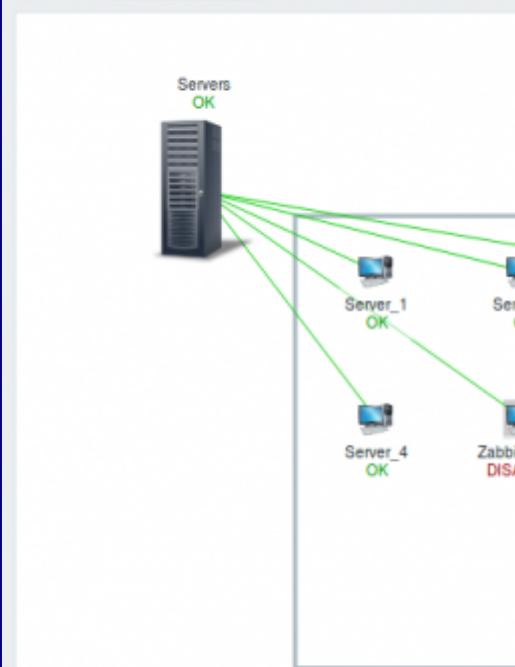
Эта опция доступно только, если выбрана опция отображения “Элементы группы узлов сети”. При выборе “Элементы группы узлов сети” опцией *отобразить*, вы сначала увидите только одну иконку для группы узлов сети. Однако, когда вы сохраните карту сети и перейдете в просмотр карты сети, вы увидите, что карта сети включает в себя все элементы (узлы сети) заданной группы узлов сети:

Network maps



Maps

All maps / Local network 2



Обратите внимание, как используется макрос {HOST.NAME}. В диалоге редактирования карты сети имя макроса не раскрывается, тогда как при просмотре карты сети отображаются все уникальные имена узлов сети.

7.6.2.3 Индикаторы связей

Обзор

Вы можете назначить несколько триггеров на [связь](#) между элементами на карте сети. Когда эти триггеры перейдут в состояние проблемы, связь сможет отразить это.

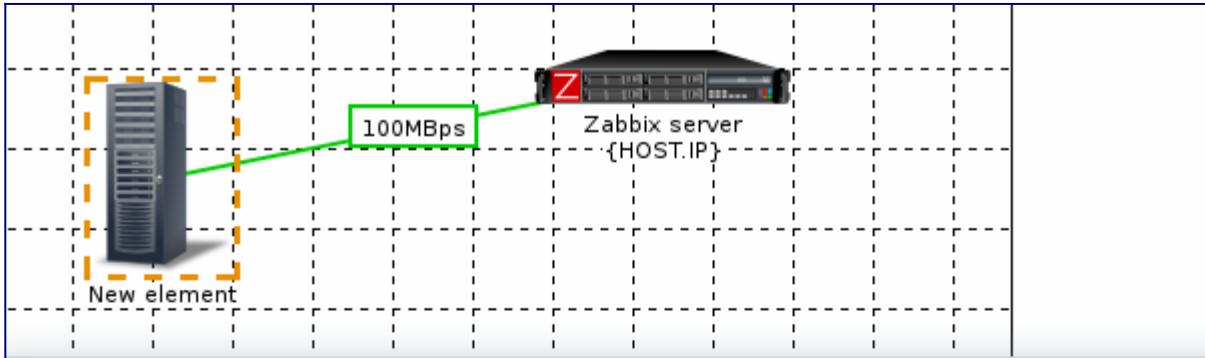
Когда вы настраиваете связь, вы указываете значения по умолчанию для типа и цвета связи. Когда вы назначаете триггеры к связи, вы можете назначить разные типы и цвета у связи с помощью этих триггеров.

Если какие-либо из этих триггеров перейдут в состояние проблема, их стиль связи и цвет будет отображен на связи. Возможно, цвет по умолчанию вашей связи был зеленой линией. Теперь, когда триггер в состоянии проблема, ваша связь стала жирной красной линией (если вы определили её так).

Настройка

Для назначения триггеров как индикаторов связи, сделайте следующее:

- выберите элемент карты
- нажмите на *Изменить* в разделе *Связи* у соответствующей связи
- нажмите на *Добавить* в блоке *Индикаторы связи* и выберите один или более триггеров



Map element

Type	Host								
Label	New element								
Label location	Default								
*Host	New host <input type="button" value="X"/> <input type="button" value="Select"/>								
Application	<input type="button" value="Select"/>								
Automatic icon selection	<input type="checkbox"/>								
Icons	<table border="1"> <tr> <td>Default</td> <td>Server_(96)</td> </tr> <tr> <td>Problem</td> <td>Server_(128)</td> </tr> <tr> <td>Maintenance</td> <td>Server_(24)</td> </tr> <tr> <td>Disabled</td> <td>Default</td> </tr> </table>	Default	Server_(96)	Problem	Server_(128)	Maintenance	Server_(24)	Disabled	Default
Default	Server_(96)								
Problem	Server_(128)								
Maintenance	Server_(24)								
Disabled	Default								
Coordinates	X <input type="text" value="89"/> Y <input type="text" value="127"/>								
URLs	<table border="1"> <thead> <tr> <th>NAME</th> <th>URL</th> </tr> </thead> <tbody> <tr> <td><input type="text"/></td> <td><input type="text"/></td> </tr> <tr> <td colspan="2">Add</td> </tr> </tbody> </table> <input type="button" value="Apply"/> <input type="button" value="Remove"/> <input type="button" value="Close"/>	NAME	URL	<input type="text"/>	<input type="text"/>	Add			
NAME	URL								
<input type="text"/>	<input type="text"/>								
Add									
Links	<table border="1"> <thead> <tr> <th>ELEMENT NAME</th> <th>LINK INDICATORS</th> </tr> </thead> <tbody> <tr> <td>Zabbix server</td> <td>New host: Zabbix agent on New host is unreachable for 5 minutes</td> </tr> </tbody> </table>	ELEMENT NAME	LINK INDICATORS	Zabbix server	New host: Zabbix agent on New host is unreachable for 5 minutes				
ELEMENT NAME	LINK INDICATORS								
Zabbix server	New host: Zabbix agent on New host is unreachable for 5 minutes								
Label	<input type="text" value="100MBps"/>								
Connect to	<input type="button" value="Zabbix server"/>								
Type (OK)	<input type="button" value="Bold line"/>								
Colour (OK)	<input type="color" value="#00CC00"/>								
Link indicators	<table border="1"> <thead> <tr> <th>TRIGGER</th> <th>TYPE</th> <th>COLO</th> </tr> </thead> <tbody> <tr> <td><input type="text"/></td> <td><input type="text"/></td> <td><input type="text"/></td> </tr> </tbody> </table>	TRIGGER	TYPE	COLO	<input type="text"/>	<input type="text"/>	<input type="text"/>		
TRIGGER	TYPE	COLO							
<input type="text"/>	<input type="text"/>	<input type="text"/>							

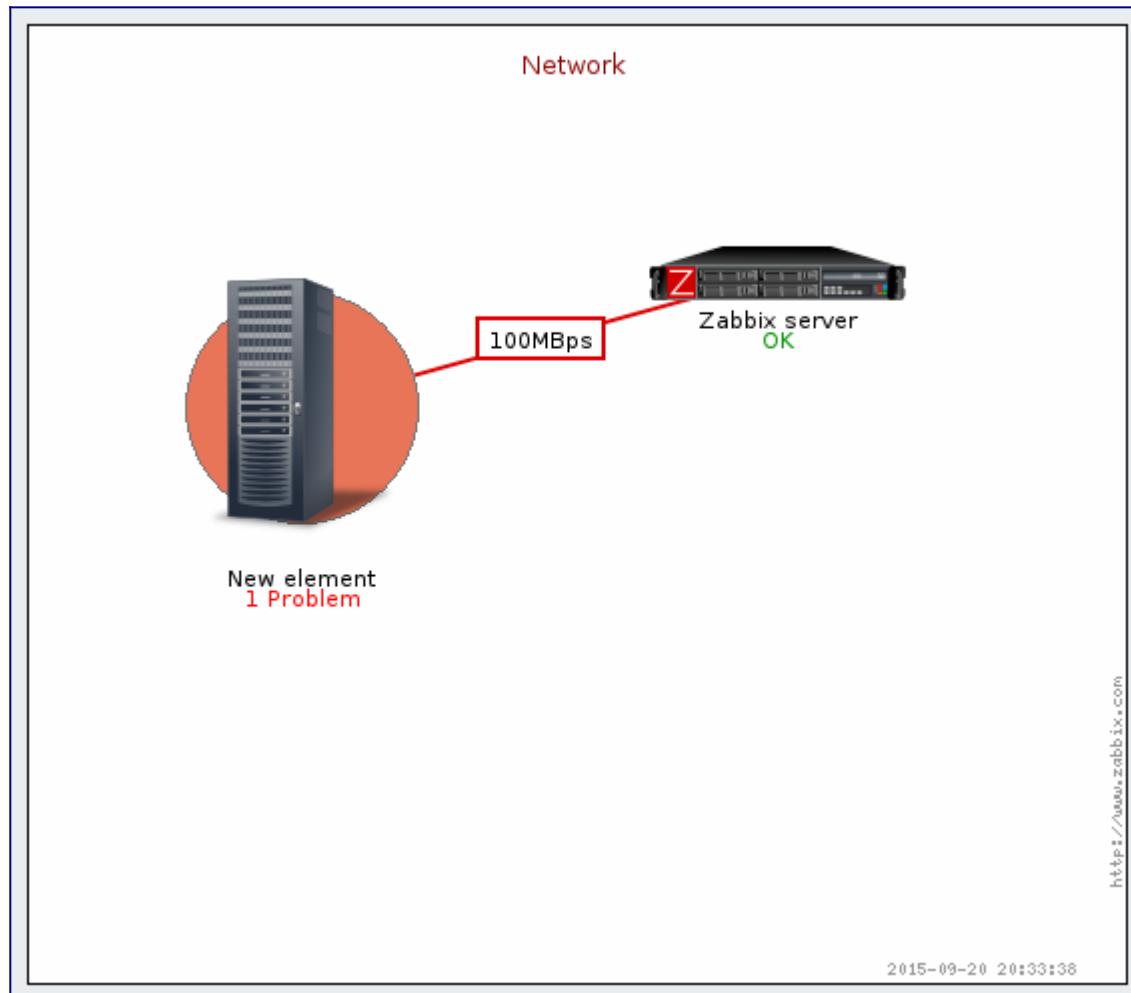
Все обязательные поля ввода отмечены красной звёздочкой.

Добавленные триггеры будут видны в списке *Индикаторы связи*.

Вы можете выбрать тип и цвет связи для каждого триггера напрямую из списка. Когда закончите, нажмите на *Применить*, закройте диалог и нажмите на *Обновить* для сохранения изменений на карте сети.

Просмотр

В *Мониторинг → Карты сети* отобразится соответствующий цвет у связи, если триггер перейдет в состояние проблема.



Если несколько триггеров перейдут в состояние проблема, то проблема с наивысшей важностью будет определять цвет и стиль связи. Если несколько триггеров с одинаковой важностью назначены на одну связь карты сети, то триггер с наименьшим ID будет иметь приоритет. Также обратите внимание что:

1. Настройки *Минимальная важность триггеров* и *Отображение подавленных проблем* из конфигурации карты сети влияют какие проблемы принимаются во внимание.
2. В случае триггеров с несколькими проблемами (множественная генерация проблем), у каждой проблемы может быть важность, которая отличается от важности триггера (изменена вручную), могут быть разные теги (из-за макросов) и может быть подавлена.

7.6.3 Комплексные экраны

Обзор

На комплексных экранах в Zabbix вы можете группировать информацию с различных источников для быстрого обзора данных на одном экране. Создание комплексных экранов очень простая и интуитивная задача.

По сути комплексный экран это таблица. Вы выбираете как много в таблице будет ячеек и какие элементы будут отображаться в этих ячейках. Следующие элементы могут быть отображены:

- простые графики
- прототипы простых графиков
- пользовательские графики
- прототипы пользовательских графиков
- карты сети
- другие комплексные экраны
- простая текстовая информация
- информация о сервере (обзор)
- информация о узлах сети (обзор)
- информация о триггере (обзор)
- события у узла сети или в группе узлов сети (состояние триггеров)
- проблемы по важности
- обзор данных
- часы
- история событий
- история недавних действий
- URL (данные полученные с другого места)

Глобальные комплексные экраны управляются в *Мониторинг* → [Комплексные экраны](#), где их можно настраивать, управлять и просматривать. Их также можно добавлять в раздел избранного на *Мониторинг* → [ПАНЕЛЬ](#).

[Комплексные экраны уровня узла сети](#) настраиваются на уровне шаблонов и затем генерируются для узлов сети, как только шаблон присоединяется к этим узлам сети.

Для настройки комплексного экрана сначала вы должны создать его, указав ему общие параметры и затем добавив к нему в ячейки индивидуальные элементы.

Все пользователи Zabbix (включая пользователей не администраторов) могут создавать комплексные экраны. Комплексные экраны имеют владельца - пользователя, который создал их.

Комплексные экраны можно сделать публичными или приватными. Публичные комплексные экраны видны всем пользователям.

Приватные комплексные экраны видимы только их владельцам. Владелец может давать общий доступ к приватным комплексным экранам другим пользователям и группам пользователей. Обычные (не Супер администраторы) пользователи могут назначать общий доступ только тем группам и пользователям, в которые они входят сами. Приватные комплексные экраны будут видны своим владельцам и пользователям с общим доступом к этому комплексному экрану так долго, пока они имеют права на чтение всех элементов комплексного экрана. Пользователи уровня Администратора, пока они имеют права на чтение всех элементов комплексного экрана, могут просматривать и редактировать приватные комплексные экраны независимо от того, являются ли они владельцами или входят ли в список пользователей общего доступа.

Как в случае с публичными, так и с приватными комплексными экранами, пользователь должен иметь по крайней мере права на чтение всех элементов комплексного экрана, чтобы его увидеть. Для добавления элемента на комплексный экран пользователь также должен иметь права по крайней мере на чтение этого элемента.

Создание комплексного экрана

Для создания комплексного экрана, сделайте следующее:

- Перейдите в *Мониторинг* → *Комплексные экраны*
- Перейдите на просмотр всех комплексных экранов
- Нажмите на *Создать комплексный экран*

Вкладка **Комплексный экран** содержит общие атрибуты комплексного экрана:

The screenshot shows the 'Complex screen' creation dialog. It has two tabs at the top: 'Screen' (selected) and 'Sharing'. The 'Screen' tab contains the following fields:

- * Owner: Admin (Zabbix Administrator) with a 'Select' button.
- * Name: Zabbix server.
- * Columns: 2.
- * Rows: 2.

At the bottom are 'Add' and 'Cancel' buttons.

Все обязательные поля ввода отмечены красной звёздочкой.

Дайте вашему комплексному экрану уникальное имя и укажите количество колонок (вертикальных ячеек) и строк (горизонтальных ячеек).

Вкладка **Общий доступ** содержит как тип комплексного экрана, так и опции общего доступа (группы пользователей, пользователи) для приватных комплексных экранов:

The screenshot shows the 'Sharing' tab of the complex screen creation dialog. It includes:

- Type: Private (selected) or Public.
- List of user group shares:

USER GROUPS	PERMISSIONS	ACTION
Zabbix administrators	Read-only	Read-write

With a 'Remove' button and an 'Add' link.
- List of user shares:

USERS	PERMISSIONS	ACTION
user (New User)	Read-only	Read-write

With a 'Remove' button and an 'Add' link.

At the bottom are 'Add' and 'Cancel' buttons.

Параметр	Описание
<i>Владелец</i>	Выберите владельца комплексного экрана.
<i>Тип</i>	Выберите тип комплексного экрана: Приватный - комплексный экран видим только выбранным группам пользователей и пользователям Публичный - комплексный экран видим всем
<i>Список групп пользователей участников</i>	Выберите группы пользователей, которые будут иметь доступ к этому комплексному экрану. Вы можете разрешить доступ только на чтение или чтение-запись.
<i>Список пользователей участников</i>	Выберите пользователей, которые будут иметь доступ к этому комплексному экрану. Вы можете разрешить доступ только на чтение или чтение-запись.
Нажмите на <i>Добавить</i> , чтобы сохранить комплексный экран.	

Добавление элементов

Для добавления элементов на комплексный экран, нажмите на *Конструктор* после имени комплексного экрана в списке.

В комплексном экране вы, возможно, увидите только ссылки с текстом *Изменить*. При нажатии на эти ссылки будет открыт диалог, в котором вы укажите, что отображать в каждой ячейке.

В существующем комплексном экране при нажатии на существующие элементы откроется диалог, в котором вы можете указать, что отображать.

Screens: Zabbix server



Все обязательные поля ввода отмечены красной звёздочкой.

Атрибуты элементов комплексных экранов:

Параметр

Описание

Отображаемая информация в ячейке:

Журнал действий - история недавних действий

Часы - цифровые или аналоговые часы, которые показывают время на сервере или местное время

Обзор данных - последние данные группы узлов сети

График - один пользовательский график

Прототип графика - пользовательский график из правила низко-уровневого обнаружения

История событий - последние события

События в группах узлов сети - состояния триггеров отфильтрованные по группе узлов сети (включая триггеры без событий, начиная с Zabbix 2.2)

Информация о узле сети - высокоуровневая информация о узле сети

События узла сети - состояния триггеров отфильтрованные по узлу сети (включая триггеры без событий, начиная с Zabbix 2.2)

Ресурс

Параметр	Описание
	<p>Карта сети - одна карта сети</p> <p>Простой текст - простые текстовые данные</p> <p>Комплексный экран - комплексный экран (один комплексный экран может содержать другие комплексные экраны)</p> <p>Простой график - один простой график</p> <p>Прототип простого графика - простой график на основе элемента данных сгенерированного правилом низко-уровневого обнаружения</p> <p>Информация о системе - высокоуровневая информация о Zabbix сервере</p> <p>Проблемы по важности - отображение проблем по важности (похоже на ПАНЕЛЬ)</p> <p>Информация о триггерах - высокоуровневая информация о триггерах</p> <p>Обзор триггеров - состояние триггеров группы узлов сети</p> <p>URL - добавление содержимого из внешнего источника</p> <p>Смотрите также более подробную информацию о настройке каждого ресурса.</p> <p>Возможные значения:</p> <ul style="list-style-type: none"> По центру По левой стороне По правой стороне <p>Возможные значения:</p> <ul style="list-style-type: none"> По середине По верхнему краю По нижнему краю
<i>Выравнивание по горизонтали</i>	
<i>Выравнивание по вертикали</i>	
<i>Объединение столбцов</i>	Расширение ячейки на указанное количество столбцов, работает так же как расширение столбцов в HTML.
<i>Объединение строк</i>	Расширение ячейки на указанное количество строк, работает так же как расширение строк в HTML.
	Обратите внимание на кнопки '+' и '-' с каждой стороны таблицы.
	При нажатии на '+' выше таблицы будет добавлена колонка. При нажатии на '-' ниже таблицы будет удалена колонка.
	При нажатии на '+' с левой стороны таблицы будет добавлена строка. При нажатии на '-' с правой стороны таблицы будет удалена строка.
	Если заданная высота графика менее 120 пикселов, то триггеры не будут отображены в легенде этого графика.
	Динамические элементы
	Для некоторых элементов имеется дополнительная опция, называемая <i>Динамический элемент</i> . При выборе этой опции сначала покажется, что ничего не изменилось.
	Однако, если вы перейдете в <i>Мониторинг</i> → <i>Комплексные экраны</i> , вы поймете, что сейчас вам доступны дополнительные выпадающие меню для выбора узла сети. Таким образом у вас есть комплексный экран в котором некоторые элементы всегда отображают одну и ту же информацию, тогда как другие отображают информацию в зависимости от выбранного в данный момент узла сети.
	Преимуществом этого метода является то, что вам не потребуется создавать дополнительные комплексные экраны только потому, что вы хотите увидеть те же самые графики содержащие данные из разных узлов сети.
	Опция <i>Динамический элемент</i> данных доступна для нескольких элементов комплексного экрана:
	<ul style="list-style-type: none"> • Графики (пользовательские графики) • Прототипы графиков

- Простые графики
- Прототипы простых графиков
- Простой текст
- URL

При нажатии на динамический график, откроется этот график в полном виде; однако, в случае с пользовательскими графиками и прототипами графиков эта опция поддерживается только для узла сети выбранного по умолчанию (то есть с 'не выбран' узлом сети из выпадающего меню). При выборе другого узла сети из выпадающего меню, динамический график создается с использованием данных элементов данных выбранного узла сети и результирующий график не является ссылкой.

Динамический элемент URL не будет отображаться в *Мониторинг → Комплексные экраны*, если не выбран узел сети. Если не выбрано ни одного узла сети, то отобразится только сообщение "Не выбран узел сети".

7.6.3.1 Элементы комплексных экранов

Обзор

Этот раздел перечисляет доступные элементы [комплексных экранов](#) и содержит подробную информацию по настройке этих элементов комплексных экранов.

7.6.3.1.1 Журнал действий

В элементе журнала действий вы можете отобразить детали операций действий (оповещения, удаленные команды). Этот элемент является копией информации с *Отчеты → Аудит*.

Для настройки, выберите ресурсом *Журнал действий*:

Resource: Action log

* Show lines: 25

Sort entries by: Time (descending)

Vertical align: Middle

* Column span: 1

* Row span: 1

Add Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Вы можете указать следующие опции:

Отображать строки Задает как много строк из журнала действий будет отображаться в ячейке комплексного экрана.

Сортировка записей по:

Сортировка записей по Время (по убыванию или по возрастанию)
Тип (по убыванию или по возрастанию)

Состояние (по убыванию или по возрастанию)

Получатель (по убыванию или по возрастанию).

7.6.3.1.2 Часы

В элементе часов вы можете отобразить локальное, серверное или время конкретного узла сети.

Для настройки, выберите ресурсом Часы:

The dialog shows settings for a 'Clock' resource:

- Resource: Clock
- Time type: Local time
- Width: 500
- Height: 100
- Horizontal align: Center (selected)
- Vertical align: Middle (selected)
- * Column span: 1
- * Row span: 1
- Add (button)
- Cancel (button)

Вы можете указать следующие опции:

Тип времени Выберите локальное, серверное или время конкретного узла сети.

Элемент данных Выберите элемент данных для отображения времени. Для отображении времени на узле сети, используется элемент данных [system.localtime\[local\]](#). Этот элемент данных должен существовать у узла сети.

Это поле доступно только при выбранном *Время узла сети*.

Ширина Выберите ширину часов.

Высота Выберите высоту часов.

7.6.3.1.3 Обзор данных

В элементе обзора данных вы можете отобразить последние данные группы узлов сети. Этот элемент является копией информации с *Мониторинг* → *Обзор* (при выбранном там Типе равном *Данные*).

Для настройки, выберите ресурсом *Обзор данных*:

The dialog shows settings for a 'Data overview' resource:

- Resource: Data overview
- * Group: Discovered hosts (with a 'Select' button)
- Application: CPU
- Hosts location: Top (selected)
- Vertical align: Middle (selected)
- * Column span: 1
- * Row span: 1
- Add (button)
- Cancel (button)

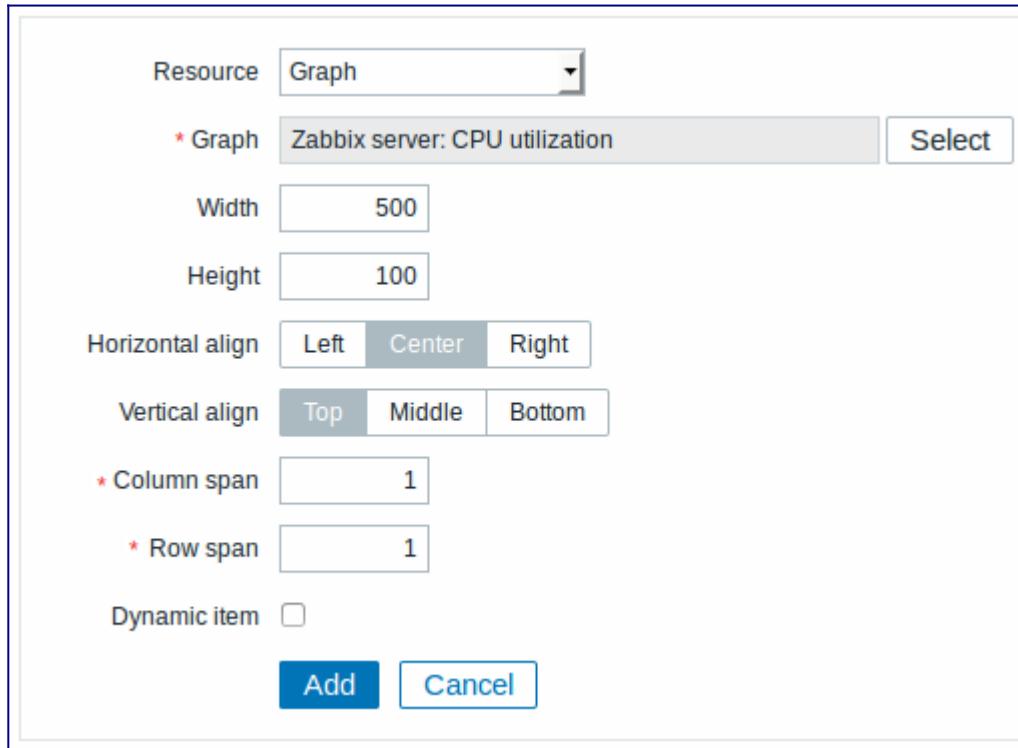
Вы можете указать следующие опции:

Группа Выберите группу узлов сети.
Группа элементов данных Введите имя группы элементов данных.
Расположение узлов сети Выберите расположение узлов сети - слева или сверху.

7.6.3.1.4 График

В элементе графика вы можете отобразить один пользовательский график.

Для настройки, выберите ресурсом *График*:



The dialog box shows the following settings:

- Resource:** Graph (selected from a dropdown menu)
- * Graph:** Zabbix server: CPU utilization (selected from a dropdown menu)
- Select:** Button to choose a graph
- Width:** 500
- Height:** 100
- Horizontal align:** Center (selected from radio buttons)
- Vertical align:** Middle (selected from radio buttons)
- * Column span:** 1
- * Row span:** 1
- Dynamic item:**
- Add:** Button to add the configuration
- Cancel:** Button to cancel the configuration

Вы можете указать следующие опции:

Имя графика Выберите отображаемый график.
Ширина Выберите ширину графика.
Высота Выберите высоту графика.
Динамический элемент Укажите графику отображать разные данные в зависимости от выбранного узла сети.

7.6.3.1.5 Прототип графика

В элементе прототипа графика вы можете отобразить пользовательский график из правила низкоуровневого обнаружения.

Для настройки, выберите ресурсом *Прототип графика*:

Resource: Graph prototype

* Graph prototype: Zabbix server: Network traffic on {#IFNAME}

* Max columns: 3

Width: 500

Height: 100

Horizontal align: Left Center Right

Vertical align: Top Middle Bottom

* Column span: 1

* Row span: 1

Dynamic item:

Вы можете указать следующие опции:

Имя графика	Выберите отображаемый прототип графика.
Макс колонок	Как много колонок со сгенерированными графиками должно отображаться в ячейке комплексного экрана.
Ширина	Полезно, когда генерируется много графиков из низкоуровневого обнаружения.
Высота	Выберите ширину графика.
Динамический элемент	Выберите высоту графика.
	Укажите графику отображать разные данные в зависимости от выбранного узла сети.

7.6.3.1.6 История событий

В элементе истории событий вы можете отобразить последние события.

Для настройки, выберите ресурсом *История событий*:

Resource: History of events

* Show lines: 25

Vertical align: Top Middle Bottom

* Column span: 1

* Row span: 1

Вы можете указать следующие опции:

Отображать строки	Укажите как много строк событий будут отображаться в ячейке комплексного экрана.
-------------------	--

7.6.3.1.7 События в группах узлов сети

В элементе событий в группе узлов сети вы можете отобразить детали проблем отфильтрованных по группе узлов сети.

Цвет важности проблемы отображается на основе цвета важности начального триггера, но его можно скорректировать в экране [обновления проблемы](#).

Для настройки, выберите ресурсом *События в группах узлов сети*:

The dialog shows settings for displaying host group issues:

- Resource: Host group issues
- Group: Linux servers (with a 'Select' button)
- * Show lines: 25
- Sort triggers by: Last change (descending)
- Vertical align: Top (selected), Middle, Bottom
- * Column span: 1
- * Row span: 1
- Add (blue button) and Cancel (white button) buttons at the bottom

Вы можете указать следующие опции:

Группа Выберите группу узлов сети.

Отображать строки Укажите как много строк проблем будут отображаться в ячейке комплексного экрана.

Сортировать триггеры по Выберите из выпадающего списка сортировку проблем по последнему изменению, важности (оба по убыванию) или узлу сети (по возрастанию).

7.6.3.1.8 Информация об узлах сети

В элементе информации об узлах сети вы можете отобразить высокоуровневую информацию по доступности узлов сети.

Для настройки, выберите ресурсом *Информация об узлах сети*:

The dialog shows settings for displaying host information:

- Resource: Host info
- Group: Linux servers (with a 'Select' button)
- Style: Horizontal (selected), Vertical
- Vertical align: Middle (selected), Top, Bottom
- * Column span: 1
- * Row span: 1
- Add (blue button) and Cancel (white button) buttons at the bottom

Вы можете указать следующие опции:

Группа Выберите группу(ы) узлов сети.

Стиль Выберите вертикальное или горизонтальное отображение.

7.6.3.1.9 События у узла сети

В элементе событий узла сети вы можете отобразить детали проблем отфильтрованных по узлу сети.

Цвет важности проблемы отображается на основе цвета важности начального триггера, но его можно скорректировать в экране [обновления проблемы](#).

Для настройки, выберите ресурсом *События у узла сети*:

The dialog shows the following settings:

- Resource:** Host issues
- Host:** New host
- Show lines:** 25
- Sort triggers by:** Last change (descending)
- Vertical align:** Top (selected), Middle, Bottom
- Column span:** 1
- Row span:** 1
- Add** (blue button)
- Cancel** (white button)

Вы можете указать следующие опции:

Узел сети

Выберите узел сети.

Отображать строки

Укажите как много строк проблем будут отображаться в ячейке комплексного экрана.

Сортировать триггеры по

Выберите из выпадающего списка сортировку проблем по последнему изменению, важности (оба по убыванию) или узлу сети (по возрастанию).

7.6.3.1.10 Карта сети

В элементе карты сети вы можете отобразить настроенную карту сети.

Для настройки, выберите ресурсом *Карта сети*:

The dialog shows the following settings:

- Resource:** Map
- * Map:** Network
- Horizontal align:** Right (selected), Left, Center
- Vertical align:** Top, Middle, Bottom
- * Column span:** 1
- * Row span:** 1
- Add** (blue button)
- Cancel** (white button)

Вы можете указать следующие опции:

Карта сети Выберите отображаемую карту.

7.6.3.11 Простой текст

В элементе простого текста вы можете отобразить последние данные по элементу данных как простой текст.

Для настройки, выберите ресурсом *Простой текст*:

The screenshot shows the configuration dialog for a 'Plain text' resource. It includes fields for selecting an item ('Item' set to 'New host: Checksum of /etc/passwd'), specifying the number of lines to show ('Show lines' set to 25), and options for vertical alignment ('Vertical align' set to 'Top') and spans ('Column span' and 'Row span' both set to 1). There is also a checkbox for 'Dynamic item'. At the bottom are 'Add' and 'Cancel' buttons.

Вы можете указать следующие опции:

Элемент данных

Выберите элемент данных.

Отображать строки

Укажите как много последних данных будут отображаться в ячейке комплексного экрана.

Показать текст в виде HTML

Задайте отображение текста HTML кодом.

Динамический элемент

Укажите отображение разных данных в зависимости от выбранного узла сети.

7.6.3.12 Комплексный экран

В элементе комплексного экрана вы можете отобразить другой комплексный экран Zabbix. Один комплексный экран может содержать в себе другие комплексные экраны.

Для настройки, выберите ресурсом *Комплексный экран*:

Resource Screen

* Screen Zabbix server2 Select

Vertical align Top Middle Bottom

* Column span 1

* Row span 1

Add Cancel

Вы можете указать следующие опции:

Комплексный экран Выберите отображаемый комплексный экран.

7.6.3.1.13 Простой график

В элементе простого графика вы можете отобразить один простой график.

Для настройки, выберите ресурсом *Простой график*:

Resource Simple graph

* Item New host: Incoming network traffic on eth0 Select

Width 500

Height 100

Horizontal align Left Center Right

Vertical align Top Middle Bottom

* Column span 1

* Row span 1

Dynamic item

Add Cancel

Вы можете указать следующие опции:

Элемент данных

Выберите элемент данных для простого графика.

Ширина

Выберите ширину графика.

Высота

Выберите высоту графика.

Динамический элемент

Укажите графику отображать разные данные в зависимости от выбранного узла сети.

7.6.3.1.14 Прототип простого графика

В элементе прототипа простого графика вы можете отобразить простой график на основе элемента данных сгенерированного низкоуровневым обнаружением.

Для настройки, выберите ресурсом *Прототип простого графика*:

Resource Simple graph prototype

* Item prototype New host: Incoming network traffic on {#IFNAME}

* Max columns 3

Width 500

Height 100

Horizontal align Left Center Right

Vertical align Top Middle Bottom

* Column span 1

* Row span 1

Dynamic item

Вы можете указать следующие опции:

Прототип элемента данных

Выберите прототип элемента данных для простого графика.

Макс колонок

Как много колонок со сгенерированными графиками должно отображаться в ячейке комплексного экрана.

Полезно, когда генерируется много графиков из низкоуровневого обнаружения.

Ширина

Выберите ширину графика.

Высота

Выберите высоту графика.

Динамический элемент

Укажите графику отображать разные данные в зависимости от выбранного узла сети.

7.6.3.1.15 Информация о системе

В элементе информации о системе вы можете отобразить высокоуровневую информацию о Zabbix и Zabbix сервере.

Для настройки, выберите ресурсом *Информация о системе*:

Resource System information

Vertical align Top Middle Bottom

* Column span 1

* Row span 1

7.6.3.1.16 Проблемы по важности

В этом элементе вы можете отобразить проблемы по важности аналогично виджету на панели.

Для настройки, выберите ресурсом *Проблемы по важности*:

Resource	Problems by severity
Vertical align	<input checked="" type="radio"/> Top <input type="radio"/> Middle <input type="radio"/> Bottom
* Column span	<input type="text" value="1"/>
* Row span	<input type="text" value="1"/>
<input type="button" value="Add"/> <input type="button" value="Cancel"/>	

7.6.3.1.17 Информация о триггерах

В элементе информации о триггерах вы можете отобразить высокоуровневую информацию о статусах триггеров.

Для настройки, выберите ресурсом *Информация о триггерах*:

Resource	Trigger info
Group	<input type="text" value="Linux servers"/> <input type="button" value="Select"/>
Style	<input checked="" type="radio"/> Horizontal <input type="radio"/> Vertical
Vertical align	<input checked="" type="radio"/> Top <input type="radio"/> Middle <input type="radio"/> Bottom
* Column span	<input type="text" value="1"/>
* Row span	<input type="text" value="1"/>
<input type="button" value="Add"/> <input type="button" value="Cancel"/>	

Вы можете указать следующие опции:

Группа Выберите группу(ы) узлов сети.

Стиль Выберите вертикальное или горизонтальное отображение.

7.6.3.1.18 Обзор триггеров

В элементе обзора триггеров вы можете отобразить состояния триггеров из группы узлов сети. Этот элемент является копией информации с *Мониторинг → Обзор* (при выбранном там Типе равном *Триггеры*).

Для настройки, выберите ресурсом *Обзор триггеров*:

Resource	Trigger overview	<input type="button" value="▼"/>	
* Group	Linux servers <input type="button" value="X"/>	<input type="button" value="Select"/>	
Application	<input type="text"/>		
Hosts location	Left <input type="radio"/>	Top <input checked="" type="radio"/>	
Vertical align	Top <input type="radio"/>	Middle <input checked="" type="radio"/>	Bottom <input type="radio"/>
* Column span	<input type="text" value="1"/>		
* Row span	<input type="text" value="1"/>		
<input type="button" value="Add"/> <input type="button" value="Cancel"/>			

Вы можете указать следующие опции:

Группа Выберите группу(ы) узлов сети.

Группа элементов данных Введите имя группы элементов данных.

Расположение узлов сети Выберите расположение узлов сети - слева или сверху.

7.6.3.1.19 URL

В элементе URL вы можете отобразить содержимое URL из внешнего источника.

Для настройки, выберите ресурсом *URL*:

Resource	URL	<input type="button" value="▼"/>	
* URL	<input type="text"/>		
Width	<input type="text" value="500"/>		
Height	<input type="text" value="100"/>		
Horizontal align	Left <input type="radio"/>	Center <input checked="" type="radio"/>	Right <input type="radio"/>
Vertical align	Top <input type="radio"/>	Middle <input checked="" type="radio"/>	Bottom <input type="radio"/>
* Column span	<input type="text" value="1"/>		
* Row span	<input type="text" value="1"/>		
Dynamic item	<input type="checkbox"/>		
<input type="button" value="Add"/> <input type="button" value="Cancel"/>			

Вы можете указать следующие опции:

URL Введите отображаемый URL.

Ширина Выберите ширину окна.

Высота Выберите высоту окна.

Динамический элемент Укажите отображение разного содержимого URL в зависимости от выбранного узла сети.

Браузеры возможно не станут загружать HTTP страницу, которая включена в комплексный экран (при использовании элемента URL), если веб-интерфейс Zabbix доступен через протокол HTTPS.

7.6.4 Слайд-шоу

Обзор

В слайд-шоу вы можете указать какие [комплексных экранов](#) будет отображаться один за другим через заданные интервалы.

Иногда вы можете хотеть переключаться между некоторыми настроенными комплексными экранами. Хотя это можно сделать и вручную, но сделав это более, чем один или два раза, процесс может стать весьма утомительным занятием. Вот здесь функция слайд-шоу приходит к спасению.

Все пользователи Zabbix (включая пользователей не администраторов) могут создавать слайд-шоу. Слайд-шоу имеют владельца - пользователя, который создал их.

Слайд-шоу можно сделать публичными или приватными. Публичные слайд-шоу видимы всем пользователям, однако, они должны иметь по крайней мере права на чтение всех элементов слайд-шоу, чтобы его увидеть. Для добавления элемента на слайд-шоу пользователь также должен иметь права по крайней мере на чтение этого элемента.

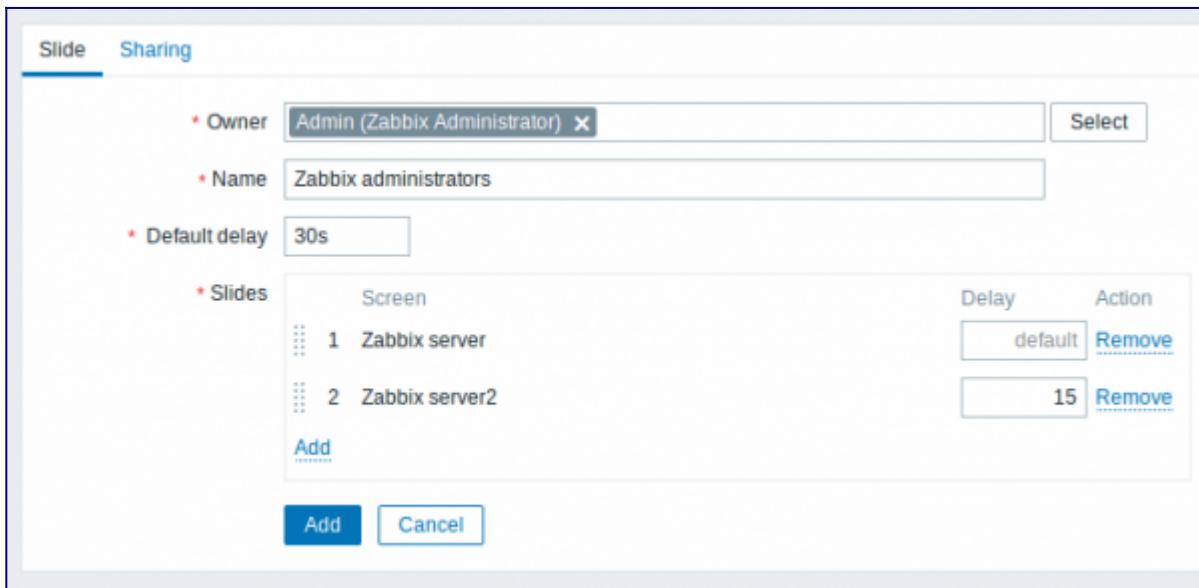
Приватные слайд-шоу видимы только их владельцам. Владелец может давать общий доступ к приватным слайд-шоу другим пользователям и группам пользователей. Обычные (не Супер администраторы) пользователи могут назначать общий доступ только тем группам и пользователям, в которые они входят сами. Приватные слайд-шоу будут видны своим владельцам и пользователям с общим доступом к этому слайд-шоу так долго, пока они имеют права на чтение всех элементов слайд-шоу. Пользователи уровня Администратора, пока они имеют права на чтение всех добавленных комплексных экранов, могут просматривать и редактировать приватные слайд-шоу независимо от того, являются ли они владельцами или входят ли в список пользователей общего доступа.

Настройка

Для создания слайд-шоу сделайте следующее:

- Перейдите в *Мониторинг* → *Комплексные экраны*
- Выберите *Слайд-шоу* в выпадающем меню
- Перейдите к просмотру всех слайд-шоу
- Нажмите на *Создать слайд-шоу*

Вкладка **Слайд** содержит общие атрибуты слайд-шоу:



Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Описание

<i>Владелец</i>	Выберите владельца слайд-шоу. Указывать владельца обязательно.
<i>Имя</i>	Уникальное имя слайд-шоу.
<i>Задержка по умолчанию</i>	Как долго будет отображаться каждый комплексный экран по умолчанию, до перехода к следующему. Поддерживаются суффиксы времени , например 30s, 5m, 2h, 1d.
<i>Слайды</i>	Список комплексных экранов, которые ротируются. Нажмите на <i>Добавить</i> , чтобы выбрать комплексные экраны. Стрелки <i>Вверх/Вниз</i> до имени комплексного экрана позволяют перетаскивать комплексный экран выше и ниже для порядка сортировки просмотра. Если вы хотите отображать только, скажем, один график в слайд-шоу, создайте комплексный экран который содержит только один график.
<i>Комплексный экран</i>	Имя комплексного экрана.
<i>Задержка</i>	Пользовательское значение как долго комплексный экран будет отображаться, в секундах. Если задано значение 0, то будет использоваться значение из <i>Задержка по умолчанию (в сек)</i> .
<i>Действие</i>	Нажмите на <i>Удалить</i> для удаления комплексного экрана из слайд-шоу.

Слайд-шоу из этого примера состоит из двух комплексных экранов, которые отображаются в следующем порядке:

Zabbix сервер ⇒ Отображается 30 секунд ⇒ Zabbix сервер2 ⇒ Отображается 15 секунд ⇒ Zabbix сервер ⇒ Отображается 30 секунд ⇒ Zabbix сервер2 ⇒ ...

Вкладка **Общий доступ** содержит как тип слайд-шоу, так и опции общего доступа (группы пользователей, пользователи) для приватных слайд-шоу:

Параметр

Tip

Список групп пользователей участников

Список пользователей участников

Нажмите на *Добавить*, чтобы сохранить слайд-шоу.

Описание

Выберите тип слайд-шоу:

Приватный - слайд-шоу видимо только выбранным группам пользователей и пользователем

Публичный - слайд-шоу видимо всем

Выберите группы пользователей, которые будут иметь доступ к этому слайд-шоу.

Вы можете разрешить доступ только на чтение или чтение-запись.

Выберите пользователей, которые будут иметь доступ к этому слайд-шоу.

Вы можете разрешить доступ только на чтение или чтение-запись.

Просмотр

Готовые слайд-шоу можно просмотреть в *Мониторинг* → *Комплексные экраны*, затем выберите *Слайд-шоу* из выпадающего списка и нажмите на имя слайд-шоу.

Выбрав нужный пункт Меню из выпадающего списка, вы можете ускорить или замедлить отображение, указав множитель задержки обновления:

Если задержка менее 5 секунд (либо была указана задержка менее 5 секунд или она стала очень маленькой из-за множителя задержки обновления), тогда будет использоваться минимальная задержка в 5 секунд.

7.6.5 Комплексные экраны узлов сети

Обзор

Комплексные экраны узлов сети выглядят также как и [глобальные комплексные экраны](#), однако, комплексные экраны узлов сети отображают данные только о конкретном узле сети. Комплексные экраны узлов сети настраиваются на уровне [шаблона](#) и затем генерируются для узлов сети, как только шаблон присоединяется к этим узлам сети.

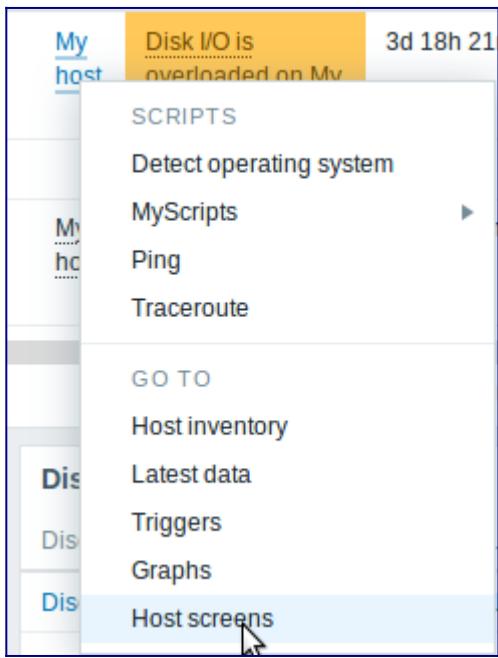
Комплексные экраны узлов сети нельзя настраивать или напрямую обращаться через раздел [Мониторинг → Комплексные экраны](#), который предназначается для глобальных комплексных экранов. Способы доступа к комплексным экранам узлов сети перечислены в этом разделе ниже.



Доступ к комплексным экранам узлов сети

Доступ к комплексным экранам возможен:

- Из [меню узлов сети](#), которое доступно во многих разделах веб-интерфейса:
 - нажмите на имени узла сети и затем выберите *Комплексные экраны узла сети* из всплывающего меню



- При поиске имени узла сети в [глобальном поиске](#):
 - нажмите на ссылку *Комплексные экраны*, указанную в результатах поиска
- При нажатии на имя узла сети в *Инвентаризация* → [Узлы сети](#):
 - нажмите на указанную ссылку *Комплексные экраны*

7.7 Шаблоны

Обзор

Шаблоном является набор объектов, который можно удобно и просто применить к нескольким узлам сети.

Объектами могут быть:

- элементы данных
- триггеры
- графики
- группы элементов данных
- комплексные экраны (*начиная с Zabbix 2.0*)
- правила низкоуровневого обнаружения (*начиная с Zabbix 2.0*)
- веб-сценарии (*начиная с Zabbix 2.2*)

Так как многие узлы сети в реальной жизни идентичны или более менее похожи, отсюда вытекает, что наборы объектов (элементы данных, триггеры, графики,...), созданные вами для одного узла сети, могут быть полезны для многих узлов сети. Конечно, вы можете скопировать их в каждый новый узел сети, но это потребовало бы много ручной работы. Вместо этого, используя шаблоны вы можете скопировать эти объекты в один шаблон и затем применить шаблон для любого количества требуемых узлов сети.

Когда шаблон присоединён к узлу сети, все объекты (элементы данных, триггеры, графики,...) из шаблона добавляются к этому узлу сети. Шаблоны соединяются непосредственно с каждым узлом сети (не к группе узлов сети).

Шаблоны обычно используются для группировки объектов конкретных сервисов или приложений (таких как Apache, MySQL, PostgreSQL, Postfix...) и затем применяются к узлам сети на которых эти сервисы запущены.

Еще одно преимущество в использовании шаблонов, это когда что-либо потребуется изменить на всех узлах сети. Изменение чего-либо на уровне шаблона распространяется на все присоединенные к этому шаблону узлы сети.

Таким образом, использование шаблонов является отличным способом снижения своей нагрузки и рационализации настройки Zabbix.

Перейдем к [созданию и настройке шаблона](#).

7.7.1 Настройка шаблона

Обзор

Для настройки шаблона необходимо сначала создать его, указав общие параметры, и только затем добавлять объекты (элементы данных, триггеры, графики и прочее) к этому шаблону.

Создание шаблона

Для создания шаблона, выполните следующее:

- Перейдите в *Настройка* → *Шаблоны*
- Нажмите на *Создать шаблон*
- Измените атрибуты шаблона

Вкладка **Шаблон** содержит общие атрибуты шаблона.

The screenshot shows the 'Template' creation dialog. At the top, there are three tabs: 'Template' (which is selected and highlighted in blue), 'Linked templates', and 'Macros'. Below the tabs, there are several input fields:

- 'Template name': A text input field containing 'Template OS Linux'.
- 'Visible name': An empty text input field.
- 'Groups': A dropdown menu currently showing 'Templates' with a delete icon, and a 'Select' button to its right.
- 'Description': A large text area for entering a description, which is currently empty.

At the bottom of the dialog are two buttons: a blue 'Add' button and a light blue 'Cancel' button.

Все обязательные поля ввода отмечены красной звёздочкой.

Атрибуты шаблонов:

Параметр	Описание
Имя шаблона	Уникальное имя шаблона.
Видимое имя	Если вы укажите это имя, то только оно будет видимо в списках, картах и прочем.
Группы	Группы узлов сети/шаблонов к которым принадлежит этот шаблон.
Описание	Введите описание шаблона.
Вкладка Соединенные шаблоны	позволяет вам присоединить один и более “вложенных” шаблонов к этому шаблону. Все объекты (элементы данных, триггеры, графики и прочее) будут унаследованы из присоединенных шаблонов.

Для присоединения нового шаблона, начните печатать в поле *Присоединить новые шаблоны* до появления списка шаблонов соответствующих введенной букве (буквам). Прокрутите список и выберите. Когда все шаблоны будут присоединены, нажмите на *Добавить*.

Для отсоединения шаблона, воспользуйтесь одной из двух опций в блоке *Присоединенных шаблонов*:

- *Отсоединить* - отсоединить шаблон, но оставить его элементы данных, триггеры и графики
- *Отсоединить и очистить* - отсоединить шаблон и удалить все его элементы данных, триггеры и графики

Вкладка **Макросы** позволяет вам определить пользовательские макросы уровня шаблона. Вы также можете здесь просмотреть макросы из присоединенных шаблонов и глобальные макросы, если вы выберите опцию *Унаследованные и макросы из шаблонов*. Это то место, где отображаются все определенные пользовательские макросы для этого шаблона со своими раскрытыми значениями, а также информация о том откуда эти макросы.

MACRO	EFFECTIVE VALUE	TEMPLATE VALUE	GLOBAL VALUE
{\$NESTED_TEMPLATE_MACRO}	→ 53689	Change ← Template2: "53689"	
{\$SNMP_COMMUNITY}	→ public	Change	← "public"
{\$TEMPLATE_MACRO}	→ 25864	Remove	

Для удобства имеются ссылки на настройку соответствующих шаблонов и глобальных макросов. Также имеется возможность изменить макрос уровня шаблона/глобальный на уровне этого шаблона, фактически создав копию этого макроса у шаблона.

Кнопки:

Add

Добавление шаблона. Добавленный шаблон должен появиться в списке.

Update

Обновление свойств существующего шаблона.

Clone

Создание другого шаблона основанного на свойствах текущего шаблона, включая все объекты (элементы данных, триггеры и прочее) унаследованные от присоединенных шаблонов.

Full clone

Создание другого шаблона основанного на свойствах текущего шаблона, включая все объекты (элементы данных, триггеры и прочее) как унаследованные от присоединенных шаблонов, так и напрямую присоединенные к текущему шаблону.

Delete

Удаление шаблона; объекты из шаблона (элементы данных, триггеры и прочее) останутся присоединенными к узлам сети.

Delete and clear

Удаление шаблона и всех его объектов из присоединенных узлов сети.

Cancel

Отмена изменения свойств шаблона.

Когда шаблон создан, самое время добавить в него какие-нибудь объекты.



Элементы данных должны быть добавлены первыми в шаблон. Триггеры и графики нельзя добавить без наличия соответствующих элементов данных.

Добавление элементов данных, триггеров, графиков

Для добавления элементов данных в шаблон, сделайте следующее:

- Перейдите в *Настройка* → *Узлы сети* (или *Шаблоны*)
- Нажмите на *Элементы данных* в строке с требуемым узлом сети/шаблоном
- Отметьте элементы данных которые вы хотите добавить в шаблон
- Выберите *Копировать выбранное в...* ниже списка
- Выберите шаблон (или группу шаблонов) в который необходимо скопировать элементы данных и нажмите на *Копировать*

Все выбранные элементы данных должны будут скопироваться в шаблон.

Добавление триггеров и графиков осуществляется похожим образом (из списка триггеров и графиков соответственно), опять же, имейте ввиду, что они могут быть добавлены только после того, как сначала будут добавлены требуемые элементы данных.

Добавление комплексных экранов

Для добавления комплексных экранов в шаблон из *Настройка* → *Шаблоны*, сделайте следующее:

- Нажмите на *Комплексные экраны* в строке с шаблоном
- Настройте комплексный экран обычным способом [настройки комплексных экранов](#)



Элементы, которые можно добавить в комплексные экраны шаблонов: простые графики, пользовательские графики, часы, простой текст, URL.



Для получения информации касательно доступа к комплексным экранам узлов сети, которые создаются с шаблонных комплексных экранов,смотрите раздел [комплексный экран узла сети](#).

Настройка правил низкоуровневого обнаружения

Смотрите раздел по [низкоуровневому обнаружению](#) в этом руководстве.

Добавление веб-сценариев

Чтобы добавить веб-сценарии в шаблон в *Настройка* → *Шаблоны*, сделайте следующее:

- Нажмите на *Веб* в строке с шаблоном
- Настройте веб-сценарии обычным способом [настройки веб-сценариев](#)

7.7.2 Присоединение/отсоединение

Обзор

Присоединением называется процесс при котором шаблоны применяются к узлам сети, тогда как отсоединение удаляет связь шаблона с узлом сети.



Шаблоны присоединяются непосредственно к каждому узлу сети, не к группе узлов сети. Простое добавление шаблона в группу узлов сети не присоединит узлы сети с ним. Группы узлов сети используются только для логической группировки узлов сети и шаблонов.

Присоединение шаблона

Для присоединения шаблона к узлу сети, сделайте следующее:

- Перейдите в *Настройка* → *Узлы сети*
- Нажмите на требуемом узле сети и переключитесь на вкладку *Шаблоны*
- Нажмите на *Добавить после Присоединить новые шаблоны*
- Выберите один или несколько шаблонов из всплывающего окна

- Нажмите на Сохранить в форме атрибутов узла сети

Узел сети теперь будет иметь все объекты (элементы данных, триггеры, графики, графики и т.д.) из шаблона.



Присоединение нескольких шаблонов к одному узлу сети завершится ошибкой, если в этих шаблонах будут элементы данных с одинаковыми ключами. И, так как триггеры и графики используют элементы данных, их нельзя присоединить к одному узлу сети несколькими шаблонами в случае, если используются одинаковые ключи элементов данных.

Когда объекты (элементы данных, триггеры, графики и прочее) добавлены из шаблона:

- ранее существовавшие идентичные объекты у узла сети будут обновлены объектами из шаблона
- объекты из шаблона будут добавлены
- любые напрямую присоединенные объекты, которые существовали до присоединения шаблона, останутся нетронутыми

В списках, все объекты из шаблона теперь имеют префикс имени шаблона, который указывает на принадлежность этих объектов соответствующему шаблону. Само название шаблона (серым текстом) является ссылкой дающей доступ к списку этих объектов на уровне шаблона.

Если какой-то объект (элемент данных, триггер, график и прочее) не имеет префикса имени шаблона, то это означает, что этот объект существовал у узла сети до добавления шаблона.

Критерий уникальности объекта

При добавлении объектов (элементов данных, триггеров, графиков и прочего) из шаблона очень важно знать какие из этих объектов уже существуют у узла сети и должны быть обновлены и какие объекты различаются. Критерием 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
- для групп элементов данных - имя группы элементов данных

Присоединение шаблонов к несколькими узлами сети

Для обновления соединения шаблона с несколькими узлами сети, в *Настройка* → *Узлы сети* выберите несколько узлов сети, отметив их, затем выберите **Массовое обновление** ниже списка и на вкладке *Шаблоны* выберите присоединение дополнительных шаблонов:

The screenshot shows a software interface for managing network templates. At the top, there are tabs: Host, Templates (which is selected), IPMI, Inventory, and Encryption. Below the tabs, a section titled 'Link templates' has a checked checkbox labeled 'Link templates'. A search bar contains the text 'Template SNMP Device'. Below the search bar are two checkboxes: 'Replace' (checked) and 'Clear when unlinking' (unchecked). At the bottom of the dialog are two buttons: 'Update' (blue) and 'Cancel' (white).

Поле *Присоединить шаблоны* имеет функцию автодополнения, начните вводить название шаблона до появления выпадающего списка совпадающих шаблонов. Просто прокрутите список, чтобы выбрать интересующие шаблоны.

Опция *Заменить* позволяет отсоединить ранее присоединенные шаблоны, перед тем как присоединить новые. Опция *Очистить при отсоединении* позволяет не только отсоединить ранее присоединенные шаблоны, но и так же удалить все элементы унаследованные из этих шаблонов (элементы данных, триггеры и прочее).



Zabbix предлагает значительный набор предустановленных шаблонов. Вы можете использовать их для присоединения, но будьте осторожны с их использованием без предварительных изменений в рабочей инсталляции, поскольку они могут содержать очень много элементов данных и слишком часто опрашивать данные. Если вы желаете использовать их, шаблонам потребуется оптимизация, для того чтобы они действительно помогли вам.

Изменение соединенных объектов

Если вы попытаетесь изменить элемент данных или триггер, который присоединен из шаблона, вы можете увидеть, что большинство ключевых опций отключены для редактирования. Такой подход имеет смысл, так как идея шаблонов - в изменении объектов за одно касание на уровне шаблона. Тем не менее, вы все еще можете, например, активировать/деактивировать элемент данных у отдельных узла 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 формы), но имейте ввиду, что все эти изменения будут влиять на все узлы сети, которые соединены с этим шаблоном.

Отсоединение шаблона

Для отсоединения шаблона от узла сети, сделайте следующее:

- Перейдите в *Настройка* → *Узлы сети*
- Нажмите на требуемый узел сети и переключитесь на вкладку *Шаблоны*
- Нажмите на *Отсоединить* или на *Отсоединить и очистить* для отсоединения шаблона
- Нажмите на *Обновить* в диалоге атрибутов узла сети

Опция *Отсоединить* просто удалит связь с шаблоном, но оставит все его объекты (элементы данных, триггеры, графики и прочее) у узла сети.

Опция *Отсоединить и очистить* удалит и связь с шаблоном, и все его объекты (элементы данных, триггеры, графики и прочее).

7.7.3 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 соединить несколько шаблонов вместе, в одном “вложенном” шаблоне.

Преимущества наследования в том, что вы должны присоединить только один шаблон к узлу сети и узел сети унаследует все объекты из присоединенных шаблонов автоматически.

Настройка наследуемого шаблона

Если вы хотите соединить несколько шаблонов, то для начала вы должны взять существующий шаблон или какой-нибудь новый, затем:

- Откройте диалог свойств шаблона
- Найдите вкладку *Соединенные шаблоны*
- Нажмите на *Добавить*, чтобы выбрать шаблоны для присоединения из всплывающего окна
- Нажмите на *Добавить* для добавления выбранных шаблонов в список
- Нажмите на *Добавить/Обновить* в диалоге свойств шаблона

Теперь шаблон должен содержать все объекты (элементы данных, триггеры, пользовательские графики и прочее) из присоединенных шаблонов.

Для отсоединения любого из присоединенных шаблонов, воспользуйтесь в этой же форме кнопками *Отсоединить* или *Отсоединить и очистить* и нажмите *Обновить*.

Опция *Отсоединить* просто удалит связь с шаблоном, но оставит все его объекты (элементы данных, триггеры, графики и прочее).

Опции *Отсоединить и очистить* удалит и связь с шаблоном, и все его объекты (элементы данных, триггеры, графики и прочее).

7.8 Готовые шаблоны

Обзор

Zabbix стремится поставлять растущий список полезных готовых [шаблонов](#). Готовые шаблоны заранее настроены и, таким образом, являются полезным способом ускорения развертывания задач мониторинга.

7.8.1 Стандартизованные шаблоны для сетевых устройств

Обзор

Чтобы предоставлять мониторинг сетевых устройств, таких как коммутаторы и роутеры, мы создали две так называемые модели: для самого сетевого устройства (в основном его шасси) и для сетевого интерфейса.

Начиная с Zabbix 3.4, поставляются шаблоны для многих семейств сетевых устройств. Все шаблоны охватывают (где возможно получить эти элементы данных с устройства):

- Мониторинг ошибок шасси (источники питания, вентиляторы и температура, общее состояние)
- Мониторинг производительности шасси (элементы данных CPU и памяти)
- Сбор инвентарных данных шасси (серийные номера, имя модели, версия прошивки)
- Мониторинг сетевых интерфейсов при помощи IF-MIB и EtherLike-MIB (состояние интерфейса, загрузка трафика на порту, состояние дуплекса для Ethernet)

Эти шаблоны доступны:

- В *Настройка → Шаблоны* в новых инсталляциях;
- В [официальном Zabbix репозитории шаблонов](#). Если вы обновляетесь с версии pre-3.4 Zabbix, вы можете импортировать эти шаблоны из XML.

Если вы импортируете новые готовые шаблоны, вы возможно захотите обновить глобальное регулярное выражение `@Network interfaces for discovery` на:

Результат ложь: ^Software Loopback Interface
 Результат ложь: ^^(In)?[lL]oop[bB]ack[0-9._]*\$
 Результат ложь: ^NULL[0-9.]*\$
 Результат ложь: ^[lL]o[0-9.]*\$
 Результат ложь: ^[sS]ystem\$
 Результат ложь: ^Nu[0-9.]*\$

для фильтрации loopback и нулевых интерфейсов с большинства систем.

Устройства

Список семейств устройств, для которых доступны шаблоны:

Имя шаблона	Производитель	Семейство устройств	Известные модели	ОС	Использованные MIB
<i>Template Net Alcatel Timetra TiMOS SNMPv2</i>					
<i>Template Net Brocade FC SNMPv2</i>	Brocade	Коммутаторы Brocade FC	Brocade 300 SAN Switch-	-	SW-MIB, ENTITY-MIB
<i>Template Net Brocade_Foudry Stackable SNMPv2</i>	Brocade	Brocade ICX	Brocade ICX6610, Brocade ICX7250-48, Brocade ICX7450-48F	-	FOUNDRY-SN-AGENT-MIB, FOUNDRY-STACKING-MIB
<i>Template Net Brocade_Foudry Nonstackable SNMPv2</i>	Brocade, Foundry	Brocade MLX, Foundry	Brocade MLXe, Foundry FLS648, Foundry FWSX424	-	FOUNDRY-SN-AGENT-MIB
<i>Template Net Cisco IOS SNMPv2</i>	Cisco	Cisco IOS версии > 12.2 3.5	Cisco C2950	IOS	CISCO-PROCESS-MIB, CISCO-POOL-MIB, CISCO-ENVMON-MIB
<i>Template Net Cisco releases later than 12.0_3_T and prior to 12.2_3.5_ SNMPv2</i>	Cisco	Cisco IOS > 12.0 3 T и < 12.2 3.5	-	IOS	CISCO-PROCESS-MIB, CISCO-POOL-MIB, CISCO-ENVMON-MIB
<i>Template Net Cisco releases prior to 12.0_3_T SNMPv2</i>	Cisco	Cisco IOS < 12.0 3 T	-	IOS	OLD-CISCO-CPU-MIB, CISCO-POOL-MIB
<i>Template Net D-Link DES_DGS Switch SNMPv2</i>	D-Link	Коммутаторы DES/DGX	D-Link DES-xxxx/DGS-xxxx, DLINK DGS-3420-26SC	-	DLINK-AGENT-MIB, EQUIPMENT-MIB, ENTITY-MIB

Имя шаблона	Производитель	Семейство устройств	Известные модели	ОС	Использованные MIB
<i>Template Net D-Link DES 7200</i>	D-Link	DES-7xxx	D-Link DES 7206	-	ENTITY-MIB,MY-SYSTEM-MIB,PROCESS-MIB,MY-MEMORY-MIB
<i>SNMPv2</i>					
<i>Template Net Dell Force S-Series</i>	Dell	Dell Force S-Series	S4810		F10-S-SERIES-CHASSIS-MIB
<i>SNMPv2</i>					
<i>Template Net Extreme Exos SNMPv2</i>	Extreme	Extreme EXOS X670V-48x	EXOS		EXTREME-SYSTEM-MIB,EXTREME-SOFTWARE-MONITOR-MIB
<i>Template Net Huawei VRP SNMPv2</i>	Huawei	Huawei VRP	S2352P-EI	-	ENTITY-MIB,HUAWEI-ENTITY-MIB
<i>Template Net Intel_Qlogic Infiniband SNMPv2</i>	Intel/QLogic	Устройства Intel/QLogic Infiniband	Infiniband 12300		ICS-CHASSIS-MIB
<i>Template Net Juniper SNMPv2</i>	Juniper	Модели MX,SRX,EX	Juniper MX240, Juniper EX4200-24F	JunOS	JUNIPER-MIB
<i>Template Net Mellanox SNMPv2</i>	Mellanox	Устройства Mellanox Infiniband	SX1036	MLNX-OS	HOST-RESOURCES-MIB,ENTIT MIB,ENTITY-SENSOR-MIB,ME MIB
<i>Template Net Mikrotik SNMPv2</i>	Mikrotik	Mikrotik RouterOS devices	Mikrotik 912UAG-5HPnD, Mikrotik 941-2nD, Mikrotik 951G-2HnD, Mikrotik 1100AHx2	RouterOS	MIKROTIK-MIB,HOST-RESOU
<i>Template Net QTech QSW SNMPv2</i>	QTech	Устройства Qtech	Qtech QSW-2800-28T	-	QTECH-MIB,ENTITY-MIB
<i>Template Net Ubiquiti AirOS SNMPv1</i>	Ubiquiti	Беспроводные устройства Ubiquiti AirOS	NanoBridge, NanoStation, Unifi	AirOS	FROGFOOT-RESOURCES-MIB,IEEE802dot11-MIB
<i>Template Net HP Comware HH3C SNMPv2</i>	HP	HP (H3C) Comware	HP A5500-24G-4SFP HI Switch		HH3C-ENTITY-EXT-MIB,ENTIT
<i>Template Net HP Enterprise</i>	HP	HP Enterprise Switch	HP ProCurve J4900B Switch 2626,		STATISTICS-MIB,NETSWITCH-ICF-CHASSIS,ENTITY-MIB,SEM

Имя шаблона	Производитель	Семейство устройств	Известные модели	ОС	Использованные MIB
Switch SNMPv2			HP J9728A 2920-48G Switch		
Template Net TP-LINK SNMPv2	TP-LINK	TP-LINK	T2600G-28TS v2.0		TPLINK-SYSMONITOR-MIB, TPLINK-SYSINFO-MIB
Template Net Netgear Fastpath SNMPv2	Netgear	Netgear Fastpath	M5300-28G		FASTPATH-SWITCHING-MIB, E1000-BOXSERVICES-PRIVATE-MIB

Дизайн шаблонов

Шаблоны разработаны с учетом следующего:

- Пользовательские макросы используются где только возможно, поэтому пользователь может отрегулировать триггеры
- Низкоуровневые обнаружения используются где только возможно, чтобы минимизировать количество неподдерживаемых элементов данных
- Шаблоны поставляются для SNMPv2. SNMPv1 используется, если известно что большинство устройств не поддерживают SNMPv2.
- 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 фильтруются при обнаружении, также как и интерфейсы с ifAdminStatus = down(2)
- Используются 64-битные счетчики из IF-MIB::ifXTable, где это возможно. Если они не поддерживаются, вместо них используются 32-битные счетчики по умолчанию.
- У всех обнаруженных сетевых интерфейсов имеется триггер, который наблюдает за его рабочим состоянием (канал связи).
 - Если вы не хотите контролировать это состояние по какому-либо конкретному интерфейсу, создайте контекст пользовательского макроса со значением 0. Например:

Macro	Value
{\$IFCONTROL: "Gi0/0"}	⇒ 0

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 только триггер только ограниченному количеству интерфейсов, например приходящим каналам связи

Macro	Value
`\${IFCONTROL}`	⇒ 0
`\${IFCONTROL: "Gi0/0"}`	⇒ 1
`\${IFCONTROL: "Gi0/1"}`	⇒ 1

Теги

- Прозводительность – MIB'ы семейства устройств дают возможность мониторинга элементов данных CPU и памяти;
- Ошибки - MIB'ы семейства устройств дают возможность мониторинга по крайней мере одного сенсора температуры;
- Инвентарь – MIB'ы семейства устройств дают возможность сбора по крайней мене серийного номера устройства и его наименование модели;
- Сертифицированный – охватываются все три основные категории выше.

7.9 Оповещения на события

Обзор

Предположим, мы уже настроили несколько элементов данных и триггеров и теперь получаем некоторые события, которые появляются в результате 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 оповещен при наступлении определенного 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 удовлетворяют, будут произведены операции. Имеется две основных операции это отправка сообщений (оповещений) и выполнение удаленных команд.

Для созданных событий обнаружениями и автоматической регистрацией, доступны несколько дополнительных операций. Они включают в себя добавление и удаление узлов сети, присоединение шаблонов и др.

7.9.1 Способы оповещений

Обзор

Способы оповещений являются каналами доставки при отправке оповещений и предупреждений из Zabbix.

Вы можете настроить несколько способов оповещений:

- [E-mail](#)
- [SMS](#)
- [Jabber](#)
- [Ez Texting](#)
- [Пользовательские скрипты](#)

7.9.1.1 E-mail

Обзор

Для настройки e-mail каналом отправки сообщений, вам необходимо настроить e-mail способом оповещений и назначить соответствующие адреса пользователям.

Настройка

Для настройки e-mail способом оповещений:

- Перейдите в *Администрирование* → *Способы оповещений*
- Нажмите на *Создать способ оповещения* (или нажмите на *E-mail* в списке с уже имеющимися способами оповещений).

Вкладка **Способ оповещения** содержит общие атрибуты способа оповещения:

Media type Options

* Name	Email
Type	Email
* SMTP server	mail.example.com
SMTP server port	25
* SMTP helo	example.com
* SMTP email	Zabbix_info <zabbix@example.com>
Connection security	<input type="radio"/> None <input type="radio"/> STARTTLS <input checked="" type="radio"/> SSL/TLS
SSL verify peer	<input type="checkbox"/>
SSL verify host	<input type="checkbox"/>
Authentication	<input type="radio"/> None <input checked="" type="radio"/> Username and password
Username	<input type="text"/>
Password	<input type="password"/>
Message format	<input type="radio"/> HTML <input checked="" type="radio"/> Plain text
Enabled	<input checked="" type="checkbox"/>
<input type="button" value="Add"/> <input type="button" value="Cancel"/>	

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Описание

Имя	Имя способа оповещений.
Тип	Выберите <i>Email</i> , как тип.
SMTP сервер	Укажите SMTP сервер для обработки исходящих сообщений.
Порт SMTP сервера	Укажите номер порта SMTP сервера для обработки исходящих сообщений. Эта опция поддерживается начиная с Zabbix 3.0.
SMTP helo	Укажите корректное значение SMTP helo, обычно имя домена.
SMTP email	Адрес, указанный здесь, будет использован в адресе От в отправленных сообщениях. Использование отображения имени отправителя (например как “Zabbix-HQ” в <i>Zabbix-HQ <zabbix@company.com></i> на снимке экрана выше) с фактическим e-mail адресом поддерживается начиная с Zabbix 2.2. Есть некоторые ограничения на отображение имени e-mail адресов в Zabbix по стравнению с тем, что допускается по RFC 5322, как иллюстрировано в примерах: Корректные примеры: <i>zabbix@company.com</i> (только email адрес, нет нужды использовать угловые скобки) <i>Zabbix HQ <zabbix@company.com></i> (отображаемое имя и email адрес в угловых скобках) <i>Ω-monitoring <zabbix@company.com></i> (UTF-8 символы в имени)

Параметр	Описание
<i>Безопасность подключения</i>	<p>Ошибочные примеры: <i>Zabbix HQ zabbix@company.com</i> (отображаемое имя присутствует, но нет угловых скобок вокруг email адреса) <i>"Zabbix\@ \<H(comment)Q\ >" <zabbix@company.com></i> (хотя допустимо с точки зрения RFC, парные кавычки и комментарии не поддерживаются в Zabbix email)</p> <p>Выберите уровень безопасности подключения:</p> <p>Нет - не использовать опцию CURLOPT_USE_SSL</p> <p>STARTTLS - использовать опцию CURLOPT_USE_SSL со значением CURLUSESSL_ALL</p> <p>SSL/TLS - использование CURLOPT_USE_SSL опционально</p> <p>Эта опция поддерживается начиная с Zabbix 3.0.</p> <p>Отметьте для верификации SSL сертификата SMTP сервера.</p> <p>При валидации сертификата значение директивы "SSLCALocation" конфигурации сервера должно быть помещено в CURLOPT_CAPATH.</p> <p>Этот параметр использует cURL опцию CURLOPT_SSL_VERIFYPEER.</p> <p>Эта опция поддерживается начиная с Zabbix 3.0.</p>
<i>Проверка SSL узла</i>	<p>Отметьте для верификации, что поле <i>Common Name</i> или поле <i>Subject Alternate Name</i> сертификата SMTP сервера совпадают.</p> <p>Этот параметр использует cURL опцию CURLOPT_SSL_VERIFYHOST.</p> <p>Эта опция поддерживается начиная с Zabbix 3.0.</p> <p>Выберите уровень аутентификации:</p> <p>Нет - опции cURL не задаются</p>
<i>Аутентификация</i>	<p>(начиная с 3.4.2) Имя пользователя и пароль - означает "AUTH=*", оставляя выбор механизма аутентификации библиотеке cURL</p> <p>(до 3.4.2) Нормальный пароль - CURLOPT_LOGIN_OPTIONS задается равным "AUTH=PLAIN"</p> <p>Эта опция поддерживается начиная с Zabbix 3.0.</p> <p>Имя пользователя для использования в аутентификации.</p> <p>Этот параметр использует cURL опцию CURLOPT_USERNAME.</p> <p>Эта опция поддерживается начиная с Zabbix 3.0.</p>
<i>Пароль</i>	<p>Пароль для использования в аутентификации.</p> <p>Этот параметр использует cURL опцию CURLOPT_PASSWORD.</p> <p>Эта опция поддерживается начиная с Zabbix 3.0.</p>
<i>Активировано</i>	<p>Отметьте для активации способа оповещения.</p> <p>Чтобы сделать опции SMTP аутентификации доступными, Zabbix сервер необходимо скомпилировать с --with-libcurl опцией компиляции с cURL 7.20.0 или выше.</p>

Опции

Вкладка **Опции** в [настройках e-mail](#) способа оповещения содержит настройки обработки оповещений. Этот же набор опций также доступен для настройки и по другим способам оповещений.

Все способы оповещений обрабатываются в параллельном режиме. Максимальное количество одновременных сессий настраивается по каждому способу оповещений индивидуально, но общее количество процессов оповещения на сервере можно ограничить StartAlerters [параметром](#). Оповещения, которые сгенерированы одним триггером, обрабатываются последовательно.

Media type Options

Concurrent sessions	One	Unlimited	Custom	5
* Attempts	3			
* Attempt interval	10s			

Параметр	Описание
Одновременные сессии	Выберите количество параллельных сессий оповещений для выбранного способа оповещения: Одна - одна сессия Без ограничений - неограниченное количество сессий Пользовательское - укажите пользовательское количество сессий Неограниченное/высокие значения означают большее количество одновременных сессий и повышают плотность отправки оповещений. Неограниченное/высокие значения необходимо использовать в больших средах с множеством оповещений, которые могут быть отправлены одновременно.
Попыток	Количество попыток повтора отправки оповещения. Можно указать до 10 попыток; значением по умолчанию является '3'. Если указано значение '1', Zabbix отправит оповещение один раз и больше не будет пытаться, если отправка провалилась.
Интервал между попытками	Частота между попытками отправки оповещений в случае провала отправки, в секундах (0-60). Если указано значение '0', Zabbix попытается отправить снова незамедлительно. Поддерживаются суффиксы времени, например, 5s, 1m.

Оповещение пользователей

Для назначения соответствующего адреса пользователю:

- Перейдите в *Администрирование → Пользователи*
- Откройте диалог свойств пользователя
- На вкладке *Оповещения*, нажмите на *Добавить*

Media

Type	<input type="button" value="Email"/>
* Send to	<input type="text" value="Recipient name <address@company.com>"/> Remove <input type="text" value="address2@company.com"/> Remove Add
* When active	<input type="text" value="1-7,00:00-24:00"/>
Use if severity	<input checked="" type="checkbox"/> Not classified <input checked="" type="checkbox"/> Information <input checked="" type="checkbox"/> Warning <input checked="" type="checkbox"/> Average <input checked="" type="checkbox"/> High <input checked="" type="checkbox"/> Disaster
Enabled	<input checked="" type="checkbox"/>
Add Cancel	

Атрибуты оповещения пользователя:

Параметр	Описание
<i>Тип</i>	Выберите <i>Email</i> , как тип. Укажите e-mail адрес на который будут отправляться сообщения.
<i>Отправлять на</i>	Для отправки более чем на один адрес нажмите <i>Добавить</i> под полем адреса. Если указаны несколько e-mail адресов, одно e-mail сообщение будет отправлено на все заданные адреса.
<i>Когда активно</i>	Вы можете добавить отображаемое имя получателя (например, "имя получателя" в <i>Имя получателя <address1@company.com></i> как на снимке экрана выше) вместе с действительным e-mail адресом. Смотрите примеры и ограничения для отображаемого имени и e-mail адреса в описании атрибута способа оповещения <u>SMTP email</u> . Вы можете ограничить время отправки сообщений, например, только рабочие дни (1-5,09:00-18:00). Смотрите страницу <u>Спецификации периодов времени</u> для получения подробных сведений о формате. Поддерживаются <u>пользовательские макросы</u> .
<i>Использовать, если важность</i>	Отметьте важности триггеров, по которым вы хотите получать оповещения. <i>Обратите внимание</i> , что для не связанных с триггерами <u>событий</u> используется важность ('Не классифицирован') по умолчанию. После сохранения выбранные важности триггера отобразятся соответствующими цветами важностей, тогда как не выбранные важности будут заполнены серым цветом.
<i>Состояние</i>	Состояние оповещения пользователя. Активировано - используется. Деактивировано - не используется.

7.9.1.2 SMS

Обзор

Zabbix поддерживают отправку SMS сообщений с использованием GSM модемов, подключенных к последовательному порту Zabbix сервера.

Убедитесь, что:

- Скорость последовательного устройства (обычно /dev/ttyS0 порт 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 в последовательное устройство. Выполните команду ls -l /dev/ttyS0, для просмотра текущих прав доступа к последовательному устройству.
- В GSM модеме введен ПИН-код и он сохраняется даже после перезагрузки. Помимо этого, вы можете отключить запрос ПИН-кода на самой SIM карте. ПИН-код можно ввести путем использования команды AT+CPIN="NNNN" (где, NNNN - это ваш ПИН-код, кавычки должны присутствовать) в терминальном приложении, в таких как Unix minicom или Windows HyperTerminal.

Zabbix был протестирован на следующих GSM модемах:

- Siemens MC35
- Teltonika ModemCOM/G10

Для настройки SMS каналом отправки сообщений, вам необходимо настроить SMS способом оповещений и назначить соответствующие номера телефонов пользователям.

Настройка

Для настройки SMS способом оповещений:

- Перейдите в Администрирование → Способы оповещений
- Нажмите на Создать способ оповещения (или нажмите на SMS в списке с уже имеющимися способами оповещений).

Вкладка **Способ оповещения** содержит общие атрибуты способа оповещения:

Параметр	Описание
Описание	Имя способа оповещения.
Type	Выберите SMS, как тип.
GSM modem	Укажите имя serialного устройства GSM модема.
Вкладка Опции	содержит настройки обработки оповещений, которые являются общими по всем способам оповещений. Обратите внимание, что параллельная обработка отправки SMS оповещений невозможна.

Оповещение пользователя

Для назначения номера телефона пользователю:

- Перейдите в Администрирование → Пользователи
- Откройте диалог свойств пользователя
- На вкладке Оповещения, нажмите на Добавить

Атрибуты оповещения пользователя:

Параметр	Описание
<i>Тип</i>	Выберите <i>SMS</i> , как тип.
<i>Отправлять на</i>	Укажите номер телефона на который будут отправляться сообщения.
<i>Когда активно</i>	Вы можете ограничить время отправки сообщений, например, только рабочие дни (1-5,09:00-18:00). Смотрите страницу Спецификации периодов времени для получения подробных сведений о формате.
<i>Использовать, если важность</i>	Отметьте важности триггеров, по которым вы хотите получать оповещения. <i>Обратите внимание</i> , что для не связанных с триггерами событий используется важность ('Не классифицирован') по умолчанию.
<i>Состояние</i>	После сохранения выбранные важности триггера отобразятся соответствующими цветами важностей, тогда как не выбранные важности будут заполнены серым цветом. Состояние оповещения пользователя. Активировано - используется. Деактивировано - не используется.

7.9.1.3 Jabber

Обзор

Zabbix поддерживает отправку сообщений через Jabber.

При отправке оповещений, Zabbix попытается найти первую запись SRV Jabber, и если не удалось найти, тогда будет использоваться адрес записи этого домена. Среди записей SRV Jabber, будет выбрана одна с наивысшим приоритетом и с максимальным весом. Если не удалось найти такую запись, другие записи проверяться не будут.

Для настройки Jabber каналом отправки сообщений, вам необходимо настроить Jabber способом оповещений и назначить соответствующие адреса пользователям.

Настройка

Для настройки Jabber способом оповещений:

- Перейдите в *Администрирование* → *Способы оповещений*
- Нажмите на *Создать способ оповещения* (или нажмите на *Jabber* в списке с уже имеющимися способами оповещений).

Вкладка **Способ оповещения** содержит общие атрибуты способа оповещения:

Параметр	Описание
<i>Описание</i>	Имя способа оповещения.
<i>Тип</i>	Выберите <i>Jabber</i> , как тип.
<i>Идентификатор Jabber</i>	Введите идентификатор Jabber.
<i>Пароль</i>	Введите пароль Jabber.

Вкладка **Опции** содержит [настройки обработки](#) оповещений, которые являются общими по всем способам оповещений.

Оповещение пользователя

Для назначения Jabber адреса пользователю:

- Перейдите в *Администрирование* → *Пользователи*
- Откройте диалог свойств пользователя
- На вкладке Оповещения, нажмите на *Добавить*

Атрибуты оповещения пользователя:

Параметр	Описание
Тип	Выберите <i>Jabber</i> , как тип.
Отправлять на	Укажите адрес на который будут отправляться сообщения.
Когда активно	Вы можете ограничить время отправки сообщений, например, только рабочие дни (1-5,09:00-18:00). Смотрите страницу Спецификации периодов времени для получения подробных сведений о формате.
Использовать, если важность	Отметьте важности триггеров, по которым вы хотите получать оповещения. <i>Обратите внимание</i> , что для не связанных с триггерами событий используется важность ('Не классифицирован') по умолчанию. После сохранения выбранные важности триггера отобразятся соответствующими цветами важностей, тогда как не выбранные важности будут заполнены серым цветом.
Состояние	Состояние оповещения пользователя. Активировано - используется. Деактивировано - не используется.

7.9.1.4 Ez Texting

Обзор

Вы можете использовать [технологического партнера Zabbix](#) Ez Texting для отправки сообщений.

Для настройки Ez Texting каналом отправки сообщений, вам необходимо настроить Ez Texting способом оповещения и назначить идентификаторы получателя пользователям.

Настройка

Для настройки Ez Texting способом оповещений:

- Перейдите в Администрирование → Способы оповещений
- Нажмите на Создать способ оповещения



Все обязательные поля ввода отмечены красной звёздочкой.

Вкладка **Способ оповещения** содержит общие атрибуты способа оповещения:

Параметр	Описание
Описание	Имя способа оповещения.
Тип	Выберите <i>Ez Texting</i> , как тип.
Имя пользователя	Введите имя пользователя Ez Texting.
Пароль	Введите пароль к Ez Texting.
Ограничение текста сообщения	Выберите ограничение текста сообщения. США (160 символов) Канада (136 символов)

Вкладка **Опции** содержит [настройки обработки](#) оповещений, которые являются общими по всем способам оповещений.

Оповещение пользователя

Для назначения идентификации получателя Ez Texting пользователю:

- Перейдите в Администрирование → Пользователи
- Откройте диалог свойств пользователя
- На вкладке Оповещения, нажмите на Добавить

Атрибуты оповещения пользователя:

Параметр	Описание
Тип	Выберите Ez Texting, как тип.
Отправлять на	Укажите получателя, которому будут отправляться сообщения.
Когда активно	Вы можете ограничить время отправки сообщений, например, только рабочие дни (1-5,09:00-18:00). Смотрите страницу Спецификации периода времени для получения подробных сведений о формате.
Использовать, если важность	Отметьте важности триггеров, по которым вы хотите получать оповещения. <i>Обратите внимание</i> , что для не связанных с триггерами событий используется важность ('Не классифицирован') по умолчанию. После сохранения выбранные важности триггера отобразятся соответствующими цветами важностей, тогда как не выбранные важности будут заполнены серым цветом.
Состояние	Состояние оповещения пользователя. Активировано - используется. Deактивировано - не используется.

7.9.1.5 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.

Скрипты оповещений выполняются на стороне Zabbix сервера. Эти скрипты размещаются в папке указанной в переменной **AlertScriptsPath** в [файле конфигурации Zabbix сервера](#).

Пример скрипта оповещения:

```
#!/bin/bash

to=$1
subject=$2
body=$3

cat <<EOF | mail -s "$subject" "$to"
$body
EOF
```



Начиная с версии 3.4, Zabbix проверяет код выхода выполненной команды и скриптов. Любой код отличный от 0 рассматривается как ошибка [выполнения команды](#). В этом случае Zabbix попытается снова повторить выполнение, которое завершилось с ошибкой.

Переменные среды не сохраняются для скрипта, таким образом их необходимо обработать в явном виде.

Настройка

Для настройки пользовательского скрипта оповещения способом оповещений:

- Перейдите в Администрирование → Способы оповещений
- Нажмите на Создать способ оповещения

Вкладка **Способ оповещения** содержит общие атрибуты способа оповещения:

PARAMETER	ACTION
{ALERT.SENDTO}	Remove
{ALERT.SUBJECT}	Remove
{ALERT.MESSAGE}	Remove
	Remove

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Описание	Имя способа оповещения.
Тип	Выберите <i>Скрипт</i> , как тип.
Имя скрипта	Введите имя скрипта.
Параметры скрипта	Добавьте к скрипту параметры командной строки. <u>Макросы</u> {ALERT.SENDTO}, {ALERT.SUBJECT} и {ALERT.MESSAGE} поддерживаются в параметрах скрипта. Индивидуальная настройка параметров скриптов поддерживается начиная с Zabbix 3.0.

Вкладка **Опции** содержит [настройки обработки](#) оповещений, которые являются общими по всем способам оповещений.



Поскольку начиная с Zabbix 3.4.0 реализована параллельная обработка способов оповещения, важно отметить, что при наличии более одного способа оповещения с типом скрипт, эти скрипты могут обрабатываться процессами оповещения параллельно. Общее количество процессов оповещения ограничено StartAlerters [параметром](#).

Оповещение пользователя

Для назначения пользовательских скриптов оповещения пользователю:

- Перейдите в Администрирование → Пользователи
- Откройте диалог свойств пользователя

- На вкладке Оповещения, нажмите на Добавить

Атрибуты оповещения пользователя:

Параметр	Описание
Тип	Выберите пользовательские скрипты, как тип.
Отправлять на	Укажите получателя, которому будут отправляться сообщения.
Когда активно	Вы можете ограничить время отправки сообщений, например, только рабочие дни (1-5,09:00-18:00). Смотрите страницу Спецификации периодов времени для получения подробных сведений о формате.
Использовать, если важность	Отметьте важности триггеров, по которым вы хотите получать оповещения. <i>Обратите внимание</i> , что для не связанных с триггерами событий используется важность ('Не классифицирован') по умолчанию. После сохранения выбранные важности триггера отобразятся соответствующими цветами важностей, тогда как не выбранные важности будут заполнены серым цветом.
Состояние	Состояние оповещения пользователя. Активировано - используется. Деактивировано - не используется.

7.9.2 Действия

Обзор

Если вы хотите, чтобы некоторые операции выполнялись в результате событий (например, отправка оповещений), вам нужно настроить действия.

Действия можно задать для событий со всех поддерживаемых источников:

- События на триггеры - при изменениях состояния триггеров с ОК на ПРОБЛЕМА и обратно
- События на обнаружения - когда производится сетевое обнаружение
- События на авторегистрацию - при авторегистрации новых активных агентов (или при изменениях в метаданных узла сети, который уже зарегистрирован)
- Внутренние события - когда элементы данных становятся неподдерживаемыми и триггеры переходят в неизвестное состояние

Настройка действия

Для настройки действия, сделайте следующее:

- Перейдите в *Настройка* → *Действия*
- Из выпадающего меню *Источник события* выберите требуемый источник
- Нажмите на *Создать действие*
- Имя действия
- Выберите условия при которых будут выполняться операции
- Выберите осуществляемые [операции](#)
- Выберите осуществляемые [операции восстановления](#)

Общие атрибуты действий:

Actions

Action Operations Recovery operations Update operations

* Name Report problems to Zabbix administrators

Type of calculation And/Or A and B

Conditions

Label	Name
A	Problem is not suppressed
B	Host group = Zabbix servers

New condition

Host group = type here to search

Add

Enabled

* At least one operation, recovery operation or update operation must exist.

Add Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	Уникальное имя действия. Выберите опцию вычисления условий действия (с более чем одним условием): И - должны быть выполнены все условия Или - достаточно выполнения одного условия
Тип вычисления	И/Или - комбинация из двух опций: И с различными типами условий и ИЛИ с одинаковым типом условий Пользовательское выражение - формула вычисления, введенная пользователем, для оценки условий действия.
Условия	Список условий действия.
Новое условие	Выберите новое условие действия и нажмите на <i>Добавить</i> .
Активировано	Отметьте для активации действия. В противном случае действие будет деактивировано.

7.9.2.1 Условия

Обзор

Действие будет выполнено только в случае, если событие удовлетворяет заданному набору условий. Условия задаются при настройке [действия](#).

Следующие условия можно задать для действий основанных на триггерах:

Тип условия	Поддерживаемые операторы	Описание
Группа элементов данных	= содержит не содержит	Укажите группу элементов данных или исключение группы элементов данных. = - событие относится к триггеру, элемент данных которого принадлежит указанной группе элементов данных.

Тип условия	Поддерживаемые операторы	Описание
<i>Группа узлов сети</i>	= <>	содержит - событие относится к триггеру, элемент данных которого принадлежит группе элементов данных, содержащей указанную строку. не содержит - событие относится к триггеру, элемент данных которого принадлежит группе элементов данных, не содержащей указанную строку. Укажите группу узлов сети или исключение группы узлов сети. = - событие относится к указанной группе узлов сети. <> - событие не относится к указанной группе узлов сети. Указав родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Чтобы выбрать только родительскую группу, все вложенные группы необходимо дополнительно указать с оператором <>.
<i>Шаблон</i>	= <>	Укажите шаблон или исключение шаблона. = - событие относится к триггеру унаследованному из указанного шаблона. <> - событие не относится к триггеру унаследованному из указанного шаблона.
<i>Узел сети</i>	= <>	Укажите узел сети или исключение узла сети. = - событие относится к указанному узлу сети. <> - событие не относится к указанному узлу сети.
<i>Тег</i>	= <> содержит не содержит	Укажите тег события или исключение тега события. = - событие имеет указанный тег <> - событие не имеет указанный тег содержит - событие имеет тег, который содержит указанную строку не содержит - событие имеет тег, который не содержит указанную строку
<i>Значение тега</i>	= <> содержит не содержит	Укажите комбинацию тега события и его значение или исключение комбинации тега и значения. = - событие имеет указанные тег и значение <> - событие не имеет указанные тег и значение содержит - событие содержит указанные строки тега и значения не содержит - событие не содержит указанные строки тега и значения
<i>Триггер</i>	= <>	Укажите триггер или исключение триггера. = - событие сгенерировано указанным триггером. <> - событие сгенерировано любым триггером отличным от указанного.
<i>Имя триггера</i>	содержит не содержит	Укажите строку из имени триггера или исключение строки. содержит - событие произошло от триггера, который содержит в имени указанную строку. Регистрозависимое. не содержит - указанная строка не содержится в имени триггера. Регистrozависимое. <i>Обратите внимание:</i> Заданное значение будет сравниваться с именем триггера, со всеми раскрытыми макросами.
<i>Важность триггера</i>	= <> >= <=	Укажите важность триггера. = - совпадает с указанной важностью триггера <> - не совпадает с указанной важностью триггера => - выше или совпадает указанной важности триггера

Тип условия	Поддерживаемые операторы	Описание
Период времени	в не в	<p><= - ниже или совпадает указанной важности триггера</p> <p>Укажите период времени или исключение периода времени.</p> <p>в - время события в течении указанного периода времени.</p> <p>не в - время события не входит в указанный период времени.</p> <p>Смотрите страницу Спецификации периодов времени для получения более подробных сведений об этом формате.</p> <p>Поддерживаются пользовательские макросы начиная с Zabbix 3.4.0.</p> <p>Укажите должен быть узел сети находиться в обслуживании или нет.</p> <p>в - узел сети в режиме обслуживания.</p> <p>не в - узел сети не в режиме обслуживания.</p> <p>Обратите внимание: Если в выражении триггера указано несколько узлов сети, условие выполняется, если по крайней мере один из узлов сети не/в режиме обслуживания.</p> <p>Укажите, если проблема подавлена (не отображается) по причине обслуживания узла сети.</p> <p>нет - проблема не подавлена.</p> <p>да - проблема подавлена.</p>
Состояние обслуживания	в не в	
Проблема подавлена	нет да	

Следующие условия можно задать для действий основанных на обнаружении:

Тип условия	Поддерживаемые операторы	Описание
IP узла сети	= <>	<p>Укажите диапазон IP адресов или исключение диапазона у обнаруженного узла сети.</p> <p>= - IP адрес узла сети в указанном диапазоне.</p> <p><> - IP адрес узла сети не из указанного диапазона.</p> <p>Доступны следующие форматы: :</p> <p>Одиночный IP: 192.168.1.33</p> <p>Диапазон IP адресов:</p> <p>192.168.1.1-254 (до Zabbix 2.4.4)</p> <p>192.168.1-10.1-254 (начиная с Zabbix 2.4.4)</p> <p>маска IP: 192.168.4.0/24</p> <p>Список диапазонов: 192.168.1.1-254, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24</p> <p>Поддержка пробелов в формате списка имеется начиная с Zabbix 3.0.0.</p> <p>Укажите тип сервиса обнаруженного сервиса или исключение типа сервиса.</p> <p>= - совпадает с обнаруженным сервисом.</p> <p><> - не совпадает с обнаруженным сервисом.</p>
Тип сервиса	= <>	<p>Доступные типы сервисов: SSH, LDAP, SMTP, FTP, HTTP, HTTPS (доступно с Zabbix 2.2), POP, NNTP, IMAP, TCP, Zabbix агент, SNMPv1 агент, SNMPv2 агент, SNMPv3 агент, ICMP пинг, telnet (доступно начиная с версии Zabbix 2.2)</p>
Порт сервиса	= <>	<p>Укажите диапазон TCP портов или исключение диапазона у обнаруженного узла сети.\</p> <p>= - порт сервиса из указанного диапазона.</p> <p><> - порт сервиса не из указанного диапазона.</p>
Правило обнаружения	= <>	<p>Укажите правило обнаружения или исключение правила обнаружения.</p> <p>= - использование указанного правила обнаружения.</p> <p><> - использование любого другого правила обнаружения,</p>

Тип условия	Поддерживаемые операторы	Описание
Проверка обнаружения	= <>	отличного от указанного. Укажите проверку обнаружения или исключение проверки обнаружения. = - использование указанной проверки обнаружения. <> - использование любой другой проверки обнаружения, отличной от указанной.
Обнаруженный объект	=	Укажите обнаруженный объект. = - совпадает с обнаруженным объектом (устройством или сервисом).
Состояние обнаружения	=	Доступен - совпадает с событиями 'Узел сети доступен' и 'Сервис доступен' Недоступен - совпадает с событиями 'Узел сети недоступен' и 'Сервис недоступен' Обнаружен - совпадает с событиями 'Узел сети обнаружен' и 'Сервис обнаружен' Потерян - совпадает с событиями 'Узел сети потерян' и 'Сервис потерян'
Доступен/ Недоступен	>= <=	Время доступности для событий 'Узел сети доступен' и 'Сервис доступен'. Время недоступности для событий 'Узел сети недоступен' и 'Сервис недоступен'. >= - больше или совпадает. Параметр задается в секундах. <= - меньше или совпадает. Параметр задается в секундах.
Полученное значение	= <> >= <=	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). = - совпадает с указанным значением. <> - не совпадает с указанным значением. >= - больше или совпадает с указанным значением. <= - меньше или совпадает с указанным значением. содержит - содержит указанную подстроку. Параметр задается в виде строки. не содержит - не содержит указанную подстроку. Параметр задается в виде строки.
Прокси	= <>	Укажите прокси или исключение прокси. = - используется указанный прокси. <> - используется любой другой прокси отличный от указанного.



Проверки сервисов в правиле обнаружения, которые приводят к событиям обнаружения, не выполняются одновременно. Поэтому, если настроено **несколько** значений для Тип сервиса, Порт сервиса или Полученное значение условий в действии, они будут сравниваться с одним событием обнаружения за раз, но **не** сравнивается с несколькими событиями одновременно. В результате действия с несколькими значениями по одним и тем же типам проверок могут быть выполнены некорректно.

Следующие условия можно задать для действий основанных на авто-регистрации активного агента:

Тип условия	Поддерживаемые операции	Описание
Метаданные	содержит	Укажите метаданные узла сети или метаданные узла сети

Тип условия	Поддерживаемые операции	Описание
узла сети	не содержит	для исключения. содержит - метаданные узла сети содержат указанную строку не содержит - метаданные узла сети не содержат указанную строку. Метаданные узла сети можно задать в файле конфигурации агента
Имя узла сети	содержит не содержит	Укажите имя узла сети или имя узла для исключения. содержит - имя узла сети содержит указанную строку не содержит - имя узла сети не содержит указанную строку.
Прокси	= <>	Укажите прокси или прокси для исключения: = - используется указанный прокси. <> - используется любой другой прокси отличный от указанного.

Следующие условия можно задать для действий основанных на внутренних событиях:

Тип условия	Поддерживаемые операции	Описание
Группа элемента данных	= содержит не содержит	Укажите группу элементов данных или группу элементов данных для исключения. = - событие относиться к элементу данных, который принадлежит указанной группе элементов данных. содержит - событие относиться к элементу данных, принадлежит указанной группе элементов данных содержащей указанную строку не содержит - событие относиться к элементу данных, который принадлежит указанной группе элементов данных не содержащей указанную строку
Тип события =		Элемент данных в состоянии “не поддерживается” - соответствует событию, когда элемент данных переходит из состояния 'активировано' в состояние 'не поддерживается' Правило низкоуровневого обнаружения в состоянии “не поддерживается” - соответствует событию, когда правило низкоуровневого обнаружения переходит из состояния 'активировано' в состояние 'не поддерживается' Триггер в состоянии “неизвестно” - соответствует событию, когда триггер переходит из состояния 'активировано' в состояние 'неизвестно'
Группа узла сети	= <>	Укажите группу узла сети или группу узла сети для исключения. = - событие относится к указанной группе узлов сети. <> - событие не относится к указанной группе узлов сети.
Шаблон	= <>	Укажите шаблон или шаблон для исключения. = - событие относится к элементу данных/триггеру/правилу низкоуровневого обнаружения унаследованных из указанного шаблона. <> - событие не относится к элементу данных/триггеру/правилу низкоуровневого обнаружения унаследованных из указанного шаблона.
Узел сети	= <>	Укажите узел сети или узел сети для исключения. = - событие относится к указанному узлу сети. <> - событие не относится к указанному узлу сети.

Тип вычисления

Для вычисления условий доступны следующие опции:

- **И** - должны быть выполнены все условия

Обратите внимание, что вычисление “И” не нужно использовать между несколькими триггерами, когда они добавлены с условием Триггер=. Действия могут выполняться, основываясь на событии только одного триггера.

- **Или** - достаточно выполнения одного условия
- **И/Или** - комбинация из двух опций: И с различными типами условий и ИЛИ с одинаковым типом условий, например:

Группа узлов сети = Oracle сервера

Группа узлов сети = MySQL сервера

Имя триггера содержит 'База данных не работает'

Имя триггера содержит 'База данных недоступна'

вычисляется как

(Группа узлов сети = Oracle сервера **или** Группа узлов сети = MySQL сервера) **и** (Имя триггера содержит 'База данных не работает' **или** Имя триггера содержит 'База данных недоступна')

- **Пользовательское выражение** - формула вычисления, введенная пользователем, для оценки условий действия. Она должна включать в себя все условия (представленные в виде прописных букв A, B, C, ...) и может включать пробелы, символы табуляции, скобки (), **and** (с учетом регистра), **or** (с учетом регистра), **not** (с учетом регистра).

Тогда как приведенный пример с И/Или был бы представлен в виде (A или B) и (C или D), в пользовательском выражении вы также можете использовать несколько других методов вычисления:

(A и B) и (C или D)

(A и B) или (C и D)

((A или B) и C) или D

(not (A or B) and C) or not D

и так далее.

Действия деактивированы из-за удаленных объектов

Если заданный объект (узел сети, шаблон, триггер и т.п.), используемый в условиях/операции действия, удаляется, то условие/операция удаляются и действие деактивируется во избежании некорректного выполнения действия. Действие можно активировать снова вручную.

Такое поведение имеет место при удалении:

- групп узлов сети (условие “группа узлов сети”, операция “удаленная команда” на указанной группе узлов сети);
- узлы сети (условие “узел сети”, операция “удаленная команда” на указанном узле сети);
- шаблоны (условие “шаблон”, операции “присоединить к шаблону” и “отсоединить от шаблона”);
- триггеры (условие “триггер”);
- правила обнаружения (при использовании условий “правило обнаружения” и “проверка обнаружения”).

Обратите внимание: Если удаленная команда имеет несколько целевых узлов сети, и мы удаляем один из них, будет удален только этот узел из списка целей, сама операция останется. Однако, если операция состоит только из одного узла сети, она будет также удалена. Такое поведение касается также операций “присоединить к шаблону” и “отсоединить от шаблона”.

Действия не деактивируются при удалении пользователя или группы пользователей, которые используются в операции “отправка сообщений”.

7.9.2.2 Операции

Обзор

Вы можете указать следующие операции для всех источников:

- отправка сообщений
- выполнение удаленных команд (включая IPMI)



Zabbix сервер не создает оповещения, если доступ к узлу сети "запрещен" пользователю явным образом, который задан в операции получателем, или, если пользователь вовсе не имеет заданных прав доступа к этому узлу сети.

Для событий обнаружения и авторегистрации, также доступны дополнительные операции:

- [добавление узла сети](#)
- добавление узла сети
- удаление узла сети
- активация узла сети
- деактивация узла сети
- добавление в группу узлов сети
- удаление из группы узлов сети
- присоединение шаблона
- отсоединение шаблона
- установка режима инвентарных данных узлам сети

Настройка операции

Для настройки операции, в диалоге [настройки](#) действия перейдите на вкладку *Операции* и в блоке Операции нажмите на *Новая*. Измените шаг операции и нажмите *Добавить* для добавления в список *Операции*.

Атрибуты операции:

Action Operations Recovery operations Update operations

* Default operation step duration

Default subject Problem: {EVENT.NAME}

Default message

```
Problem started at {EVENT.TIME} on {EVENT.DATE}
Problem name: {TRIGGER.NAME}
Host: {HOST.NAME}
Severity: {EVENT.SEVERITY}

Original problem ID: {EVENT.ID}
{TRIGGER.URL}
```

Pause operations for suppressed problems

Operations	Steps Details	Start in	Duration	Action
1	Send message to user groups: Zabbix administrators via Email	Immediately	Default	Edit Remove
3	Send message to user groups: Managers via SMS	02:00:00	Default	Edit Remove
4	Run remote commands on current host	03:00:00	Default	Edit Remove

Operation details

Steps - (0 - infinitely)

Step duration (0 - use action default)

Operation type [Send message](#)

* At least one user or user group must be selected.

Send to User groups

User group	Action
Managers	Remove
Add	

Send to Users

User	Action
Add	

Send only to [SMS](#)

Default message

Conditions

Label	Name	Action
A	Event acknowledged = Not Ack	Remove
New		

[Update](#) [Cancel](#)

* At least one operation, recovery operation or update operation must exist.

[Add](#) [Cancel](#)

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Длительность шага операции по умолчанию

Тема по умолчанию

Сообщение по умолчанию

Описание

Длительность одного шага операции по умолчанию (от 60 секунд до 1 недели).

Например, длительность шага в один час означает, что если эта операция выполняется, то следующая операция начнется через один час.

Начиная с Zabbix 3.4.0, поддерживаются [суффиксы времени](#), например, 60s, 1m, 2h, 1d.

Поддерживаются [пользовательские макросы](#) начиная с 3.4.0.

Тема сообщений по умолчанию для оповещений. Тема может содержать [макросы](#). Это поле ограничено длиной в 255 символов.

Сообщение по умолчанию для оповещений. Сообщение может содержать [макросы](#). Это поле ограничено некоторым количеством символов, которое зависит от

Параметр	Описание
<i>Приостановить операции для подавленных проблем</i>	типа базы данных (для получения более подробных сведений смотрите Отправка сообщения). Отметьте эту опцию для задержки начала операций в течении периода обслуживания. Когда операции начнутся, после периода обслуживания, выполняются все операции, включая те, которые для событий во время обслуживания.
<i>Операции</i>	Если вы не отметите, операции будут выполняться без задержки даже в процессе периода обслуживания. Эта опция поддерживается начиная с Zabbix 3.2.0. Операции действия отображаются со следующими деталями: Шаги - шаг(и) эскалации к которым назначаются операции Детали - тип операции и ее получатель/цель. Начиная с Zabbix 2.2, список операций также отражает способ оповещений (e-mail, SMS, Jabber и т.п.), используемый для отправки сообщения, а также имя и фамилию (в круглых скобках после псевдонима) получателя оповещения. Начинать в - через какое время после события выполнится эта операция Длительность (сек) - отображение длительности шага. Отображается по умолчанию, если шаг имеет длительность по умолчанию, и отображается время, если используется указанное время. Действие - отображаются ссылки для изменения и удаления операции. Для добавления новой операции, нажмите на <i>Новый</i> . Этот блок используется для настройки параметров операции. Выберите шаг(и) для назначения операции в расписание эскалаций : От - выполнять, начиная с этого шага До - выполнять, до этого шага (0=бесконечно, выполнение не будет ограничено) Пользовательская длительность для этих шагов (0=использовать длительность шага по умолчанию). Начиная с Zabbix 3.4.0, поддерживаются суффиксы времени , например, 60s, 1m, 2h, 1d. Поддерживаются пользовательские макросы начиная с 3.4.0. На один шаг можно назначить несколько операций. Если такие операции будут иметь разную длительность шага, то учитывается самая короткая и она будет применена к шагу. Для всех событий доступно два типа операций: Отправлять сообщение - отправление сообщения пользователю Удаленная команда - выполнение удаленной команды Остальные операции доступны для событий основанных на обнаружении и автоматической регистрации (см. выше). Тип операции: отправка сообщений
<i>Детали операций</i>	
<i>Шаги</i>	
<i>Длительность шага</i>	
<i>Тип операции</i>	

Параметр	Описание
<i>Отправлять группам пользователей</i>	Нажмите на <i>Добавить</i> для выбора групп пользователей, которым будет отправляться сообщение. Группа пользователей должна иметь по крайней мере права доступа на “чтение” узла сети, чтобы получить уведомления.
<i>Отправлять пользователям</i>	Нажмите на <i>Добавить</i> для выбора пользователей, которым будет отправляться сообщение. Пользователь должен иметь по крайней мере права доступа на “чтение” узла сети, чтобы получить уведомления.
<i>Отправлять только через Сообщение по умолчанию</i>	Отправка сообщения по всем заданным способам оповещений или только по выбранному. Если выбрано, будет использовано сообщение по умолчанию (смотрите выше).
<i>Тема</i>	Тема пользовательского сообщения. Тема может содержать макросы. Это поле ограничено длиной в 255 символов.
<i>Сообщение</i>	Пользовательское сообщение. Сообщение может содержать макросы. Это поле ограничено некоторым количеством символов, которое зависит от типа базы данных (для получения более подробных сведений смотрите Отправка сообщения).
<i>Список целей</i>	Тип операции: удаленная команда Выберите цели, на которых будет выполняться команда: Текущий узел сети - команда будет выполнена на узле сети триггер которого вызвал это событие о проблеме. Эта опция не будет работать, если триггер содержит несколько узлов сети. Узел сети - выберите узел/узлы сети, на котором команда будет выполнена. Группа узлов сети - выберите группу(ы) узлов сети, на котором команда будет выполнена. Указав родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Таким образом, удаленная команда будет также выполнена на узлах сети из вложенных групп. Команда выполняется на узле сети только один раз, независимо от того совпадает ли узел сети более одного раза (например, входит в несколько групп узлов сети; указан отдельно и входит в группу узлов сети). Список целей не имеет смысла, если пользовательский скрипт выполняется на Zabbix сервере. В этом случае выбор нескольких целей приведет только к тому, что скрипт выполнится на сервере несколько раз. Обратите внимание, выбор цели также зависит от настройки <i>Группа пользователей</i> в конфигурации глобального скрипта. Выберите тип команды: IPMI - выполнение IPMI команды Пользовательский скрипт - выполнение пользовательского набора команд. SSH - выполнение SSH команды Telnet - выполнение Telnet команды
<i>Тип</i>	

Параметр

Описание

Глобальный скрипт - выполнение одного из глобальных скриптов определенных в *Администрирование → Скрипты*.

Выполнение пользовательского скрипта на:

Zabbix агент - скрипт будет выполнен Zabbix агентом на хосте

Zabbix сервер (прокси) - скрипт будет выполнен Zabbix сервером или прокси - в зависимости от того кем наблюдается узел сети сервером или прокси

Zabbix сервер - скрипт будет выполнен только на стороне Zabbix сервера

Для выполнения скриптов на стороне агента необходимо настроить (включён параметр *EnableRemoteCommands*) разрешение выполнения удаленных команд с сервера.

Для выполнения скриптов на стороне прокси необходимо настроить (включён параметр *EnableRemoteCommands*) разрешение выполнения удаленных команд с сервера.

Это поле доступно, если как *Тип* выбран 'Пользовательский скрипт'.

Введите команду(ы).

Поддерживаемые макросы будут раскрыты на основании выражения триггера, которое вызвало событие.

Например, макрос узла сети будет раскрыт в узлы сети из выражения триггера (не в узлы сети из списка целей).

Условие выполнения операции:

Не подтверждено - только, если событие не подтверждено

Подтверждено - только, если событие подтверждено

Выполнить на

Команды

Условия

7.9.2.2.1 Отправка сообщений

Обзор

Отправка сообщений является одним из лучших способов уведомить 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 вы должны:

- задать оповещения для отправки сообщения



Важность триггера по умолчанию ('Не классифицировано') **должна быть** выбрана в настройках оповещения пользователя, если вы хотите получать оповещения на события не связанные с триггерами, такими как обнаружение, авторегистрация активных агентов или внутренние события.

- настроить операцию действия, которая отправит сообщения по одному из заданных способов оповещений



Zabbix отправит оповещения только тем пользователям, которые имеют по крайней мере права на 'чтение' узла сети, который сгенерировал это событие. По крайней мере один узел сети из выражения триггера должен быть доступен.

Вы можете настроить пользовательские сценарии для отправки сообщений, используя эскалации.

Для успешного получения и чтения e-mail от Zabbix, e-mail сервера/клиенты должны поддерживать стандарт 'SMTP/MIME e-mail', поскольку Zabbix отправляет UTF-8 данные (если тема содержит только ASCII символы, она не будет перекодирована в UTF-8). Тема и тело сообщения кодируются в base64, чтобы они соответствовали стандарту 'SMTP/MIME e-mail' формата.

Ограничение длины сообщения после того, как все макросы раскрыты, такое же как и ограничение длины сообщения при [Удаленных командах](#).

Отслеживание сообщений

Вы можете увидеть состояние отправления сообщений в *Мониторинг → Проблемы*.

В колонке *Действия* вы можете видеть обобщенную информацию о принятых мерах. Здесь, зеленые цифры означают отправленные сообщения, красные - сообщения с ошибками. В процессе указывает на то, что действие началось. Ошибочно информирует о том, что действия не были выполнены удачно.

Если вы нажмёте на времени события для просмотра деталей события, вы также увидите блок *Сообщения действий*, который содержит детали об отправленных (или не отправленных) сообщениях связанных с выбранным событием.

В *Отчеты → Журнал действий* вы увидите подробную информацию о всех мерах, предпринятых для этих событий, которые имеют какое-либо настроенное действие.

7.9.2.2 Удаленные команды

Обзор

С помощью удаленных команд вы можете указать какие из предустановленных команд будут выполнены на наблюдаемом узле сети при выполнении некоторых условий.

Таким образом, удаленные команды являются мощным механизмом для умного про-активного мониторинга.

Из наиболее очевидных применений функции вы можете попробовать:

- Автоматический перезапуск какого-либо приложения (веб-сервер, промежуточное ПО, CRM), если оно не отвечает
- Использовать команду IPMI 'reboot' для перезагрузки некоторого удаленного сервера, если он не отвечает на запросы
- Автоматическую очистку диска (удаление старых файлов, очистка /tmp), если закончилось место на диске
- Миграцию VM с одного физического сервера на другой, в зависимости от загрузки CPU
- Добавление новых узлов в облачную среду при недостаточных ресурсах CPU (диска, памяти, чего-угодно)

Настройка действий для удаленных команд аналогична настройке действий для отправки сообщений, с той лишь разницей, что вместо отправки сообщения Zabbix будет выполнять команду.

Удаленные команды могут выполняться на стороне Zabbix сервера, прокси или агента. Удаленные команды на Zabbix агенте могут выполняться напрямую Zabbix сервером или через Zabbix прокси. По умолчанию удаленные команды отключены как у Zabbix агента, так и у Zabbix прокси. Их можно включить, изменив значение параметра `EnableRemoteCommands` на значение '1'.

Ограничение длины удаленных команд после раскрытия всех макросов зависит от типа базы данных и заданной кодировки (для хранения не-ASCII символов требуется более одного байта):

<i>База данных</i>	<i>Ограничение в символах</i>	<i>Ограничение в байтах</i>
MySQL	65535	65535
Oracle Database	2048	4000
PostgreSQL	65535	не ограничено
IBM DB2	2048	2048
SQLite (только Zabbix прокси)	65535	не ограничено
Удаленные команды выполняемые Zabbix сервером запускаются как описано в разделе Выполнение команд , включая проверку кода выхода.		

Удаленные команды будут выполнены даже если намеченный узел сети в обслуживании.

Следующее руководство содержит пошаговые инструкции о том, как настроить удаленные команды.

Настройка

Удаленные команды, которые выполняются на стороне Zabbix агента (пользовательские скрипты) сначала должны быть включены в соответствующем [zabbix_agentd.conf](#).

Убедитесь, что параметр **EnableRemoteCommands** равен **1** и раскомментирован. Перезапустите демона агента, если изменили этот параметр.



Удаленные команды не работают с активными Zabbix агентами.

Затем, при настройке нового действия в *Настройка* → *Действия*:

- Задайте соответствующие условия. В этом примере, укажите что команда активируется в случае любых чрезвычайных проблем в одном из приложений Apache:

Action	Operations	Recovery operations								
* Name	Serious problem with Apache									
Type of calculation	And/Or	A and B and C								
Conditions	<table border="1"> <thead> <tr> <th>Label</th> <th>Name</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Maintenance status not in <i>maintenance</i></td> </tr> <tr> <td>B</td> <td>Application like <i>Apache</i></td> </tr> <tr> <td>C</td> <td>Trigger severity >= <i>Disaster</i></td> </tr> </tbody> </table>		Label	Name	A	Maintenance status not in <i>maintenance</i>	B	Application like <i>Apache</i>	C	Trigger severity >= <i>Disaster</i>
Label	Name									
A	Maintenance status not in <i>maintenance</i>									
B	Application like <i>Apache</i>									
C	Trigger severity >= <i>Disaster</i>									

Все обязательные поля ввода отмечены красной звёздочкой.

- На вкладке [Операции](#) выберите тип операции **Удаленная команда**
- Выберите тип удаленной команды (IPMI, Пользовательский скрипт, SSH, Telnet, Глобальный скрипт)
- Если выбран тип *Пользовательский скрипт*, тогда укажите каким способом этот скрипт будет выполняться (Zabbix агентом, Zabbix сервером (прокси) или только Zabbix сервером)
- Введите удаленную команду

Например:

```
sudo /etc/init.d/apache restart
```

В этом случае, Zabbix попытается перезапустить процесс Apache. Используя эту команду, убедитесь, что она будет выполнена Zabbix агентом (нажмите на *Zabbix агент* напротив *Выполнять на*).



Обратите внимание на использование **sudo** - пользователь Zabbix не имеет прав на перезапуск системных сервисов по умолчанию. Смотрите подсказки ниже о том, как настроить **sudo**.



Zabbix агент должен работать на удаленном узле сети и должен принимать входящие подключения. Zabbix агент выполняет команды в фоновом режиме.

Удаленные команды на стороне Zabbix агента выполняются без учета времени ожидания при помощи ключа system.run[,nowait], таким образом результаты выполнения не проверяются. На стороне Zabbix сервера команды выполняются с учетом времени ожидания, как указано в параметре TrapperTimeout в файле zabbix_server.conf и результаты выполнения [проверяются](#).

Права доступа

Убедитесь, что пользователь 'zabbix' имеет права на выполнение добавленных команд. Один наиболее интересный вариант - использование **sudo** для предоставления доступа к привилегированным командам. Для настройки доступа, выполните из под root:

```
# visudo
```

Примеры строк, которые можно использовать в файле sudoers:

```
# разрешает пользователю 'zabbix' выполнение любых команд без пароля.  
zabbix ALL=NOPASSWD: ALL
```

```
# разрешает пользователю 'zabbix' перезапуск apache без пароля.  
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/apache restart
```



В некоторых системах настройки в файле sudoers будут препятствовать не-локальным пользователям выполнять команды. Для изменения этого, раскомментируйте опцию requiretty в /etc/sudoers.

Удаленные команды при наличии нескольких интерфейсов

Если целевая система имеет несколько интерфейсов выбранного типа (Zabbix агент или IPMI), удаленные команды будут выполнены на интерфейсе по умолчанию.

Имеется возможность выполнения удаленных команд через SSH и Telnet, используя отличный интерфейс 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

Удаленные команды через IPMI

В случае удаленных команд через IPMI необходимо использовать следующий синтаксис:

<команда> [<значение>]

где

- <команда> - одна из IPMI команд без пробелов

- <значение> - 'on', 'off' или любое положительное целое число. <значение> является optionalным параметром.

Примеры

Пример 1

Перезагрузка Windows при определенном условии.

Для того, чтобы автоматически перезагрузить Windows, когда Zabbix обнаруживает проблемы, определим следующие действия:

ПАРАМЕТР	Описание
Тип операции	'Удаленная команда'
Тип	'Пользовательский скрипт'
Команда	c:\windows\system32\shutdown.exe -r -f

Пример 2

Перезагрузка узла сети с использованием управления по IPMI.

ПАРАМЕТР	Описание
Тип операции	'Удаленная команда'
Тип	'IPMI'
Команда	reset

Пример 3

Выключение узла сети с использованием управления по IPMI.

ПАРАМЕТР	Описание
Тип операции	'Удаленная команда'
Тип	'IPMI'
Команда	power off

7.9.2.2.3 Дополнительные операции

Обзор

В этом разделе вы можете найти некоторые детали по [дополнительным операциям](#) на события обнаружения/авторегистрации.

Добавление узла сети

Узлы сети добавляются в процессе обнаружения, как только узел сети был обнаружен, не в конце процесса обнаружения.



Сетевое обнаружение может занять некоторое время из-за большого количества недоступных узлов сети/сервисов, рекомендуется, набраться терпения и использовать разумные диапазоны IP.

При добавлении узла сети, его имя определяется стандартной функцией **gethostbyname**. Если имя узла сети можно узнать, то будет использоваться найденное имя. Если нет, будет использован IP адрес. Кроме того, если для имени узла сети должен использоваться IPv6 адрес, то все ":" (двоеточия) заменяются "_" (подчеркиваниями), так как двоеточки не разрешены в именах узлов сети.



Если обнаружение выполняется через прокси, в настоящее время преобразование имени хоста (hostname lookup) по-прежнему происходит на Zabbix сервере.



Если узел сети уже существует в конфигурации Zabbix с таким же именем, как и добавляемый, версии Zabbix до 1.8 добавят другой узел сети с таким же именем. Zabbix 1.8.1 и более поздние добавят _N к имени узла сети, где N увеличивающееся число, начиная с 2.

7.9.2.2.4 Использование макросов в сообщениях

Обзор

Вы можете использовать макросы в теме и в тексте сообщений для более эффективном оповещении о проблемах.

Имеется [полный список](#) поддерживаемых макросов в Zabbix.

Примеры

Примеры приведенные здесь, иллюстрируют то, как вы можете использовать макросы в сообщениях.

Пример 1

Тема сообщения:

Проблема: {TRIGGER.NAME}

При получении сообщения, тема этого сообщения будет заменена на нечто подобное:

Проблема: Загрузка процессора слишком высокая на Zabbix сервер

Пример 2

Сообщение:

Загрузка процессора: {zabbix.zabbix.com:system.cpu.load[,avg1].last()}

При получении сообщения, само сообщение будет заменено на нечто подобное:

Загрузка процессора: 1.45

Пример 3

Сообщение:

Последнее значение: {{HOSTNAME}}:{TRIGGER.KEY}.last()
МАКС за 15 минут: {{HOSTNAME}}:{TRIGGER.KEY}.max(900)
МИН за 15 минут: {{HOSTNAME}}:{TRIGGER.KEY}.min(900)

При получении сообщения, само сообщение будет заменено на нечто подобное:

Последнее значение: 1.45
МАКС за 15 минут: 2.33
МИН за 15 минут: 1.01

Пример 4

Сообщение:

http://<ip_сервера_или_имя>/zabbix/events.php?triggerid={TRIGGER.ID}&filter_set=1

При получении сообщения оно будет содержать ссылку на все события проблемного триггера.

Пример 5

Информирование о значениях с нескольких узлов сети из выражения триггера.

Сообщение:

Имя проблемы: {TRIGGER.NAME}
Выражение триггера: {TRIGGER.EXPRESSION}

- Значение элемента данных на {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1})
- Значение элемента данных на {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})

При получении сообщения, само сообщение будет заменено на нечто подобное:

Имя проблемы: Загрузка процессора слишком высокая на локальном хосте
Выражение триггера: {Myhost:system.cpu.load[percpu,avg1].last()}>5 | {Myotherhost:system.cpu.load[percpu,avg1].last()}>5

- Значение элемента данных на Myhost: 0.83 (Загрузка процессора (усреднение за 1 мин по ядрам))
- Значение элемента данных на Myotherhost: 5.125 (Загрузка процессора (усреднение за 1 мин по ядрам))

Пример 6

Получение деталей по событию о проблеме и событию о восстановлении в сообщении о восстановлении:

Сообщение:

Проблема:

ID события: {EVENT.ID}
Значение события: {EVENT.VALUE}
Состояние события: {EVENT.STATUS}
Время события: {EVENT.TIME}
Дата события: {EVENT.DATE}
Возраст события: {EVENT.AGE}
Подтверждение события: {EVENT.ACk.STATUS}
История обновления события: {EVENT.UPDATE.HISTORY}

Восстановление:

ID события: {EVENT.RECOVERY.ID}
Значение события: {EVENT.RECOVERY.VALUE}
Состояние события: {EVENT.RECOVERY.STATUS}
Время события: {EVENT.RECOVERY.TIME}
Дата события: {EVENT.RECOVERY.DATE}

При получении сообщения, эти макросы будут заменены на нечто подобное:

Проблема:

ID события: 21874
Значение события: 1
Состояние события: PROBLEM
Время события: 13:04:30
Дата события: 2018.01.02
Возраст события: 5m
Подтверждение события: Yes
История обновления события: 2018.01.02 13:05:51 "John Smith (Admin)"
Действия: подтверждено.

Восстановление:

ID события: 21896
Значение события: 0
Состояние события: OK
Время события: 13:10:07
Дата события: 2014.01.02



Раздельные макросы оповещений для оригинального события о проблеме и события о восстановлении поддерживаются начиная с Zabbix 2.2.0.

7.9.2.3 Операции восстановления

Обзор

Операции восстановления позволяют вам получать оповещения когда проблемы решены.

В операциях восстановления поддерживаются как сообщения, так и удаленные команды. Операции восстановления не поддерживают эскалации - все операции назначаются на единственный шаг.

Сценарии применения

Некоторые сценарии использования операций восстановления заключаются в следующем:

1. Оповещение всех пользователей, которые были оповещены о проблеме
 - Выберите типом операции 'Отправка сообщения о восстановлении'
2. Наличие нескольких операций по восстановлению: отправка оповещений и удаленная команда
 - 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 взаимодействует с системой поддержки клиентов
 - Создайте действие, которое имеет 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} как ссылку на оригинальную проблему

Настройка операции восстановления

Для настройки операции восстановления:

- Перейдите на вкладку *Операции восстановления* в [настройках](#) действия
- Нажмите на *Новый* в блоке Операций
- Измените детали операции и нажмите на *Добавить*

Можно добавить несколько операций.

Атрибуты операции восстановления:

Action Operations Recovery operations Update operations

Default subject	Resolved: {EVENT.NAME}																				
Default message	Problem has been resolved at {EVENT.RECOVERY.TIME} on {EVENT.RECOVERY.DATE} Problem name: {TRIGGER.NAME} Host: {HOST.NAME} Severity: {EVENT.SEVERITY}																				
	Original problem ID: {EVENT.ID} {TRIGGER.URL}																				
Operations	Details	Action																			
	Notify all involved	Edit Remove																			
	Run remote commands on current host	Edit Remove																			
Operation details	Operation type: Remote command <table border="1"> <tr> <td>* Target list</td> <td>Target</td> <td>Action</td> </tr> <tr> <td></td> <td>Current host</td> <td>Remove</td> </tr> <tr> <td></td> <td>New</td> <td></td> </tr> <tr> <td>Type</td> <td>Custom script</td> <td></td> </tr> <tr> <td>Execute on</td> <td>Zabbix agent</td> <td>Zabbix server (proxy)</td> <td>Zabbix server</td> </tr> <tr> <td>* Commands</td> <td colspan="2">sudo /etc/init.d/apache2 restart</td> </tr> </table>		* Target list	Target	Action		Current host	Remove		New		Type	Custom script		Execute on	Zabbix agent	Zabbix server (proxy)	Zabbix server	* Commands	sudo /etc/init.d/apache2 restart	
* Target list	Target	Action																			
	Current host	Remove																			
	New																				
Type	Custom script																				
Execute on	Zabbix agent	Zabbix server (proxy)	Zabbix server																		
* Commands	sudo /etc/init.d/apache2 restart																				
	Update Cancel																				
* At least one operation, recovery operation or update operation must exist.																					
	Add	Cancel																			

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Тема по умолчанию

Сообщение по умолчанию

Операции

Детали операций

Tip операции

Описание

Тема сообщений по умолчанию для оповещений о восстановлении. Тема может содержать [макросы](#).

Сообщение по умолчанию для оповещений о восстановлении. Сообщение может содержать [макросы](#).

Детали операции восстановления.

Для настройки новой операции восстановления, нажмите на *Новая*.

Этот блок используется для настройки деталей операции восстановления.

Для событий восстановления доступны три типа операций: **Отправлять сообщение** - отправление сообщения о восстановлении конкретному пользователю

Удаленная команда - выполнение удаленной команды

Оповещать всех участников - отправка сообщения о восстановлении всем пользователям, которые ранее получали

Параметр	Описание
<i>Отправлять группам пользователей</i>	оповещения о событии проблем Обратите внимание что, если в нескольких типах операций задан один и тот же получатель с не изменёнными темой/сообщением по умолчанию, тогда дубликаты оповещений отправлены не будут.
<i>Отправлять пользователям</i>	Тип операции: <u>отправлять сообщение</u> Нажмите на Добавить для выбора групп пользователей, которым будет отправляться сообщение о восстановлении. Группа пользователей должна иметь по крайней мере права доступа на “чтение” узла сети, чтобы получить уведомления.
<i>Отправлять только через Сообщение по умолчанию</i>	Нажмите на Добавить для выбора пользователей, которым будет отправляться сообщение о восстановлении. Пользователь должен иметь по крайней мере права доступа на “чтение” узла сети, чтобы получить уведомления.
<i>Тема</i>	Отправка сообщения о восстановлении по всем заданным способам оповещений или только по выбранному. Если выбрано, будет использовано сообщение по умолчанию (смотрите выше).
<i>Сообщение</i>	Тема пользовательского сообщения. Тема может содержать макросы. Пользовательское сообщение. Сообщение может содержать макросы.
<i>Список целей</i>	Тип операции: <u>удаленная команда</u> Выберите цели для выполнения команды: Текущий узел сети - команда выполнится на узле сети триггера, который вызвал событие о проблеме. Эта опция не будет работать, если в выражении триггера имеется несколько узлов сети. Узел сети - выберите узел(ы) сети для выполнения команды. Группа узлов сети - выберите группу(ы) узлов сети для выполнения команды. Указав родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Таким образом команда будет выполнена на узлах сети из вложенных групп. 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). Список целей не имеет смысла, если команда выполняется на стороне Zabbix сервера. В этом случае выбор нескольких целей приведет только к тому, что команда выполнится на сервере несколько раз. Обратите внимание, что в случае глобальных скриптов выбор цели также зависит от настройки <i>Группа узлов сети</i> в настройках глобального скрипта. Выберите тип команды: IPMI - выполнение IPMI команды Пользовательский скрипт - выполнение пользовательского набора команд. SSH - выполнение SSH команды Telnet - выполнение Telnet команды Глобальный скрипт - выполнение одного из глобальных скриптов определенных в <i>Администрирование → Скрипты</i> . Выполнение пользовательского скрипта на:
<i>Tip</i>	
<i>Выполнить на</i>	

Параметр	Описание
	Zabbix агент - скрипт будет выполнен на узле сети Zabbix агентом
	Zabbix сервер (прокси) - скрипт будет выполнен Zabbix сервером или прокси - в зависимости от того кем наблюдается узел сети сервером или прокси
	Zabbix сервер - скрипт будет выполнен только на стороне Zabbix сервера
	Для выполнения скриптов на стороне агента необходимо настроить разрешение выполнения удаленных команд с сервера. Это поле доступно, если как <i>Тип</i> выбран 'Пользовательский скрипт'.
	Введите команду(ы).
Команды	Поддерживаемые макросы будут раскрыты на основании выражения триггера, которое вызвало событие. Например, макрос узла сети будет раскрыт в узлы сети из выражения триггера (не в узлы сети из списка целей).
	Тип операции: оповещать всех участников
Сообщение по умолчанию	Если выбрано, будет использовано сообщение по умолчанию (смотрите выше).
Тема	Тема пользовательского сообщения. Тема может содержать макросы.
Сообщение	Пользовательское сообщение. Сообщение может содержать макросы.

7.9.2.4 Операции обновления

Обзор

Операции обновления позволяют вам получать оповещения, когда проблемы [обновляются](#), например:

- добавлен комментарий
- выполнено подтверждение
- изменилась важность
- закрыт (вручную)

Операции обновления доступны в действиях с источником событий *Триггеры*.

В операциях обновления поддерживаются как сообщения, так и удаленные команды. Операции обновления не поддерживают эскалации - все операции назначаются на единственный шаг.

Настройка операции обновления

Для настройки операции обновления:

- Перейдите на вкладку *Операции обновления* в [настройках](#) действия
- Нажмите на *Новый* в блоке Операций
- Измените детали операции и нажмите на *Добавить*

Можно добавить несколько операций.

Атрибуты операции обновления:

Action Operations Recovery operations Update operations

Default subject: Updated problem: {EVENT.NAME}

Default message:

```
{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem at
{EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}:
{EVENT.UPDATE.MESSAGE}

Current problem status is {EVENT.STATUS}, acknowledged:
{EVENT.ACK.STATUS}
```

Operations:

Details
Notify all involved
Send message to user groups: Zabbix administrators via SMS

Operation details:

Operation type: Notify all involved
Default media type: - All -
Default message:
Update Cancel

* At least one operation, recovery operation or update operation must exist.

Add Cancel

Параметр	Описание
Тема по умолчанию	Тема сообщений по умолчанию для оповещений на обновление. Тема может содержать макросы .
Сообщение по умолчанию	Сообщение по умолчанию для оповещений на обновление. Сообщение может содержать макросы .
Операции	Детали операции на обновление. Для настройки новой операции на обновление, нажмите на <i>Новая</i> .
Детали операций	Этот блок используется для настройки деталей операции на обновление. Для операций на обновление доступны три типа операций: Отправлять сообщение - отправление сообщения на обновление конкретному пользователю, когда событие обновляется, например, подтверждается Удаленная команда - выполнение удаленной команды, когда событие обновляется, например, подтверждается Оповещать всех участников - отправка сообщения всем пользователям, которые ранее получали оповещения о появлении проблемы и/или на обновление события о проблеме. Человек, который выполнил обновление не получит оповещение о своем собственном обновлении. Если в нескольких типах операций задан один и тот же получатель с не изменёнными темой/сообщением по умолчанию, тогда дубликаты оповещений отправлены не будут.
Тип операции	Тип операции: отправлять сообщение

Параметр	Описание
<i>Отправлять группам пользователей</i>	Нажмите на <i>Добавить</i> для выбора групп пользователей, которым будет отправляться сообщение на обновление. Группа пользователей должна иметь по крайней мере права доступа на “чтение” узла сети, чтобы получить уведомления.
<i>Отправлять пользователям</i>	Нажмите на <i>Добавить</i> для выбора пользователей, которым будет отправляться сообщение на обновление. Пользователь должен иметь по крайней мере права доступа на “чтение” узла сети, чтобы получить уведомления.
<i>Отправлять только через Сообщение по умолчанию</i>	Отправка сообщения на обновление по всем заданным способам оповещений или только по выбранному. Если выбрано, будет использовано сообщение по умолчанию (смотрите выше).
<i>Тема</i>	Тема пользовательского сообщения. Тема может содержать макросы.
<i>Сообщение</i>	Пользовательское сообщение. Сообщение может содержать макросы.
<i>Список целей</i>	<p>Тип операции: удаленная команда</p> <p>Выберите цели для выполнения команды:</p> <p>Текущий узел сети - команда выполнится на узле сети триггера, который вызвал событие о проблеме. Эта опция не будет работать, если в выражении триггера имеется несколько узлов сети.</p> <p>Узел сети - выберите узел(ы) сети для выполнения команды.</p> <p>Группа узлов сети - выберите группу(ы) узлов сети для выполнения команды. Указав родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Таким образом команда будет выполнена на узлах сети из вложенных групп.</p> <p>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). Список целей не имеет смысла, если команда выполняется на стороне Zabbix сервера. В этом случае выбор нескольких целей приведет только к тому, что команда выполнится на сервере несколько раз.</p> <p>Обратите внимание, что в случае глобальных скриптов выбор цели также зависит от настройки <i>Группа узлов сети</i> в настройках глобального скрипта.</p> <p>Выберите тип команды:</p> <p>IPMI - выполнение IPMI команды</p> <p>Пользовательский скрипт - выполнение пользовательского набора команд.</p> <p>SSH - выполнение SSH команды</p> <p>Telnet - выполнение Telnet команды</p> <p>Глобальный скрипт - выполнение одного из глобальных скриптов определенных в <i>Администрирование → Скрипты</i>.</p> <p>Выполнение пользовательского скрипта на:</p> <p>Zabbix агент - скрипт будет выполнен на узле сети Zabbix агентом</p> <p>Zabbix сервер (прокси) - скрипт будет выполнен Zabbix сервером или прокси - в зависимости от того кем наблюдается узел сети сервером или прокси</p> <p>Zabbix сервер - скрипт будет выполнен только на стороне</p>
<i>Тип</i>	
<i>Выполнить на</i>	

Параметр	Описание
<i>Команды</i>	Zabbix сервера Для выполнения скриптов на стороне агента необходимо настроить разрешение выполнения удаленных команд с сервера. Это поле доступно, если как <i>Тип</i> выбран 'Пользовательский скрипт'. Ведите команду(ы). Поддерживаемые макросы будут раскрыты на основании выражения триггера, которое вызвало событие. Например, макрос узла сети будет раскрыт в узлы сети из выражения триггера (не в узлы сети из списка целей).
<i>Тип способа оповещения по умолчанию</i>	Тип операции: оповещать всех участников Пользователи, которые выполнили обновление проблемы, но не получали оповещения о проблеме, получат оповещения о дальнейших обновлениях с учетом выбранного способа оповещения по умолчанию - Email, Jabber или SMS. Это поле доступно начиная с Zabbix 3.4.2.
<i>Сообщение по умолчанию</i>	Если выбрано, будет использовано сообщение по умолчанию (смотрите выше).
<i>Тема</i>	Тема пользовательского сообщения. Тема может содержать макросы.
<i>Сообщение</i>	Пользовательское сообщение. Сообщение может содержать макросы.

7.9.2.5 Эскалации

Обзор

Используя эскалации, вы можете создавать пользовательские сценарии для отправки оповещений или выполнения удаленных команд.

С практической точки зрения эскалации означают, что:

- Пользователи могут быть информированы о новых проблемах немедленно
- Оповещения могут повторяться до решения проблемы
- Отправка оповещения может быть выполнена с задержкой
- Оповещения могут эскалироваться другой “более высокой” группе пользователей
- Удаленные команды могут быть выполнены незамедлительно или когда проблема не 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 действия](#). Эскалации поддерживаются только операциями на проблемы, не восстановления.

Различные аспекты поведения эскалаций

Давайте рассмотрим что произойдет при разных обстоятельствах, если действие содержит несколько шагов эскалаций.

Ситуация	Поведение
<p><i>Узел сети, о котором идет речь, переходит в состояние обслуживания после отправки первоначального оповещения о проблеме</i></p>	<p>В зависимости от опции <i>Приостановить операции для подавленных проблем в настройках</i> действия, все оставшиеся шаги эскалаций выполняются либо с задержкой, вызванной периодом обслуживания, или без задержки. Период обслуживания не отменяет операции.</p>
<p><i>Период времени, указанный в условии Период времени действия завершается после первоначальной отправки оповещения</i></p>	<p>Выполняются все оставшиеся шаги эскалации. Условие <i>Период времени</i> не может прервать операции; это условие имеет влияние в отношении действия началось/не началось, но не имеет влияния на операции.</p>
<p><i>Проблема началась в процессе обслуживания и продолжается (не исправлена) после окончания обслуживания</i></p>	<p>В зависимости от опции <i>Приостановить операции для подавленных проблем в настройках</i> действия, все шаги выполняются либо с момента завершения обслуживания или сразу же.</p>
<p><i>Проблема началась в процесса обслуживания без сбора данных и продолжается (не исправлена) после окончания обслуживания</i></p>	<p>Действие должно дождаться пока триггер не перейдет в состояние проблемы, до выполнения всех шагов эскалации.</p>
<p><i>Разные эскалации следуют в тесном порядке и перекрываются</i></p>	<p>Выполнение каждой новой эскалации заменяет предыдущую эскалацию, но как минимум один шаг эскалации обязательно выполняется из предыдущей эскалации. Такое поведение тесно связано с действиями на события, которые создаются на КАЖДОЕ вычисление проблемы триггера.</p>
<p><i>В процессе выполнения эскалации (например, во время отправки сообщения), основанном на любом типе событий:</i></p> <ul style="list-style-type: none">- действие деактивировано- событие удалено <p><i>На основе события на триггер:</i></p> <ul style="list-style-type: none">- триггер деактивирован или удален- узел сети или элемент данных деактивирован <p><i>На основе внутреннего события о триггере:</i></p> <ul style="list-style-type: none">- триггер деактивирован или удален <p><i>На основе внутреннего события о элементах данных/правилах низкоуровневого</i></p>	<p>Будет отправлено сообщение, которое уже в процессе отправки, затем будет отправлено еще одно сообщение эскалации. Последующие сообщения будут иметь следующий текст в начале тела сообщения: (<i>NOTE: Escalation cancelled</i>) с указанием причины (например, <i>NOTE: Escalation cancelled: action '<Action name>' disabled</i>). Таким образом получатель будет проинформирован о том, что эскалация отменена и дальнейшие шаги не будут выполнены. Это сообщение отправляется адресатам указанным в текущем шаге эскалации. Причина отмены также будет записана в файл журнала сервера (начиная с уровня отладки 3=Предупреждения).</p>

Ситуация

обнаружения:

- элемент данных деактивирован или удален
- узел сети деактивирован

В процессе выполнения
эскалации (например, во
время отправки
сообщения) удалено
действие

Поведение

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:

- Задайте *Длительность шага операции по умолчанию* равным '30m' (30 минут) на вкладке *Операции*
- Укажите шаги эскалаций С '1' До '5'
- Выберите группу 'MySQL администраторы' получателями сообщения

Action Operations Recovery operations Update operations

* Default operation step duration: 30m

Default subject: Problem: {EVENT.NAME}

Default message:

```
Problem started at {EVENT.TIME} on {EVENT.DATE}
Problem name: {TRIGGER.NAME}
Host: {HOST.NAME}
Severity: {EVENT.SEVERITY}

Original problem ID: {EVENT.ID}
{TRIGGER.URL}
```

Pause operations while in maintenance:

Operations	Steps	Details	Start in	Duration
	1 - 5	Send message to user groups: MySQL Administrators via all media	Immediately	Default
	New			

* At least one operation, recovery operation or update operation must exist.

Все обязательные поля ввода отмечены красной звёздочкой.

Оповещения будут отправлены в 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.



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:

- Задайте *Длительность шага операции по умолчанию* равным '10h' секунд (10 часов) на вкладке *Операции*
- Укажите шаги эскалации С '2' До '2'

Action Operations Recovery operations Update operations

* Default operation step duration	10h															
Default subject	Problem: {EVENT.NAME}															
Default message	Problem started at {EVENT.TIME} on {EVENT.DATE} Problem name: {TRIGGER.NAME} Host: {HOST.NAME} Severity: {EVENT.SEVERITY} Original problem ID: {EVENT.ID} {TRIGGER.URL}															
Pause operations while in maintenance	<input checked="" type="checkbox"/>															
<table border="1"> <thead> <tr> <th>Operations</th> <th>Steps</th> <th>Details</th> <th>Start in</th> <th>Duration</th> </tr> </thead> <tbody> <tr> <td></td> <td>2</td> <td>Send message to user groups: Managers via Email</td> <td>10:00:00</td> <td>Default</td> </tr> <tr> <td></td> <td></td> <td>New</td> <td></td> <td></td> </tr> </tbody> </table>		Operations	Steps	Details	Start in	Duration		2	Send message to user groups: Managers via Email	10:00:00	Default			New		
Operations	Steps	Details	Start in	Duration												
	2	Send message to user groups: Managers via Email	10:00:00	Default												
		New														

* At least one operation, recovery operation or update operation must exist.

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 более чем 10 часов'.

Пример 3

Эскалирование проблемы Руководству.

В первом примере мы настраивали периодическую отправку сообщений Администраторам MySQL. В этом 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 еще не подтверждена, предположительно, никто не работает над ней.

Action Operations Recovery operations Update operations

* Default operation step duration

Default subject

Default message

Pause operations while in maintenance

Operations	Steps Details	Start in	Duration
1 - 0	Send message to user groups: MySQL Administrators via Email	Immediately	Default
5	Send message to users: Database manager (Mr Swift) via all media	02:00:00	Default

Operation details

Steps - (0 - infinitely)

Step duration (0 - use action default)

Operation type

* At least one user or user group must be selected.

Send to User groups	User group	Action
	Add	

Send to Users	User	Action
	Database manager (Mr Swift)	Remove
	Add	

Send only to

Default message

Subject

Message

Conditions

Label	Name	Action
A	Event acknowledged = Not Ack	Remove
	New	

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 произойдет, если проблема присутствует через 2:30 часов и еще не была подтверждена.

Если проблема все еще существует, спустя еще 30 минут Zabbix отправит сообщение всем гостевым пользователям.

Если и это не поможет, спустя еще час Zabbix перезагрузит сервер с этой базой данных MySQL (вторая удаленная команда), используя IPMI команды.

Action Operations Recovery operations Update operations

* Default operation step duration

Default subject

Default message

Pause operations while in maintenance

Operations

Steps	Details	Start in	Duration
1 - 0	Send message to user groups: MySQL Administrators via Email	Immediately	Default
5	Send message to users: Database manager (Mr Swift) via all media	02:00:00	Default
6	Run remote commands on current host	02:30:00	Default
7	Send message to user groups: Guests via all media	03:00:00	Default
9	Run remote commands on current host	04:00:00	Default

New

* At least one operation, recovery operation or update operation must exist.

Пример 5

Эскалация с несколькими операциями назначенными на один шаг и с использованием пользовательских интервалов. Длительность шага операции по умолчанию равна 30 минутам.

Action Operations Recovery operations Update operations

* Default operation step duration

Default subject

Default message

Pause operations while in maintenance

Operations

Steps	Details	Start in	Duration
1 - 4	Send message to user groups: MySQL Administrators via Email	Immediately	Default
5 - 6	Send message to users: Database manager (Mr Swift) via all media	02:00:00	1h
5 - 7	Send message to user groups: Zabbix administrators via Email	02:00:00	10m
11	Send message to user groups: Guests via Email	04:00:00	Default

New

* At least one operation, recovery operation or update operation must exist.

Оповещения будут отправлены в следующем порядке:

- MySQL администраторам в 0:00, 0:30, 1:00, 1:30 после начала проблемы
- Менеджеру баз данных в 2:00 и 2:10 (в этом случае более короткая пользовательская длительность менее 10 минут перекрывает длительную пользовательскую длительность равную 1 часу)

- Zabbix администраторам в 2:00, 2:10, 2:20, 2:30 после начала проблемы (задана пользовательская длительность шага равная 10 минутам)
- Гостевым пользователям в 4:00 часа после начала проблемы (интервал по умолчанию равный 30 минутам возвращается между шагами 8 и 11)

7.9.3 Получение оповещений о неподдерживаемых элементах данных

Обзор

Получение оповещений о неподдерживаемых элементах данных - новая функция Zabbix 2.2.

Это часть нового концепта внутренних событий в Zabbix, позволяющая пользователям получать оповещения о таких случаях. Внутреннее событие отражает 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 низкоуровневого правила обнаружения из состояния “активировано” в состояние “не поддерживается” (и обратно)

Этот раздел освещает как **получать оповещения**, когда элемент данных становится не поддерживаемым.

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 e-mail, SMS или Jabber, для использования в оповещениях. Для выполнения этой задачи обратитесь к соответствующим разделам документации.



При оповещении о внутренних событиях используется важность по умолчанию ('Не классифицировано'), поэтому оставьте эту важность отмеченную при настройке [оповещения пользователя](#), если вы хотите получать оповещения о внутренних событиях.

Шаг 2

Перейдите в *Настройка* → *Действия* и выберите *Внутренняя*, как источников событий. Нажмите на *Создать действие* в правом верхнем углу для открытия диалога настройки действий.

Шаг 3

На вкладке **Действие** введите имя этого действия. Затем выберите *Тип события* в блоке *Новое условие* и выберите значением *Элемент данных* в “неподдерживаемом” состоянии.

Actions

Action Operations Recovery operations

* Name Report not supported items

Conditions

Label	Name	Action
A	Event type equals Item in "not supported" state	Remove

New condition

Event type equals

Add

Enabled

* At least one operation or recovery operation must exist.

Add Cancel

The screenshot shows a configuration interface for 'Actions'. It has tabs for 'Action', 'Operations', and 'Recovery operations'. Under 'Action', there's a field for 'Name' with the value 'Report not supported items'. A 'Conditions' section lists a single rule: 'A' where 'Event type equals Item in "not supported" state'. There's a 'Remove' link next to it. Below this, there's a 'New condition' section with a dropdown menu open, showing three options: 'Item in "not supported" state' (which is selected), 'Low-level discovery rule in "not supported" state', and 'Trigger in "unknown" state'. At the bottom, there's a note: '* At least one operation or recovery operation must exist.' and two buttons: 'Add' and 'Cancel'.

Не забудьте нажать на *Добавить* для добавления условия в список блока *Условия*.

Шаг 4

На вкладке **Операции** введите тему/содержимое сообщения о проблеме.

Нажмите на *Новая* в блоке *Операции* и выберите несколько получателей сообщения (группы пользователей/пользователей) и типы оповещений (или 'Все') при помощи которых необходимо доставлять оповещения.

Actions

Action Operations Recovery operations

Default operation step duration 3600 (minimum 60 seconds)

Default subject {ITEM.STATE}: {HOST.NAME}: {ITEM.NAME}

Default message Host: {HOST.NAME}
Item: {ITEM.NAME}
Item key: {ITEM.KEY}
State: {ITEM.STATE}
Problem event: {EVENT.ID}
So far: {ESC.HISTORY}

Operations Steps Details S
1 - 2 Send message to user groups: Zabbix administrators via Email In

Operation details Steps 1 - 2 (0 - infinitely)
Step duration 300 (minimum 60 seconds, 0 - use action duration)
Operation type Send message
Send to User groups User group Action
Zabbix administrators Remove
Add
Send to Users User Action
Add
Send only to Email
Default message
Update Cancel

Нажмите на *Добавить* в блоке *Детали операции* для добавления операции в список блока *Операции*.

Если вы желаете получать более одного оповещения, задайте длительность шага операции (интервал между отправляемыми сообщениями) и добавьте другую операцию.

Шаг 5

Вкладка **Операции восстановления** позволяют настраивать оповещения восстановления, когда элемент данных вернётся обратно в нормальное состояние.

Введите тему/содержимое сообщения о восстановлении.

Нажмите на *Новая* в блоке *Операции* и выберите несколько получателей сообщения (группы пользователей/пользователей) и типы оповещений (или 'Все') при помощи которых необходимо доставлять оповещения.

Actions

The screenshot shows the 'Operations' tab of the Zabbix Actions configuration dialog. At the top, there are tabs for 'Action', 'Operations', and 'Recovery operations', with 'Operations' being the active tab. Below the tabs, there are two main sections: 'Default subject' and 'Default message'. The 'Default subject' section contains the placeholder {{ITEM.STATE}}: {{HOST.NAME}}: {{ITEM.NAME}}. The 'Default message' section contains the following template:
Host: {{HOST.NAME}}
Item: {{ITEM.NAME}}
Item key: {{ITEM.KEY}}
State: {{ITEM.STATE}}
Recovery event: {{EVENT.RECOVERY.ID}}

Below these sections, there are tabs for 'Operations' and 'Details'. The 'Details' tab is selected, showing the text: 'Notify all who received any messages regarding the problem before'. Under the 'Operations' tab, there are settings for 'Operation details': 'Operation type' set to 'Send recovery message', and 'Default message' checked. There are also 'Update' and 'Cancel' buttons. At the bottom of the dialog are 'Add' and 'Cancel' buttons.

Нажмите на *Добавить* в блоке *Детали операции* для добавления операции в список блока *Операции*.

Шаг 6

Когда закончите, нажмите кнопку **Добавить** под диалогом.

И это все, вы это сделали! Сейчас вы можете пождать получения вашего первого оповещения от Zabbix, если какой-либо элемент данных станет не поддерживаемым.

7.10 Макросы

Обзор

Zabbix поддерживает некоторое количество макросов, которые можно использовать в различных ситуациях. Макросами являются переменные, которые имеют специальный синтаксис:

{МАКРОС}

Макросы раскрываются в требуемое значение в зависимости от контекста.

Эффективное использование макросов позволяет сохранить время и делает настройку Zabbix более прозрачной.

Самое типичное место использования макросов в шаблонах. Таким образом, триггер из шаблона можно назвать как “Слишком высокая загрузка процессора на {{HOST.NAME}}”. Когда шаблон применяется к узлу сети, такому как Zabbix сервер, имя будет раскрыто в “Слишком высокая загрузка процессора на Zabbix сервер”, когда триггер отобразится в разделе Мониторинга.

Макросы можно использовать в параметрах ключей элементов данных. Макрос можно использовать только как 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.

Смотрите также:

- полный список [поддерживаемых макросов](#)
- [функции](#) макросов
- информацию о том, как настроить [пользовательские макросы](#).

7.10.1 Функции макросов

Обзор

Функции макросов дают возможность индивидуализации значений [макросов](#).

Периодически макрос может раскрыться в значение с которым не всегда легко работать. Значение может быть длинным или содержать определённую подстроку, которая вас интересует и которую вы бы хотели извлечь. Это та задача, где функции макросов могут быть полезны.

Синтаксис функции макросов:

{<макрос>. <функция>(<параметры>)}

где:

- <макрос> - настраиваемый макрос (например {ITEM.VALUE} или {#LLDMAKPOC})
- <функция> - применяемая функция
- <параметры> - список параметров функции разделенных запятыми. Параметры необходимо заключать в кавычки, если они начинаются с (пробел), " или содержат), , .

Например:

```
 {{ITEM.VALUE}.regsub(шаблон, вывод)}  
 {{#LLDMAKPOC}.regsub(шаблон, вывод)}
```

Поддерживаемые функции макросов

Описание	ФУНКЦИЯ	Параметры	Поддерживается для
Извлечение подстроки при помощи совпадения с регулярным выражением (с учетом регистра).	regsub (<шаблон>,<вывод>)	шаблон - регулярное выражение для поиска совпадения вывод - опции вывода. Для захвата групп поддерживаются местозаменители \1 - \9 (placeholder). \0 вернёт соответствующий текст.	{ITEM.VALUE} {ITEM.LASTVALUE} Макросы низкоуровневого обнаружения (за исключением фильтра низкоуровневого обнаружения)
Извлечение подстроки при помощи совпадения с регулярным выражением (без учета регистра).	iregsub (<шаблон>,<вывод>)	шаблон - регулярное выражение для поиска совпадения вывод - опции вывода. Для захвата групп поддерживаются местозаменители \1 - \9 (placeholder). \0 вернёт соответствующий текст.	{ITEM.VALUE} {ITEM.LASTVALUE} Макросы низкоуровневого обнаружения (за исключением фильтра низкоуровневого обнаружения)

Если функция используется в [поддерживаемых местах](#), но применён макрос, который не поддерживает функции макросов, тогда макрос раскроется в 'НЕИЗВЕСТНО'.

Если шаблон не соответствует регулярному выражению, тогда макрос раскрывается в 'НЕИЗВЕСТНО' (за исключением макросов низкоуровневого обнаружения, где в этом случае функция будет проигнорирована и макрос останется нераскрытым)

Если функция макросов применяется в макрое в местах, которые не поддерживают функции макросов, тогда функция будет игнорироваться.

Примеры

Способы в которых можно использовать функции макросов для индивидуализации значений проиллюстрированы в следующих примерах, содержащие строки журналов как полученное значение:

Полученное значение	Макрос	Выход
123Log line	<code>{{ITEM.VALUE}}.regsub(^[0-9]+, Problem)}</code>	Problem
123 Log line	<code>{{ITEM.VALUE}}.regsub("^([0-9]+)", "Problem")}</code>	Problem
123 Log line	<code>{{ITEM.VALUE}}.regsub("^([0-9]+)", Problem ID: \1)}</code>	Problem ID: 123
Log line	<code>{{ITEM.VALUE}}.regsub(".*", "Problem ID: \1")}</code>	Problem ID:
MySQL crashed errno 123	<code>{{ITEM.VALUE}}.regsub("^([A-Z]+).*([0-9]+)", " Problem ID: \1\2 ")}</code>	Problem ID: MySQL_123
123 Log line	<code>{{ITEM.VALUE}}.regsub("([1-9]+", "Problem ID: \1")}</code>	*НЕИЗВЕСТНО* (некорректное регулярное выражение)
имяклиента_1	<code>{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)</code>	имяклиента
имяклиента_1	<code>{#IFALIAS}.regsub("(.*)_([0-9]+)", \2)</code>	
имяклиента_1	<code>{#IFALIAS}.regsub("(.*)_([0-9]+", \1) {#IFALIAS}.regsub("(.*)_([0-9]+", \1)}</code>	(некорректное регулярное выражение)

7.10.2 Пользовательские макросы

Обзор

В дополнение к макросам [поддерживающимся](#) из коробки в Zabbix также для большей гибкости поддерживаются пользовательские макросы.

Пользовательские макросы можно определить на глобальном, уровне шаблона и уровне узла сети. Такие макросы имеют специальный синтаксис:

`{$MAKROC}`

Zabbix раскрывает макросы в соответствии со следующей очередностью:

1. макрос, назначенный узлу сети (проверяется в первую очередь)
2. макрос, назначенный на первый уровень шаблонов у узла сети (т.е. шаблоны присоединенные напрямую к узлу сети), которые отсортированы по ID шаблона
3. макрос, назначенный на второй уровень шаблонов у узла сети, которые отсортированы по ID шаблона

4. макрос назначенный на третий уровень шаблонов у узла сети, которые отсортированы 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, присоединенных к этому узлу сети, с учетом увеличивающейся глубины шаблонов. Если макрос все еще не найден, то будет использован макрос глобального уровня, если он существует конечно.

В случае, если Zabbix не удалось найти макрос, макрос не будет раскрыт.

Пользовательские макросы можно использовать в:

- именах элементов данных
- в параметрах ключей элементов данных
- в интервалах обновления и гибких интервалах элементов данных
- именах и описаниях триггеров
- в параметрах и константах выражений триггеров (смотрите [примеры](#))
- многих других местах (смотрите [Макросы поддерживаемые по назначению](#))

Общие случаи использования глобальных макросов и макросов уровня узлов сети

- используйте глобальные макросы в нескольких местах; затем измените значения макроса и изменения конфигурации применяются во всех местах за “одно нажатие”
- получение преимуществ шаблонов со специфичными для узлов сети атрибутами: пароли, номера портов, имена файлов, регулярные выражения, и т.д.

Настройка

Для добавления пользовательских макросов, перейдите в соответствующие разделы в веб-интерфейсе:

- для глобальных макросов, посетите *Администрирование* → *Общие* → *Макросы*
- для макросов на уровне узлов сети и шаблонов, откройте свойства узла сети или шаблона и найдите вкладку *Макросы*



Если пользовательский макрос используется в элементах данных или триггерах шаблона, то предлагается добавить этот макрос к шаблону, даже если он задан на глобальном уровне. Таким образом при экспорте шаблона в XML и импорте его в другую систему элементы данных и триггеры продолжат работать, как это и ожидалось.

В именах макросов допускаются следующие символы: **A-Z , 0-9 , _ , .**

Примеры

Пример 1

Использование макроса 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”:

```
{ca_001:system.cpu.load[,avg1].last()}>{$MAX_CPULOAD}
```

Такой триггер можно создать у шаблона, он не будет изменен у отдельных узлов сети.



Если вы хотите использовать количество значений в качестве параметра функции (например, `max(#3)`), добавьте символ решетки # непосредственно в значение макроса, например вот так: `SOME_PERIOD` ⇒ `#3`

Пример 3

Использование двух макросов в триггере “Загрузка CPU слишком высокая”:

```
{ca_001:system.cpu.load[,avg1].min({$CPULOAD_PERIOD})}>{$MAX_CPULOAD}
```

Заметьте, что макрос можно использовать в качестве параметра в функциях триггеров, в этом примере в функции `min()`.



Пользовательские макросы будут раскрыты в триггерах, если макросы используются в параметрах или как константы. Они НЕ будут раскрыты, если используются как ссылка на функции, имена узлов сети, ключи элементов данных или операторы.

Пример 4

Синхронизация условия недоступности агентов с интервалом обновления элемента данных:

- задайте макрос `{$INTERVAL}` и используйте его в интервале обновления элемента данных;
- используйте `{$INTERVAL}` параметром в триггере на недоступность агента:

```
{ca_001:agent.ping.nodata({$INTERVAL})}=1
```

Пример 5

Централизация конфигурации рабочего времени:

- создайте глобальный макрос `{$WORKING_HOURS}` равный `1-5, 09:00-18:00`;
- используйте его в *Администрирование* → *Общие* → *Рабочее время*;
- используйте его в *Пользователь* → *Оповещения* → *Когда активно*;
- используйте его для настройки более частого опроса элементов данных в течении рабочего времени:

Update interval	<code>{\$LONG_INTERVAL}</code>		
Custom intervals	Type	Interval	Period
	Flexible	<code>{\$SHORT_INTERVAL}</code>	<code>{\$WORKING_HOURS}</code>

- используйте его в условии действия *Период времени*;
- отредактируйте рабочее время в *Администрирование* → *Общие* → *Макросы*, если требуется.

Контекст пользовательских макросов

В пользовательских макросах можно использовать optionalный контекст, позволяющий переопределять значение по умолчанию на значение с учётом конкретной ситуации.

Пользовательские макросы с контекстом имеют схожий синтаксис:

```
{$МАКРОС:контекст}
```

Контекстом макроса является простое текстовое значение. Общим случаем использования контекстов в макросах будет использование [значения низкоуровневого макроса](#) в качестве контекста пользовательских макросов. Например, можно определить прототип триггера для обнаружения

примонтированных файловых систем с использованием разных порогов малого свободного места в зависимости от точек монтирования или типов файловых систем.

В контекстах макроса поддерживаются только макросы низкоуровневого обнаружения. Любые другие макросы игнорируются и обрабатываются как текстовые значения.

Технически, контекст макроса задается 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 начинается с " символа. Кавычки внутри заключенном в кавычки контексте необходимо экранировать при помощи \ символа. Сам символ \ не экранируется, что означает, что невозможно задать заключенный 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 {\$MACRO:A } .
- Все пробелы до кавычек в начале и после кавычек игнорируются, но все пробелы внутри кавычек не игнорируются. Макросы {\$MACRO:"A"}, {\$MACRO: "A"}, {\$MACRO :"A" } и {\$MACRO: "A" } одинаковы, но макрос {\$MACRO:"A"} и {\$MACRO:" A "} не одинаковы.

Следующие макросы одинаковы, 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.

Когда макросы с контекстами обрабатываются, Zabbix ищет макрос со своим контекстом вычисляется. Если макрос с этим контекстом не задан на узле сети или присоединенных шаблонах, и не задан как глобальный макрос с контекстом, тогда выполняется поиск макроса без контекста.

Смотрите [пример использования](#) контекста макроса в прототипе триггера о свободном месте на диске и примите во внимание пункт об ограничениях.

7.10.3 Макросы низкоуровневых обнаружений

Обзор

Существует тип макроса, который используется в функции [низкоуровневого обнаружения](#) (LLD): {#МАКРОС}

Это такой макрос, который используется в правиле LLD и возвращает реальные значения имен файловых систем, сетевых интерфейсов и SNMP OID-ов.

Эти макросы можно использовать для создания *прототипов* элементов данных, триггеров и графиков. Затем, когда обнаруживаются реальные файловые системы, сетевые интерфейсы и т.п., такие макросы заменяются реальными значениями и становятся основой для создания реальных элементов данных, триггеров и графиков.

Также эти макросы можно использовать для создания *прототипов* узлов сети и групп узлов сети в [обнаружении](#) виртуальных машин.

Некоторые макросы низкоуровневого обнаружения “заранее добавлены” в функционал низкоуровневого обнаружения в Zabbix - {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. Однако, не обязательно придерживаться этих имен при создании [пользовательского](#)

низкоуровневого правила обнаружения. В этом случае вы можете использовать любое другое LLD имя макроса и ссылаться по этому имени.

Поддерживаемые места

LLD макросы можно использовать:

- в фильтре правила низкоуровневого обнаружения
- в прототипах элементов данных в
 - именах
 - параметрах ключей
 - единицах измерения
 - интервалах обновления
 - периодах хранения истории
 - периодах хранения динамики изменений
 - SNMP OID'ах
 - полях IPMI датчиков
 - формулах вычисляемых элементов данных
 - SSH и Telnet скриптах
 - SQL запросах монитора баз данных
 - полях endpoint JMX элементов данных
 - описаниях
- также в начиная с Zabbix 4.0:
 - шагах предобработки значений элементов данных
 - поле URL HTTP агента
 - поле HTTP поля запроса HTTP агента
 - поле запроса тела HTTP агента
 - поле требуемых кодов состояний HTTP агента
 - полях ключей и значений заголовков HTTP агента
 - поле имени пользователя аутентификации HTTP агента
 - поле пароля аутентификации HTTP агента
 - поле HTTP прокси HTTP агента
 - поле файла SSL сертификата HTTP агента
 - поле файла SSL ключа HTTP агента
 - поле пароля к SSL ключу HTTP агента
 - поле HTTP времени ожидания HTTP агента
- в прототипах триггеров в
 - именах
 - выражениях (только в константах и параметрах функций)
 - URL'ах
 - описаниях
 - именах тегов событий и значениях (за исключением параметров функций макросов)
- в прототипах графиков в
 - именах
- в прототипах узлов сети в
 - именах
 - видимых именах
 - именах прототипов групп узлов сети
 - (смотри [полный список](#))

Во всех перечисленных местах можно использовать LLD макросы в [контекстах пользовательских макросов](#).

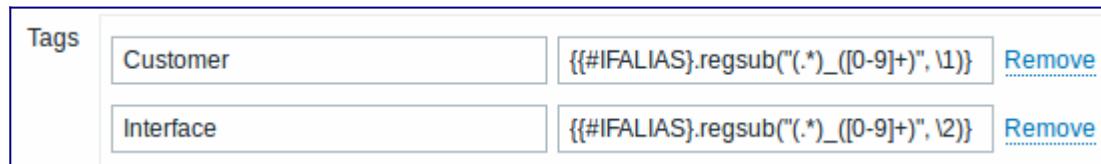
Использование функций макросов

Функции макросов поддерживаются с макросами низкоуровневого обнаружения (за исключением фильтра правила низкоуровневого обнаружения), которые позволяют извлекать некоторую часть значения макроса используя регулярное выражение.

Например, вы можете захотеть извлечь имя клиента и номер интерфейса из следующего макроса LLD для назначения тегов событий:

```
{#IFALIAS}=имяклиента_1
```

Чтобы это сделать, можно использовать функцию макроса `regsub` с макросом в поле значения тега событий прототипа триггеров:



Для получения более подробных сведений о синтаксисе функций макросов смотрите: [Функции макросов](#)

Функции макросов в макросах низкоуровневого обнаружения поддерживаются начиная с Zabbix 4.0.

7.11 Пользователи и группы пользователей

Обзор

Все пользователи в Zabbix работают с системой Zabbix через веб-интерфейс. Каждому пользователю в Zabbix присвоен 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, за исключением, если они 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
- наблюдаемым узлам сети в группах узлов сети

Начальная установка Zabbix содержит двух предустановленных пользователей - 'Admin' и 'guest'. Пользователь 'guest' используется для неавторизованных пользователей. Перед входом в систему под 'Admin', вы будете 'guest'. Перейти к [настройке пользователя](#) в Zabbix.

7.11.1 Настройка пользователя

Обзор

Для настройки пользователя:

- Перейдите в Администрирование → Пользователи
- Нажмите на Создать пользователя (или на имя пользователя для редактирования уже существующего пользователя)

- Измените в диалоге атрибуты пользователя

Общие атрибуты

Вкладка *Пользователь* состоит из общих атрибутов пользователя:

The screenshot shows a configuration dialog for a new user. At the top, there are three tabs: 'User' (selected), 'Media', and 'Permissions'. Below the tabs are several input fields with validation rules indicated by red asterisks (*):

- Alias**: Admin (no validation)
- Name**: Zabbix (no validation)
- Surname**: Administrator (no validation)
- Groups**: Zabbix administrators (with a delete icon) - A dropdown menu is open, showing 'Zabbix administrators' and a search bar below it. A 'Select' button is to the right.
- Password**: (no validation)
- Password (once again)**: (no validation)
- Language**: English (en_GB) - A dropdown menu is open.
- Theme**: System default - A dropdown menu is open.
- Auto-login**:
- Auto-logout**: 15m - A dropdown menu is open.
- Refresh**: 30s
- Rows per page**: 50
- URL (after login)**:

At the bottom of the dialog are two buttons: 'Add' (blue) and 'Cancel' (light blue).

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Псевдоним	Уникальное имя пользователя, используется как логин.
Имя	Имя пользователя (опционально). Если поле заполнено, имя отображается в информации подтверждения и сообщении-уведомлении оповещения.
Фамилия	Фамилия пользователя (опционально). Если поле заполнено, фамилия отображается в информации подтверждения и сообщении-уведомлении оповещения.
Группы	Выбор групп пользователей , к которым пользователь принадлежит. Начиная с Zabbix 3.4.3 это поле имеет функцию автодополнения, таким образом после начала ввода имени группы пользователей вам будет предложен список соответствующих групп пользователей в выпадающем меню. Прокрутите вниз, чтобы выбрать необходимую группу. Кроме того, для добавления групп нажмите на Выбор . Нажмите на 'x', чтобы удалить выбранную группу. Соответствие группам пользователей определяет к каким группам узлов сети и узлам сети пользователь имеет доступ .
Пароль	Два поля для ввода пароля пользователя. При наличии пароля отображается кнопка Пароль , при нажатии на которую появятся поля пароля.
Язык	Язык веб-интерфейса Zabbix. Для работы перевода веб-интерфейса необходимо установить расширение php gettext.
Тема	Определяет, как веб-интерфейс будет выглядеть: Системная по умолчанию - использование системных настроек по умолчанию Синяя - стандартная синяя схема Темная - альтернативная темная тема

Параметр	Описание
	Высококонтрастная светлая - светлая тема с высоким контрастом Высококонтрастная тёмная - тёмная тема с высоким контрастом
<i>Авто-вход</i>	Отметьте, если хотите, чтобы Zabbix запомнил пользователя и выполнял вход в систему автоматически в течении 30 дней. Для этого используются cookies браузера.
	При выборе этой опции пользователь будет выходить из системы автоматически, после указанного количества секунд (минимальное значение 90 секунд, максимальное 1 день).
<i>Авто-выход</i>	Поддерживаются суффиксы времени , например, 90s, 5m, 2h, 1d. Обратите внимание на случаи когда эта опция не будет работать: <ul style="list-style-type: none"> * Если активирована глобальная опция конфигурации “Показывать предупреждение, если Zabbix сервер недоступен” и веб-интерфейс Zabbix остается открытым; * Когда страницы меню Мониторинга выполняют фоновое обновление информации; * Если при входе в систему выбрана опция <i>Запоминать меня на 30 дней</i>.
<i>Обновление</i>	Выберите частоту обновления графиков, комплексных экранов, простых тестовых данных и прочего. Можно указать значение равное 0, тогда обновление будет отключено.
<i>Строк на странице</i>	Вы можете задать, как много строк на странице будет отображаться в списках.
<i>URL (после входа в систему)</i>	Zabbix перенаправит пользователя на указанный URL после успешного входа в систему, например, на страницу Проблем.

Оповещение пользователя

Вкладка *Оповещение* содержит список всех оповещений заданных для этого пользователя.

Оповещения используются для отправки оповещений. Нажмите на *Добавить* для назначения оповещения пользователю.

Смотрите раздел [Способы оповещений](#) для получения подробных сведений по настройке способов оповещения.

Права доступа

Вкладка *Права доступа* содержит информацию о:

- тип пользователя (Zabbix Пользователь, Zabbix Администратор, Zabbix Супер-Администратор). Пользователи не могут менять тип самим себе.
- группы узлов сети к которым пользователь имеет доступ. Пользователи 'Zabbix Пользователь' и 'Zabbix Администратор' не имеют прав доступа к каким-либо группам узлов сети или узлам сети по умолчанию. Для получения прав доступа им необходимо находиться в группах пользователей с соответствующими группами узлов сети и узлами сети.

Смотрите страницу [Права доступа пользователей](#) для получения подробных сведений.

7.11.2 Права доступа

Обзор

Вы можете разграничить права доступа пользователей в Zabbix, задав соответствующий тип пользователя и затем включив непривилегированных пользователей в группы пользователей, которые имеют доступ к данным группам узлов сети.

Тип пользователя

Тип пользователя определяет уровень прав к административным меню и уровень прав к данным группам узлов сети.

Тип пользователя	Описание
Zabbix Пользователь	Пользователь имеет доступ к меню Мониторинг. По умолчанию пользователь не имеет прав доступа к каким-либо ресурсам. Права доступа на группу узлов сети должны быть заданы явно.
Zabbix Администратор	Пользователь имеет доступ в меню Мониторинг и в меню Настройка. По умолчанию пользователь не имеет прав доступа к каким-либо ресурсам. Права доступа на группу узлов сети должны быть заданы явно.
Zabbix Супер-Администратор	Пользователь имеет доступ ко всему: Мониторинг, Настройка и Администрирование. Пользователь имеет права Чтения-Записи ко всем группам узлов сети. Запрет доступа к каким-либо группам узлов сети не влияет на права доступа пользователя.

Права доступа к группам узлов сети

Доступ к любым данным узлов сети в Zabbix гарантирован [группам пользователей](#) только на уровне группы узлов сети.

Это означает, что отдельному пользователю не может быть напрямую назначено разрешение на доступ к 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, которая содержит необходимый узел сети.

7.11.3 Группы пользователей

Обзор

Группы пользователей позволяют группировать пользователей для организационных целей и для назначения прав доступа к данным. Права доступа к наблюдаемым данным групп узлов сети начинаются на группы пользователей, не индивидуально пользователям.

Такой подход может часто иметь смысл для разделения какая информация доступна одной группе пользователей и какая - другой. Это может быть достигнуто путем группировки пользователей и затем назначения различных прав доступа к группам узлов сети.

Пользователь может входить в любое количество групп.

Настройка

Для настройки группы пользователей:

- Перейдите в *Администрирование* → *Группы пользователей*
- Нажмите на *Создать группу* (или на имени группы для редактирования существующей группы)
- Измените в диалоге атрибуты группы

Вкладка **Группа пользователей** содержит общие атрибуты группы:

User groups

User group Permissions Tag filter

* Group name: Security specialists

Users: Admin (Zabbix Administrator)

Frontend access: System default

Enabled:

Debug mode:

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Описание

Имя группы	Уникальное имя группы.
Пользователи	Для добавления пользователей в группу нажмите на кнопку <i>Выбор</i> . Каким образом пользователи этой группы проходят аутентификацию. Системная по умолчанию - использование метода аутентификации по умолчанию (задаётся глобально) Внутренняя - использование внутренней аутентификации Zabbix (даже, если LDAP аутентификация используется глобально). Игнорируется, если глобально используется по умолчанию HTTP аутентификация.
Доступ к веб-интерфейсу	LDAP - использование LDAP аутентификации (даже, если внутренняя аутентификация используется глобально). Игнорируется, если глобально используется HTTP аутентификация. Deactivated - доступ к веб-интерфейсу Zabbix запрещен для этой группы Состояние членов группы: Активировано - пользователи активны Деактивировано - пользователи деактивированы
Активировано	
Режим отладки	Отметьте эту опцию для активации режима отладки для пользователей.

Вкладка **Права доступа** позволяет вам управлять доступом групп пользователей к данным группам узлов сети (и таким образом к узлу сети):

Permissions Tag filter

Permissions	Host group	Permissions
All groups		None
Discovered hosts	<input type="button" value="Read-write"/> <input type="button" value="Read"/> <input type="button" value="Deny"/> <input type="button" value="None"/>	
Hypervisors	<input type="button" value="Read-write"/> <input type="button" value="Read"/> <input type="button" value="Deny"/> <input type="button" value="None"/>	
Linux servers	<input type="button" value="Read-write"/> <input type="button" value="Read"/> <input type="button" value="Deny"/> <input type="button" value="None"/>	
Templates (including subgroups)	<input type="button" value="Read-write"/> <input type="button" value="Read"/> <input type="button" value="Deny"/> <input type="button" value="None"/>	
Templates/Servers Hardware	<input type="button" value="Read-write"/> <input type="button" value="Read"/> <input type="button" value="Deny"/> <input type="button" value="None"/>	
Templates/Virtualization	<input type="button" value="Read-write"/> <input type="button" value="Read"/> <input type="button" value="Deny"/> <input type="button" value="None"/>	

Include subgroups

Текущие права доступа к группам узлов сети отображаются в блоке *Права доступа*.

Если текущие права доступа на группу узлов сети наследуются всеми вложенными группами узлов сети, это будет обозначено текстом *включая подгруппы* в круглых скобках после имени группы узлов сети.

Вы можете изменить уровень доступа к группе узлов сети:

- **Чтение-запись** - чтение-запись доступ к группе узлов сети;
- **Чтение** - доступ только на чтение группы узлов сети;
- **Запрещен** - доступ к группе узлов сети запрещен;
- **Нет** - права доступа не заданы.

Используйте поле выбора внизу, чтобы выбрать группы узлов сети и уровень доступа к ним (обратите внимание, что выбор *Нет* удалит группу узлов сети из списка, если группа узлов сети уже в этом списке). Если вы желаете включить вложенные группы узлов сети, отметьте *Включая подгруппы*. Это поле имеет функцию авто-дополнения, начните набирать и в нём появится выпадающий список совпадающих групп узлов сети. Если вы желаете увидеть все группы узлов сети, нажмите на *Выбрать*.

Обратите внимание, что для Zabbix Супер Администратор пользователей возможно в [настройках](#) группы узлов сети принудительно задать такой же уровень прав доступа к вложенным группам узлов сети, что и к родительской группе узлов сети.

Вкладка **Фильтр тегов** позволяет вам задавать права доступа группам пользователей на основе тегов, чтобы видеть проблемы отфильтрованные по имени тега и значению этого тега:

Permissions	Host group	Tags	Action
	Templates/Database	Service: MySQL	Remove

[Select](#) [tag](#) [value](#)

[Include subgroups](#)

[Add](#) [Update](#) [Delete](#) [Cancel](#)

Чтобы выбрать группу узлов сети и применить для неё фильтр тегов, нажмите на *Выбор*, чтобы получить полный список существующих групп узлов сети или начните вводить имя группы узлов сети, чтобы увидеть выпадающее меню с соответствующими группами. Если вы хотите применить фильтры тегов к вложенным группам узлов сети, отметьте опцию *Включая подгруппы*.

Фильтр тегов позволяет отделить доступ к группе узлов сети от возможности увидеть проблемы.

Например, если администратору баз данных нужно видеть проблемы только по “MySQL” базе данных, сначала необходимо создать группу пользователей для администраторов баз данных, затем указать имя тега равное “Сервис” и значение равное “MySQL”.

Если имя тега “Сервис” задано и поле значения оставлено пустым, тогда соответствующая группа пользователей будет видеть все проблемы по выбранной группе узлов сети с именем тега равным “Сервис”. Если поля имени тега и значения тега оставлены пустыми, но группа узлов сети выбрана, соответствующая группа пользователей увидит все проблемы по выбранной группе узлов сети. Убедитесь, что имя тега и значение этого тега указаны корректно, в противном случае соответствующая группа пользователей не увидит проблемы вовсе.

Давайте рассмотрим пример, когда пользователь является членом нескольких выбранных групп пользователей. В этом случае фильтрация для тегов будет использовать OR условие.

Группа пользователей А

Фильтр тегов

Группа узлов сети	Имя тега	Значение тега
Шаблоны/Базы данных	Сервис	MySQL
Шаблоны/Базы данных не выбрано	пусто	пусто
	пусто	пусто

Группа пользователей В

Группа узлов сети	Имя тега	Значение тега
Шаблоны/Базы данных	Сервис	Oracle
Шаблоны/Базы данных	Сервис	Oracle
Шаблоны/Базы данных	Сервис	Oracle

Видимый результат у пользователя (члена) обеих групп

Видны проблемы Сервис: MySQL или Oracle
Видны все проблемы
Видны проблемы Сервис:Oracle



Добавление фильтра (например, все теги в определенной группе узлов сети “Шаблоны/Базы данных”) приведет к невозможности просмотра проблем с других групп узлов сети.

Доступ к узлу сети из нескольких групп пользователей

Пользователь может принадлежать любому числу групп пользователей. Эти группы могут иметь различные права доступа к узлам сети.

Поэтому, важно знать, какие узлы сети, в результате, будут доступны для не привилегированного пользователя. Например, давайте рассмотрим, как доступ к узлу сети X (из Группы узла сети 1) будет меняться в различных ситуациях для пользователей которые состоят в группах А и В.

- Если Группа пользователей А имеет доступ только на Чтение к Группе узлов сети 1, но Группа пользователей В имеет доступ Чтение-Запись к Группе узлов сети 1. Пользователь получит доступ на Чтение-Запись для узла сети “X”.



Право доступа “Чтение-Запись” имеет более высокий приоритет перед правом доступа “Чтение” начиная с Zabbix 2.2.

- В таком же сценарии как описан выше, если узел сети “X” находится еще и в Группе узлов сети 2 которая имеет право доступа Запрещено для Групп пользователей А и В, доступ к узлу сети “X” **будет не возможен**, несмотря на наличия прав доступа Чтение-Запись к Группе узлов сети 1.
- Если Группа пользователей А не имеет заданных прав доступа и Группа узлов сети В имеет права доступа Чтение-Запись к Группе узлов сети 1, пользователь получит право доступа Чтение-Запись к узлу сети “X”.
- Если Группа пользователей А имеет доступ Запрещено к Группе узлов сети 1 и Группа пользователей В имеет доступ на Чтение-Запись к Группе узлов сети 1, пользователь **не получит** доступа к узлу сети “X”.

Другая информация

- Пользователь с уровнем Администратор с правом доступа Чтение-запись узлу сети не сможет присоединять/отсоединять шаблоны, если у него нет прав доступа к группе Шаблоны. При наличии прав доступа на Чтение к группе Шаблоны 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 ИТ.
- Любой не Zabbix Супер-Администратор пользователь (включая 'guest') может просматривать карты сети пока карта пустая или имеет только изображения. Права соблюдаются при наличии добавленных узлов сети, групп узлов сети или триггер на карте сети. Такое же поведение применимо к комплексным экранам и слайд-шоу. Пользователи, вне зависимости от прав доступа, видят любые объекты, которые напрямую или не напрямую присоединены к узлам сети.
- Zabbix сервер не будет отправлять оповещения пользователям, которые указаны получателями в операции действия, если доступ к соответствующему узлу сети явно "запрещен".

8. Мониторинг услуг

Обзор

Функция мониторинга услуг предназначена для тех, кто хочет получить мониторинг инфраструктуры более высокого уровня (бизнес). В большинстве случаев мы не заинтересованы в низкоуровневых деталях, таких как недостаток места на диске, высокая загрузка процессора и т.д. В чем мы заинтересованы, так это доступность сервиса, предоставляемым нашим IT отделом. Мы также можем быть заинтересованы в выявлении слабых мест в IT инфраструктуре, SLA различных IT услуг, структуре существующей IT инфраструктуры, и в другой информации на более высоком уровне.

Мониторинг услуг в Zabbix дает ответы на все выше упомянутые вопросы.

Услугами является иерархическое представление наблюдаемых данных.

Очень простая структура услуг может выглядеть следующим образом:

Услуга

```
| -Рабочие станции
|   | -Рабочая станция1
|   | -Рабочая станция2
| -Сервера
```

У каждого узла структуры имеется атрибут состояния. Состояние вычисляется и распространяется на верхние уровни в соответствии с выбранным алгоритмом. Самым низким уровнем услуг является триггерами. Состояние отдельных узлов дерева зависит от состояния их триггеров.



Обратите внимание, что триггеры с важностями *Без классификации* и *Информационный* не влияют на расчет SLA.

Настройка

Для настройки услуг, перейдите в *Настройка* → *Услуги*.

На этой странице вы можете построить иерархию инфраструктуры, которую вы мониторите. Самым высоким уровнем будет родительская услуга 'root'. Вы можете построить свою иерархию ниже, добавляя родительские услуги более низкого уровня, а затем и отдельные узлы дерева к ним.

Services	
Service	Action
root	Add child
▼ Servers	Add child
Server 1	Add child Delete
Server 2	Add child Delete
Server 3	Add child Delete
Server 4	Add child Delete
Server 5	Add child Delete

Нажмите на *Добавить потомка*, чтобы добавить к ней услуги. Для изменения существующей услуги, нажмите на имя этой услуги. Будет отображен диалог, где вы можете изменить атрибуты услуги.

Настройка услуги

Вкладка **Услуга** состоит из общих атрибутов услуги:

The screenshot shows the 'Service' configuration dialog. The 'Service' tab is active. The fields are as follows:

- Name: Server 1
- Parent service: SLA by service
- Status calculation algorithm: Problem, if at least one child has a problem
- Calculate SLA, acceptable SLA (in %): 99.9000
- Trigger: New host: Zabbix agent on New host is unreachable
- Sort order (0->999): 0

At the bottom are three buttons: 'Update' (blue), 'Delete' (light blue), and 'Cancel'.

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	Имя услуги.
Родительская услуга	Принадлежность к родительской услуги.
Алгоритм вычисления состояния	Метод вычисления состояния услуги: Без вычисления - не вычислять состояние услуги Проблема, если хотя бы одна дочерняя услуга в состоянии проблема - состояние проблемы, только, если по крайней мере одна дочерняя услуга в состоянии проблема Проблема, если все дочерние услуги в состоянии проблем - состояние проблемы, только, если все дочерние услуги в состоянии проблема
Вычислять SLA	Включение вычисления и отображение SLA.
Допустимый уровень SLA (в %)	Процент SLA, который является приемлемым для этой услуги. Используется для отчетности.
Триггер	Связь с триггером: Пусто - связи нет имя триггера - связана с триггером, таким образом зависит от состояния триггера Услуги нижнего уровня должны быть соединены с триггерами. (В противном случае их состояние не будет представлено точно.) Когда триггеры связаны, их состояния до связи не считаются.
Порядок сортировки	Порядок сортировки при отображении, меньшие значения идут первыми.
Вкладка Зависимости	содержит услуги зависящие от выбранной услуги. Нажмите на <i>Добавить</i> для добавления услуг из тех что уже настроены.

Service	Dependencies	Time																				
	<table border="1"> <thead> <tr> <th>Depends on</th> <th>SERVICES</th> <th>SOFT</th> <th>TRIGGER</th> </tr> </thead> <tbody> <tr> <td>Server 2</td> <td><input type="checkbox"/></td> <td></td> <td></td> </tr> <tr> <td>Server 3</td> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> </tr> <tr> <td>Server 4</td> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> </tr> <tr> <td>Add</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Depends on	SERVICES	SOFT	TRIGGER	Server 2	<input type="checkbox"/>			Server 3	<input checked="" type="checkbox"/>			Server 4	<input checked="" type="checkbox"/>			Add				<input type="button" value="Update"/> <input type="button" value="Delete"/> <input type="button" value="Cancel"/>
Depends on	SERVICES	SOFT	TRIGGER																			
Server 2	<input type="checkbox"/>																					
Server 3	<input checked="" type="checkbox"/>																					
Server 4	<input checked="" type="checkbox"/>																					
Add																						

Жесткая и нежесткая зависимость

Доступность услуги может зависеть от нескольких других услуг, не только от одной. Первая опция используется для добавления всех этих услуг, которые непосредственно являются потомками услуг.

Однако, если какая-то услуга уже добавлены где-то в другом месте в дереве услуг, она не может быть попросту перемещена к новой дочерней услуги. Как создать зависимость от неё? Ответом является “нежесткая” ссылка. Добавьте услугу и отметьте флажок *Нежесткая*. Таким образом услуга может оставаться на своем привычном месте в дереве, но и может быть зависима от нескольких других услуг. Услуги, которые являются “нежестко-связанными” отображаются в дереве серым цветом. Кроме этого, если услуга имеет только “нежесткие” зависимости, она может быть удалена сразу, без предварительного удаления дочерних услуг.

Вкладка **Время** содержит спецификацию интервалов времени услуги.

Service	Dependencies	Time																				
		<table border="1"> <thead> <tr> <th>Service times</th> <th>Type</th> <th>Interval</th> <th>Note</th> </tr> </thead> <tbody> <tr> <td>New service time</td> <td>Period type</td> <td>Uptime</td> <td></td> </tr> <tr> <td></td> <td>* From</td> <td>Sunday</td> <td>Time hh : mm</td> </tr> <tr> <td></td> <td>* Till</td> <td>Sunday</td> <td>Time hh : mm</td> </tr> <tr> <td></td> <td>Add</td> <td></td> <td></td> </tr> </tbody> </table>	Service times	Type	Interval	Note	New service time	Period type	Uptime			* From	Sunday	Time hh : mm		* Till	Sunday	Time hh : mm		Add		
Service times	Type	Interval	Note																			
New service time	Period type	Uptime																				
	* From	Sunday	Time hh : mm																			
	* Till	Sunday	Time hh : mm																			
	Add																					
		<input type="button" value="Add"/> <input type="button" value="Cancel"/>																				

Параметр

Интервалы времени услуги По умолчанию, все услуги, как ожидается, действуют 24x7x365. Если требуются исключения, добавьте новые интервалы времени услуги.

Новый интервал времени услуги Интервалы времени услуги:

Доступна - время доступности услуги

Недоступна - состояние услуги не влияет на SLA в течении указанного периода.

Однократная недоступность - одиночная недоступность. Состояние услуги не влияет на SLA в течении этого периода.

Добавьте соответствующие часы.

Обратите внимание: Интервалы времени влияют только на услугу, на которой они были добавлены. Таким образом, родительская услуга не принимает во внимание

Описание

Параметр

Описание

интервалы времени указанные на дочерней услуге (если также, конечно, одинаковые интервалы времени не добавлены на родительской услуге).
Интервалы времени принимаются во внимание при вычислении состояния услуги и SLA в веб-интерфейсе. Тем не менее, информация о доступности услуги добавляется в базу данных непрерывно, вне зависимости от интервалов времени услуги.

Просмотр

Для мониторинга услуг, перейдите в [Мониторинг -> Услуги](#).

9. Веб-мониторинг

Обзор

Благодаря Zabbix вы можете проверять несколько аспектов доступности веб-сайтов.



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 предопределенном порядке. Если узел сети наблюдается через прокси, тогда шаги выполняются на этом прокси.

Начиная с Zabbix 2.2 веб-сценарии привязываются к узлам сети/шаблонам тем же образом как элементы данных, триггеры и т.д. Это означает, что веб-сценарии можно создавать не 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 путям, эмулируя нажатия мышкой на странице.

Веб-мониторинг в Zabbix поддерживает и HTTP, и HTTPS. При выполнении веб-сценария, Zabbix сервер будет следовать перенаправлениям (смотрите опцию *Следовать перенаправлениям* ниже). Максимальное количество перенаправлений жестко задано в исходном коде и равняется 10 (используется cURL опция [CURLOPT_MAXREDIRS](#)). Все cookies запоминаются на протяжении выполнения одного сценария.

Смотрите также [известные проблемы](#) по веб-мониторингу при использовании HTTPS протокола.

Настройка сценария

Для настройки веб-сценария:

- Перейдите: *Настройка* → *Узлы сети* (или *Шаблоны*)
- Нажмите на *Веб* в строке с узлом сети/шаблоном
- Нажмите на *Создать сценарий* в верхнем правом углу (или на имени сценария для редактирования существующего сценария)
- Введите в диалоге параметры сценария

Вкладка **Сценарий** позволяет вам настроить общие параметры веб-сценария.

Все обязательные поля ввода отмечены красной звёздочкой.

Параметры сценария:

Параметр	Описание
Узел сети	Имя узла сети/шаблона к которому принадлежит сценарий.
Имя	Уникальное имя сценария. Начиная с Zabbix 2.2 поддерживаются пользовательские макросы и {HOST.*} макросы .
Группа элементов данных	Выберите группу элементов данных к которой будет принадлежать сценарий. Элементы данных веб-сценария будут сгруппированы под выбранной группой элементов данных в Мониторинг → Последние данные .
Новая группа элементов данных	Введите название новой группы элементов данных для сценария.
Интервал обновления	Как часто сценарий будет выполняться. Начиная с Zabbix 3.4.0, поддерживаются суффиксы времени , например, 30s, 1m, 2h, 1d. Пользовательские макросы поддерживаются, начиная с 3.4.0. Обратите внимание что, если используется пользовательский макрос и его значение изменилось (к примеру, 5m → 30s), следующая проверка будет выполнена в соответствии с предыдущим значением (в далеком будущем с примерами значений).
Попыток	Количество попыток выполнения шагов веб-сценария. В случае сетевых проблем (превышено время ожидания, отсутствие подключения и прочего) Zabbix может повторить выполнение шагов несколько раз. Указанное количество будет одинаково действовать для каждого шага сценария. Можно указать до 10 попыток, значение по умолчанию равно 1.

Параметр	Описание
<i>Агент</i>	<p><i>Примечание:</i> Zabbix не повторит шаг из-за ошибочного кода ответа или несовпадении необходимой строки.</p> <p>Данный параметр поддерживается начиная с Zabbix 2.2.</p> <p>Выбор агента клиента.</p> <p>Zabbix будет представляться выбранным браузером. Полезно для мониторинга Веб-сайтов, которые генерируют различное содержимое для разных браузеров.</p> <p><i>Начиная с Zabbix 2.2</i>, в этом поле можно использовать пользовательские макросы.</p> <p>Вы можете указать необходимый HTTP прокси, следуя следующему формату: <code>http://[имя пользователя[:пароль]@]прокси.тускомпани.ком[:порт]</code></p> <p>По умолчанию будет использоваться порт 1080.</p> <p>Если указан, прокси заменит переменные окружения связанные с прокси такие как <code>http_proxy</code>, <code>HTTPS_PROXY</code>. Если не указан, переменные окружения не будут заменены.</p>
<i>HTTP прокси</i>	<p>Введённое значение передается “как есть”, проверка правильности не производится. Вы также можете указать адрес SOCKS прокси. Если вы укажите ошибочный протокол, подключение провалится и элемент данных станет неподдерживаемым. Если протокол не указан, прокси будет считаться HTTP прокси.</p> <p><i>Примечание:</i> Для HTTP прокси поддерживается только простая аутентификация.</p> <p>В этом поле можно использовать пользовательские макросы.</p> <p>Данный параметр поддерживается начиная с Zabbix 2.2.</p> <p>Переменные, которые можно использовать в шагах сценария (URL, переменные post).</p> <p>Переменные имеют следующий формат:</p> <pre>{макрос1}=значение1 {макрос2}=значение2 {макрос3}=regex:<регулярное выражение></pre> <p>Например:</p> <pre>{username}=Alexei {password}=kj3h5kJ34bd {hostid}=regex: hostid is ([0-9]+)</pre> <p>На эти макросы затем можно ссылаться в шагах сценария, используя {username}, {password} и {hostid}. Zabbix автоматически заменит их на актуальные значения.</p> <p>Обратите внимание, что переменным с <code>regex</code>: требуется по крайней мере один шаг, чтобы получить значение с регулярного выражения, поэтому извлечённое значение можно применять только в последующих шагах.</p> <p>Если часть значения начинается с <code>regex</code>:, тогда последующая часть обрабатывается как регулярное выражение, которое будет искать указанную часть веб-страницы, и если найдет, запомнит найденное значение в переменную. Должна присутствовать как минимум одна подгруппа так, чтобы найденные значения можно было извлечь.</p> <p>Переменные, которые ищут совпадение части веб-страницы по регулярному выражению, поддерживаются начиная с Zabbix 2.2.</p> <p>Пользовательские макросы и {HOST.*} макросы поддерживаются начиная с Zabbix 2.2.</p> <p>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.</p> <p>Пользовательские HTTP заголовки, которые будут отправлены при выполнении запроса.</p> <p>Заголовки следует передавать списком используя тот же синтаксис как они могут появиться в HTTP протоколе, дополнительно можно использовать некоторые дополнительные возможности поддерживаемые CURLOPT_HTTPHEADER опциями cURL.</p> <p>Например: Accept-Charset: utf-8</p>
<i>Переменные</i>	
<i>Заголовки</i>	

Параметр

Описание

Accept-Language: en-US

Content-Type: application/xml; charset=utf-8

Пользовательские макросы и {HOST.*} [макросы](#) поддерживаются начиная с Zabbix 2.2.

Возможность указать пользовательские заголовки поддерживается начиная с Zabbix 2.4.

Активирован Сценарий активирован, если параметр отмечен, в противном случае - деактивирован.

Обратите внимание, что при редактировании существующего сценария, в диалоге будут доступны две дополнительные кнопки:

Clone

Создание другого сценария на основе свойств существующего.

Clear history and trends

Удаление у сценария данных истории и динамики изменений. Эта опция заставит сервер выполнить сценарий сразу после удаления данных.

Если поле *HTTP прокси* оставить пустым, можно воспользоваться другим способом указать HTTP прокси, для этого необходимо задать переменные окружения.



Для HTTP проверок - укажите переменную окружения **http_proxy** для пользователя Zabbix сервера. Например, `http_proxy=http://proxy_ip:proxy_port`.

Для HTTPS проверок - укажите переменную окружения **HTTPS_PROXY**. Например, `HTTPS_PROXY=http://proxy_ip:proxy_port`. Более подробную информацию можно получить, выполнив в shell команду `# man curl`.

Вкладка **Шаги** позволит вам настроить шаги веб-сценария. Чтобы добавить шаг веб-сценария, нажмите на *Добавить* в блоке *Шаги*.

Name	Timeout	URL	Required	Step
1: Home	15s	http://www.google.com	Yes	1
2: About	15s	http://www.google.com/intl/en/about	Yes	2

Настройка шагов

Step of web scenario

*Name	Home
*URL	http://www.google.com
Query fields	Name Value name value <input type="button" value="Remove"/> <input type="button" value="Add"/>
Post type	<input checked="" type="radio"/> Form data <input type="radio"/> Raw data
Post fields	Name Value name value <input type="button" value="Remove"/> <input type="button" value="Add"/>
Variables	Name Value name value <input type="button" value="Remove"/> <input type="button" value="Add"/>
Headers	Name Value name value <input type="button" value="Remove"/> <input type="button" value="Add"/>
Follow redirects	<input checked="" type="checkbox"/>
Retrieve only headers	<input type="checkbox"/>
*Timeout	15s
Required string	
Required status codes	200

Параметры шага:

Параметр	Описание
Имя	Уникальное имя шага. Начиная с Zabbix 2.2, имя может содержать поддерживаемые макросы .
URL	

9.1 Элементы данных веб-мониторинга

Обзор

Некоторые элементы данных автоматически добавляются для наблюдения при создании веб-сценариев.

Элементы данных сценария

Как только сценарий будет создан, Zabbix автоматически добавит сопутствующие элементы данных для наблюдения и привяжет их к выбранной группе элементов данных.

Элемент данных	Описание
<i>Скорость загрузки сценария <Сценарий></i>	Этот элемент данных собирает информацию о скорости загрузки (байт в секунду) всего сценария, т.е. средний показатель для всех шагов. Ключ элемента данных: web.test.in[Сценарий,,bps] Тип: Числовой (с плавающей точкой)
<i>Неуспешный шаг сценария <Сценарий></i>	Этот элемент данных отображает номер неуспешного шага в сценарии. Если все шаги в сценарии выполнены успешно, то возвращается 0. Ключ элемента данных: web.test.fail[Сценарий] Тип: Числовой (целое)
<i>Последнее сообщение об ошибке для сценария <Сценарий></i>	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. Ключ элемента данных: web.test.error[Сценарий] Тип: Символ
	Вместо “Сценарий” будет использовано реальное имя сценария.



Элементы данных веб-мониторинга добавляются с 30 дневным периодом хранения истории и с 90 дневным периодом хранения динамики изменений.



Если имя сценария начинается с двойной кавычки, содержит запятую или квадратные скобки, оно будет корректно заключено в кавычки в ключах элементов данных. В других случаях дополнительные кавычки не добавляются.

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:

```
{host:web.test.fail[Сценарий].last()}<>0
```

Не забудьте заменить 'Сценарий' на реальное название вашего сценария.

Пример 2

Для создания триггера “Проверка веб сценария завершилась ошибкой”, вы можете указать триггер с именем:

Веб сценарий "Scenario" завершился ошибкой: {ITEM.VALUE}

и выражением триггера:

```
{host:web.test.error[Сценарий, ].strlen()}>0 and {host:web.test.fail[Сценарий].last()}>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:

```
{host:web.test.in[Сценарий,,bps].last()}<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 наблюдения и привяжет их к выбранной группе элементов данных.

Элемент данных

Скорость загрузки для шага <Шаг> сценария <Сценарий>

Время ответа для шага <Шаг> сценария <Сценарий>

Код ответа для шага <Шаг> сценария <Сценарий>

Будут использоваться настоящие имена сценария и шага вместо “Сценарий” и “Шаг” соответственно.

Описание

Этот элемент данных собирает информацию о скорости загрузки (байт в секунду) для указанного шага.

Ключ элемента данных: web.test.in[Сценарий,Шаг,bps]

Тип: Числовой (с плавающей точкой)

Этот элемент данных собирает информацию о времени ответа в секундах для указанного шага. Временем ответа считается от начала отправки запроса до окончания получения всей передаваемой информации.

Ключ элемента данных: web.test.time[Сценарий,Шаг]

Тип: Числовой (с плавающей точкой)

Этот элемент данных собирает информацию о возвращенных кодах ответа указанного шага.

Ключ элемента данных: web.test.rspcode[Сценарий,Шаг]

Тип: Числовой (целое положительное)



Элементы данных Веб-мониторинга добавляются с 30-дневным сроком хранения истории и 90-дневным сроком хранения динамики изменений.



Если имя сценария начинается с двойных кавычек, содержит запятые или 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 настройке условий оповещения. Например для создания триггера “Вход в Zabbix веб-интерфейс слишком медленный”, вы можете указать следующее выражение триггера:

```
{zabbix:web.test.time[ZABBIX GUI,Login].last()}>3
```

9.2 Сценарий из реальной жизни

Обзор

Этот раздел описывает пошаговый пример из реальной жизни, того как можно использовать веб-мониторинг.

Давайте использовать веб-мониторинг Zabbix для наблюдения за Веб-интерфейсом Zabbix. Мы хотим знать доступен ли он, корректно ли возвращаемое содержимое и насколько быстро оно отдается.

Итак, сначала мы должны войти в систему, используя наше имя пользователя и пароль.

Сценарий

Шаг 1

Добавим новый веб-сценарий.

Мы добавим сценарий для наблюдения за Веб-интерфейсом Zabbix. Сценарий будет выполняться за несколько шагов.

Перейдите в *Настройка* → *Узлы сети*, выберите узел сети и нажмите на *Веб* в строке с этим узлом сети. Затем нажмите кнопку *Создать веб-сценарий*.

Scenario Steps Authentication

* Name Zabbix frontend

Application

New application Zabbix frontend

* Update interval 1m

* Attempts 1

Agent Firefox 33.0 (Linux)

HTTP proxy http://[user[:password]@]proxy.example.com[:port]

Variables

Name	Value
{user}	⇒ Admin
{password}	⇒ zabbix

Add

Headers

Name	Value
name	⇒ value

Add

Enabled

Add **Cancel**

The screenshot shows the configuration of a new scenario named "Zabbix frontend". It includes variables for user and password, and a header for the name. The scenario is enabled.

Все обязательные поля ввода отмечены красной звёздочкой.

В диалоге нового сценария мы назовем сценарий *Доступность веб-интерфейса Zabbix* и создадим для него новую группу элементов данных *Веб проверки*.

Обратите внимание, мы также создаем две переменные: {user} и {password}.

Шаг 2

Зададим шаги для нашего сценария.

Нажмите кнопку *Добавить* на вкладке *Шаги* для добавления индивидуальных шагов.

Шаг веб-сценария 1

Мы начнем с проверки того, что первая страница отвечает правильно, возвращает код ответа HTTP 200 и содержит текст "Zabbix SIA".

Step of web scenario

* Name	First page					
* URL	http://localhost/zabbix/index.php	Parse				
Query fields	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table>		Name	Value	name	value
Name	Value					
name	value					
	Add					
Post type	Form data Raw data					
Post fields	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table>		Name	Value	name	value
Name	Value					
name	value					
	Add					
Variables	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table>		Name	Value	name	value
Name	Value					
name	value					
	Add					
Headers	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table>		Name	Value	name	value
Name	Value					
name	value					
	Add					
Follow redirects	<input checked="" type="checkbox"/>					
Retrieve only headers	<input type="checkbox"/>					
*Timeout	15s					
Required string	Zabbix SIA					
Required status codes	200					
<input type="button" value="Add"/> <input type="button" value="Cancel"/>						

После завершения настройки этого шага, нажмите *Добавить*.

Шаг веб-сценария 2

Мы продолжаем, осуществляя вход в веб-интерфейс Zabbix, мы используем для этого макросы (переменные) - {user} и {password}, которые задали на уровне сценария.

Step of web scenario

* Name	Log in	<input type="button" value="Parse"/>								
* URL	http://192.168.6.87/zabbix/index.php	<input type="button" value="Parse"/>								
Query fields <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> <input type="button" value="Add"/>			Name	Value	name	value				
Name	Value									
name	value									
Post type	<input checked="" type="radio"/> Form data	<input type="radio"/> Raw data								
Post fields <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>{user}</td> </tr> <tr> <td>password</td> <td>{password}</td> </tr> <tr> <td>enter</td> <td>Sign in</td> </tr> </tbody> </table> <input type="button" value="Add"/>			Name	Value	name	{user}	password	{password}	enter	Sign in
Name	Value									
name	{user}									
password	{password}									
enter	Sign in									
Variables <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>{sid}</td> <td>regex:name="csrf-token" content="(#[0-9a-zA-Z]{16})</td> </tr> </tbody> </table> <input type="button" value="Add"/>			Name	Value	{sid}	regex:name="csrf-token" content="(#[0-9a-zA-Z]{16})				
Name	Value									
{sid}	regex:name="csrf-token" content="(#[0-9a-zA-Z]{16})									
Headers <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> <input type="button" value="Add"/>			Name	Value	name	value				
Name	Value									
name	value									
Follow redirects	<input checked="" type="checkbox"/>									
Retrieve only headers	<input type="checkbox"/>									
* Timeout	15s									
Required string	<input type="text"/>									
Required status codes	<input type="text"/> 200									
<input type="button" value="Update"/> <input type="button" value="Cancel"/>										



Обратите внимание, что в веб-интерфейсе Zabbix используется JavaScript перенаправление при входе, поэтому первый шагом мы должны сделать вход в систему, и только в дальнейших шагах мы можем осуществлять проверки, требующие авторизованной сессии. К тому же, шаг входа в систему должен выполняться используя полный URL к файлу **index.php**.

Обратите также внимание на то, как мы получаем содержимое переменной **{sid}** (ID сессии), используя синтаксис переменных с регулярным выражением: `regex:name="sid" value="([0-9a-zA-Z]{16})"`. Эта переменная потребуется в шаге 4.

Шаг веб-сценария 3

Войдя в систему, мы теперь должны проверить этот факт. Чтобы это сделать, мы проверим наличие строки, видимой только после успешного входа - например, появление раздела **Администрирование**.

Step of web scenario

* Name	Login check					
* URL	http://localhost/zabbix/index.php <input type="button" value="Parse"/>					
Query fields	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> <input type="button" value="Add"/>		Name	Value	name	value
Name	Value					
name	value					
Post type	<input checked="" type="button" value="Form data"/> <input type="button" value="Raw data"/>					
Post fields	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> <input type="button" value="Add"/>		Name	Value	name	value
Name	Value					
name	value					
Variables	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> <input type="button" value="Add"/>		Name	Value	name	value
Name	Value					
name	value					
Headers	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>name</td> <td>value</td> </tr> </tbody> </table> <input type="button" value="Add"/>		Name	Value	name	value
Name	Value					
name	value					
Follow redirects	<input checked="" type="checkbox"/>					
Retrieve only headers	<input type="checkbox"/>					
* Timeout	15s					
Required string	Administration					
Required status codes	200					

Шаг веб-сценария 4

Теперь, когда мы убедились, что Веб-интерфейс доступен и мы можем войти и получить содержимое страницы авторизованной сессии, мы должны выйти из системы - в противном случае база данных Zabbix будет “засоряться” все больше и больше от записей открытых сессий.

Step of web scenario

*Name Log out

*URL http://localhost/zabbix/index.php Parse

Query fields

Name	Value
reconnect	1
sid	{sid}

Add

Post type Form data Raw data

Post fields

Name	Value
name	value

Add

Variables

Name	Value
name	value

Add

Headers

Name	Value
name	value

Add

Follow redirects

Retrieve only headers

*Timeout 15s

Required string

Required status codes 200

Шаг веб-сценария 5

Мы также можем проверить что вышли из системы, поискав строку с **Username**.

Step of web scenario

*Name

*URL

Query fields

Name	Value
<input type="text" value="name"/>	<input type="text" value="value"/> <input type="button" value="Remove"/>

Add

Post type Form data Raw data

Post fields

Name	Value
<input type="text" value="name"/>	<input type="text" value="value"/> <input type="button" value="Remove"/>

Add

Variables

Name	Value
<input type="text" value="name"/>	<input type="text" value="value"/> <input type="button" value="Remove"/>

Add

Headers

Name	Value
<input type="text" value="name"/>	<input type="text" value="value"/> <input type="button" value="Remove"/>

Add

Follow redirects

Retrieve only headers

*Timeout

Required string

Required status codes

Весь список настройки шагов

Список добавленных шагов в Веб-сценарий должны выглядеть примерно так:

Scenario	Steps	Authentication					
	* Steps	Name	Timeout	URL	Required	Status codes	Action
	1: First page		15s	http://localhost/zabbix/index.php	Zabbix SIA	200	Remove
	2: Log in		15s	http://localhost/zabbix/index.php		200	Remove
	3: Login check		15s	http://localhost/zabbix/index.php	Administration	200	Remove
	4: Log out		15s	http://localhost/zabbix/index.php		200	Remove
	5: Logout check		15s	http://localhost/zabbix/index.php	Username	200	Remove

Шаг 3

Сохраним законченный сценарий Веб-мониторинга.

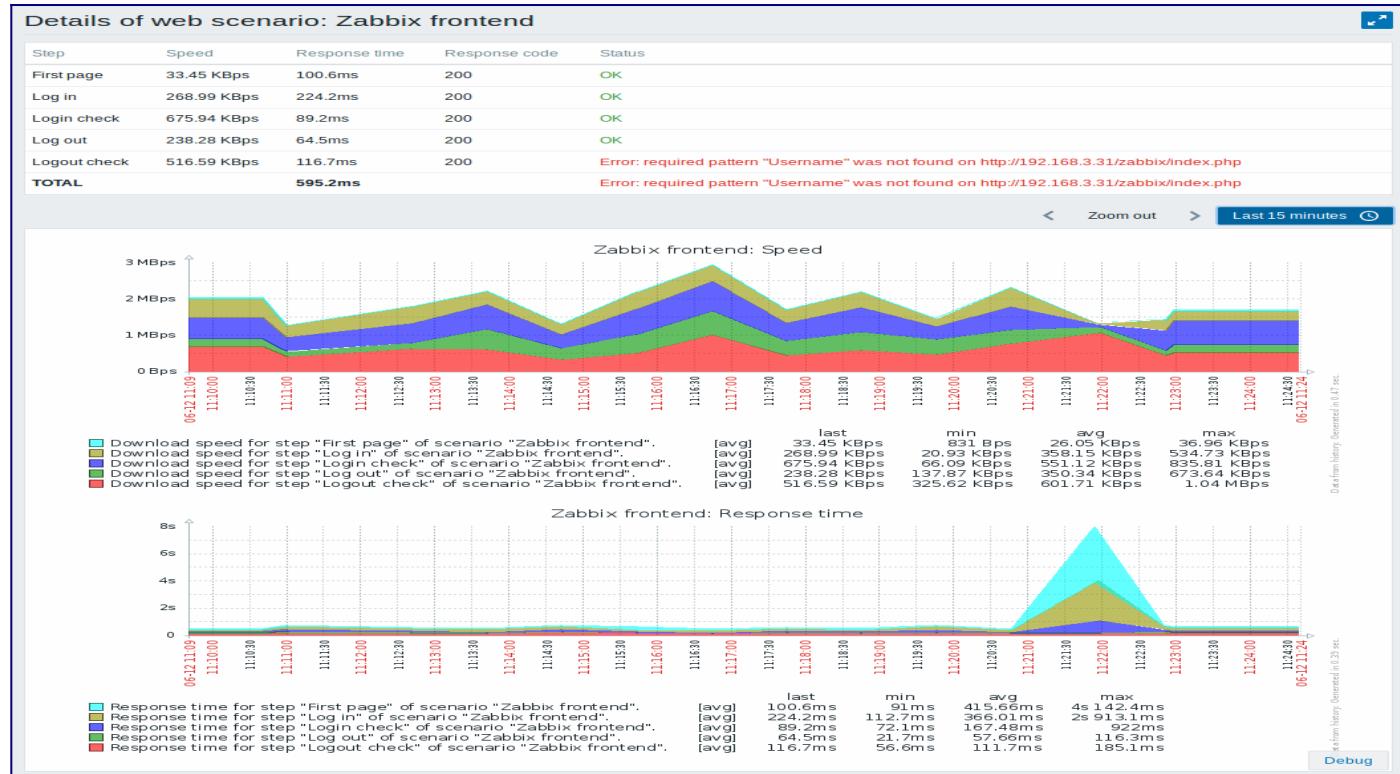
Сценарий появится в Мониторинг → Веб:

Web monitoring

Group Zabbix servers ▾ Host all ▾

Host	Name ▲	Number of steps	Last check	Status
Zabbix server	Zabbix frontend	5	2017-03-24 08:32:50	OK

Нажмите на имени сценария, чтобы просмотреть более детальную статистику:



10. Мониторинг виртуальных машин

Обзор

Поддержка мониторинга сред VMware доступна в Zabbix начиная с версии 2.2.0.

Zabbix может использовать правила низкоуровневого обнаружения для автоматического обнаружения гипервизоров и виртуальных машин VMware, и затем создавать узлы сети для наблюдения за ними, основываясь на предустановленных прототипах узлов сети.

Поставляемый с Zabbix начальный набор данных предлагает несколько готовых к использованию шаблонов для мониторинга VMware vCenter и ESX гипервизоров.

Минимально требуемой версией VMware vCenter или vSphere является 4.1.

Подробности

Мониторинг виртуальных машин выполняется в два шага. Первый, данные о виртуальных машинах собираются Zabbix процессами *vmware коллекторами*. Эти процессы получают необходимую информацию с веб служб VMware через SOAP протокол, подготовливают эту информацию и записывают в разделяемую память Zabbix сервера. Затем, эти данные извлекаются с помощью поллеров, используя [VMware ключи](#) простых проверок Zabbix.

Начиная с версии Zabbix 2.4.4 собираемые данные разделены на 2 типа: данные конфигурации VMware и данные счётчиков производительности VMware. Оба эти типа собираются независимо *vmware коллекторами*. В связи с этим рекомендуется активировать больше коллекторов, чем служб VMware под мониторингом. В противном случае получение статистики основанной на счетчиках производительности VMware может задерживаться получением данных конфигурации VMware (что занимает длительное время на больших инсталляциях).

В настоящее время только статистика по хранилищам данных, сетевых интерфейсов и дисковых устройств основывается на информации со счётчиков производительности VMware.

Настройка

Для того чтобы мониторинг виртуальных машин заработал, необходимо, чтобы Zabbix был [скомпилирован](#) с опциями сборки `--with-libxml2` и `--with-libcurl`.

Можно использовать следующие опции из файла конфигурации для оптимизации мониторинга виртуальных машин:

- **StartVMwareCollectors** - количество префорк процессов vmware коллектора.
Это значение зависит от количества служб VMware, которые вы наблюдаете. Для большинства случаев значением должно быть:
 $колвослужб < StartVMwareCollectors < (колвослужб * 2)$,
где колвослужб является количество служб VMware. Например, если у вас под наблюдением 1 служба VMware, задайте StartVMwareCollectors равным 2, если у вас 3 службы VMware, задайте параметр равным 5. Обратите внимание, что в большинстве случаев это значение не должно быть меньше 2 и не должно превышать более чем в 2 раза, чем количество служб VMware, которые вы мониторите. Также имейте в виду, что эта величина зависит также от размера вашей среды VMware, параметров конфигурации `VMwareFrequency` и `VMwarePerfFrequency` (смотрите ниже).
- **VMwareCacheSize**
- **VMwareFrequency**

- **VMwarePerfFrequency**
- **VMwareTimeout**

Для получения большей информации смотрите страницы файлов конфигурации Zabbix [сервера](#) и [прокси](#).



Для поддержки метрик объемов хранилищ данных Zabbix требуется значение vpxd.stats.maxQueryMetrics параметра VMware по крайней мере 64. Смотрите также [статью \[en\]](#) в базе знаний VMware.

Обнаружение

Zabbix может использовать правило обнаружения для автоматического обнаружения гипервизоров и виртуальных машин VMware.

Discovery rule
[Filters](#)

* Name

Type

Simple check

* Key

User name

Password

* Update interval (in sec)

Custom intervals

TYPE	INTERVAL	PERIOD
Flexible	Scheduling	50
	Add	1-7,00:

* Keep lost resources period (in days)

Description

Discovery of hypervisors.

Enabled

Все обязательные поля ввода отмечены красной звёздочкой.

Ключом правила обнаружения на снимке экрана выше является `vmware.hv.discovery[{$URL}]`.

Прототипы узлов сети

Прототипы узлов сети можно создать с помощью правила низкоуровневого обнаружения. Когда виртуальные машины обнаруживаются, их прототипы становятся реальными узлами сети. Прототипы, до момента обнаружения, не могут иметь своих собственных элементов данных и триггеров, кроме тех, которые унаследованы из присоединенных шаблонов. Обнаруженные узлы сети

будут принадлежать существующему узлу сети и унаследуют IP адрес существующего узла сети в своей конфигурации.

Discovery rules

All templates / Template Virt VMware Applications 3 Items 3 Triggers Graphs Screens Discovery

NAME ▲	ITEMS	TRIGGERS	GRAPHS	HOSTS
<input type="checkbox"/> Discover VMware clusters	Item prototypes 1	Trigger prototypes	Graph prototypes	Host prototypes
<input type="checkbox"/> Discover VMware hypervisors	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes
<input type="checkbox"/> Discover VMware VMs	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes

В настройке прототипа узлов сети используются LLD макросы для полей имени узла сети, видимого имени и прототипа группы узлов сети. Соединение с существующими группами узлов сети, присоединение шаблонов и шифрование и другие опции, которые можно указать.

Host Groups Templates Host inventory Encryption

* Host name

Visible name

Create enabled

Add Cancel

Если *Создать активированным* выбрано, узел сети добавится в активированном состоянии. Если не выбрано, узел сети добавится, но в деактивированном состоянии.

В списке узлов сети обнаруженные узлы сети имеют префикс с именем правила обнаружения, от которого они были созданы. Обнаруженные узлы сети могут быть удалены вручную. Обнаруженные узлы автоматически удаляются, основываясь на значении *Период хранения потерянных ресурсов (в днях)* правила обнаружения. Большинство опций конфигурации доступно только на чтение, исключая активацию/деактивацию узла сети и инвентарные данные. Обнаруженные узлы сети не могут иметь свои собственные прототипы узлов сети.

Готовые к использованию шаблоны

Поставляемый с Zabbix начальный набор данных предлагает несколько готовых к использованию шаблонов для мониторинга VMware vCenter и мониторинга напрямую ESX гипервизоров.

Эти шаблоны содержат предустановленные правила низкоуровневого обнаружения, а также некоторое количество встроенных проверок для мониторинга виртуальных инсталляций.

Заметьте, что шаблон “Template Virt VMware” необходимо использовать для мониторинга VMware vCenter и ESX гипервизора. Шаблоны “Template Virt VMware Hypervisor” и “Template Virt VMware Guest” используются для обнаружения и обычно не присоединяются вручную к узлам сети.

Templates

<input type="checkbox"/> TEMPLATES ▾	APPLICATIONS	ITEMS	TRIGGERS	GRAPHS	SCREENS	D
<input type="checkbox"/> Template Virt VMware Hypervisor	Applications 6	Items 19	Triggers	Graphs	Screens	Dis
<input type="checkbox"/> Template Virt VMware Guest	Applications 8	Items 17	Triggers	Graphs	Screens	Dis
<input type="checkbox"/> Template Virt VMware	Applications 3	Items 3	Triggers	Graphs	Screens	Dis

Если ваш сервер обновлен с версии до 2.2 и не имеет описанных шаблонов, вы можете их импортировать вручную, загрузив со страницы [официальных шаблонов](#) с сервера сообщества. Однако, эти шаблоны имеют зависимости от преобразований значений *VMware VirtualMachinePowerState* и *VMware status*, поэтому необходимо сначала создать эти преобразования значений (используя [SQL скрипт](#) или вручную) до импорта шаблонов.

Настройка узла сети

Для того чтобы использовать простые проверки VMware, на узел сети должны быть назначены следующие макросы:

- **{\$URL}** - служба VMware (vCenter или ESX hypervisor) SDK URL (<https://servername/sdk>).
- **{\$USERNAME}** - Имя пользователя службы VMware
- **{\$PASSWORD}** - Пароль к пользователю {\$USERNAME} службы VMware

Пример

Следующий пример демонстрирует как быстро в Zabbix можно настроить VMware мониторинг:

- скомпилируйте zabbix сервер с требуемыми опциями (--with-libxml2 и --with-libcurl)
- задайте параметр StartVMwareCollectors в файле конфигурации Zabbix сервера равным 1 или более
- создайте новый узел сети
- укажите требуемые для аутентификации в VMware макросы на уровне узла сети:

Hosts

Host	Templates	IPMI	Macros	Host inventory	Encryption								
			<input type="button" value="Host macros"/> <input type="button" value="Inherited and host macros"/>										
			<table><thead><tr><th>MACRO</th><th>VALUE</th></tr></thead><tbody><tr><td>{\$PASSWORD}</td><td>⇒ <password></td></tr><tr><td>{\$URL}</td><td>⇒ <url></td></tr><tr><td>{\$USERNAME}</td><td>⇒ <username></td></tr></tbody></table>	MACRO	VALUE	{\$PASSWORD}	⇒ <password>	{\$URL}	⇒ <url>	{\$USERNAME}	⇒ <username>		
MACRO	VALUE												
{\$PASSWORD}	⇒ <password>												
{\$URL}	⇒ <url>												
{\$USERNAME}	⇒ <username>												
			<input type="button" value="Add"/>										
			<input type="button" value="Add"/> <input type="button" value="Cancel"/>										

- Присоедините шаблон к шаблону службы VMware:

Hosts

NAME	ACTION
Template Virt VMware	Unlink

Link new templates

[Select](#)

[Add](#) [Cancel](#)

- Нажмите на кнопку *Добавить*, чтобы сохранить узел сети

Расширенное журналирование

Можно записывать в файл журнала сбор данных коллектором VMware, 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). При условии, что уровень отладки 4 уже задан, расширенное журналирование можно запустить так, как демонстрируется в следующих примерах:

Увеличение уровня журналирования по всем коллекторам vmware:
shell> zabbix_server -R log_level_increase="vmware collector"

Увеличение уровня журналирования по второму коллектору vmware:
shell> zabbix_server -R log_level_increase="vmware collector,2"

Если расширенное журналирование данных коллекторов VMware не требуется, его можно установить, используя -R log_level_decrease.

Решение проблем

- В случае недоступности метрик, пожалуйста, убедитесь, что они не стали недоступными или отключенными по умолчанию в последних версиях VMware vSphere или, не наложены ли какие-либо ограничения на запросы метрик производительности в базу данных. Смотрите [ZBX-12094](#) для получения более подробных сведений.

10.1 Поля ключей обнаружения виртуальных машин

В следующей таблице перечислены, связанные с ключами обнаружения, поля, которые возвращаются виртуальными машинами.

Описание	Ключ элемента данных		Источник данных
	Поле		
Выполнение обнаружения кластеров.	vmware.cluster.discovery	{#CLUSTER.ID} {#CLUSTER.NAME}	Идентификатор кластера. Имя кластера.
	vmware.hv.discovery		

Ключ элемента данных		
Выполнение обнаружения гипервизоров.	{#HV.UUID}	Уникальный идентификатор гипервизора.
	{#HV.ID}	Идентификатор гипервизора (Имя управляемого объекта HostSystem).
	{#HV.NAME}	Имя гипервизора.
	{#CLUSTER.NAME}	Имя кластера, может быть пустым.
	}	
vmware.hv.datastore.discovery		
Выполнение обнаружения хранилищ данных гипервизора. Обратите внимание, что несколько гипервизоров могут использовать одинаковые хранилища данных.	{#DATASTORE}	Имя хранилища данных.
vmware.vm.discovery		
Выполнение обнаружения виртуальных машин.	{#VM.UUID}	Уникальный идентификатор виртуальной машины.
	{#VM.ID}	Идентификатор виртуальной машины (Имя объекта управляемое VirtualMachine).
	{#VM.NAME}	Имя виртуальной машины.
	{#HV.NAME}	Имя гипервизора.
	{#CLUSTER.NAME}	Имя кластера, может быть пустым.
	}	
vmware.vm.net.if.discovery		
Выполнение обнаружения сетевых интерфейсов на виртуальной машине.	{#IFNAME}	Имя сетевого интерфейса.
vmware.vm.vfs.dev.discovery		
Выполнение обнаружения дисковых устройств на виртуальной машине.	{#DISKNAME}	Имя дискового устройства.
vmware.vm.vfs.fs.discovery		
Выполнение обнаружения файловых систем на виртуальной машине.	{#FSNAME}	Имя файловой системы.

11. Обслуживание

Обзор

В Zabbix вы можете указать периоды обслуживания групп узлов сети, групп узлов сети и отдельных триггеров.

Имеется два типа обслуживания - со сбором данных и без сбора данных.

В течении обслуживания “со сбором данных” триггеры обрабатываются как обычно и создаются события, если потребуется. Однако, эскалации о проблемах будут поставлены на паузу для узлов сети/триггеров в обслуживании, если в настройках действия выбрана опция *Приостановить операции для подавленных проблем*. В этом случае шаги эскалаций, которые могут включать отправку оповещений или удаленные команды будут игнорироваться до тех пор пока не завершится период обслуживания.

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, тогда шаги второй и третий будут выполнены спустя полчаса, или спустя 60 минут и 90 минут (при условии, что проблема всё еще имеется). Аналогично, если проблема произойдёт в процессе 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.



Если по крайней мере один узел сети (используемый в выражении триггера) находится не в режиме обслуживания, Zabbix отправит оповещение о проблеме.

Zabbix сервер должен работать во время обслуживания. Процессы таймера ответственны за переключение состояния узлов сети в/без обслуживания в 0 секунду каждой минуты. Обратите внимание, когда узел сети входит в режим обслуживания, процессы таймера Zabbix сервера теперь будут перечитывать все открытые проблемы для проверки требуется ли их подавить. Такое поведение может повлиять на производительность, если имеется большое количество проблем. Zabbix сервер будет также читать все открытые проблемы при запуске процесса, даже если на это время отсутствуют настроенные обслуживания.

Прокси всегда собирает данные, независимо от типа обслуживания (включая обслуживание “без данных”). Эти данные, в последствии, игнорируются сервером, если указан тип 'без сбора данных'.

После завершения обслуживания “без данных”, триггеры использующие функцию nodata() не “поджигаются” до момента следующей проверки в течении периода времени обновления этих элементов данных.

Если добавлен элемент данных журнала 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 значение со штампом времени отправляется на узел сети, который находится в обслуживании с типом обслуживания “без данных” (например при использовании [Zabbix sender](#)), тогда это значение будет отброшено, однако, имеется возможность отправить значение со штампом времени в течении истекшего периода обслуживания и оно будет принято.



Для обеспечения предсказуемого поведения периодов обслуживания (ежедневного, еженедельного, ежемесячного), требуется использовать общий часовой пояс для всех компонентов Zabbix.

Если период обслуживания, узлы сети, группы или теги изменяются пользователем, то такие изменения вступают в силу только после синхронизации кэша конфигурации.

Настройка

Для настройки периода обслуживания:

- Перейдите в: *Настройка* → *Обслуживание*
- Нажмите на *Создать период обслуживания* (или на имени уже существующего периода обслуживания)

Вкладка **Обслуживание** содержит общие атрибуты периода обслуживания:

The screenshot shows the 'Maintenance' configuration dialog. At the top, there are three tabs: 'Maintenance' (selected), 'Periods', and 'Hosts and groups'. The 'Maintenance' tab contains the following fields:

- * Name: Weekly maintenance
- Maintenance type: A radio button group where 'With data collection' is selected, and 'No data collection' is unselected.
- * Active since: Date input field containing 2018-01-01 00:00, with a calendar icon to its right.
- * Active till: Date input field containing 2019-01-01 00:00, with a calendar icon to its right.
- Description: Text area containing the text "We break and fix things at this time."

At the bottom of the dialog are two buttons: 'Add' (blue) and 'Cancel' (white).

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	Имя периода обслуживания. Можно указать два типа обслуживания:
Тип обслуживания	Со сбором данных - данные будут собираться сервером в течении обслуживания, триггеры будут обрабатываться Без сбора данных - данные не будут собираться сервером в течении обслуживания
Активно с	Дата и время, когда выполняемые периоды обслуживания становятся активными. <i>Обратите внимание:</i> Установка только этого времени не активирует период обслуживания; чтобы это сделать перейдите на вкладку <i>Периоды</i> .
Активно до	Дата и время, когда выполняемые периоды обслуживания перестают быть активными.
Описание	Описание периода обслуживания.

Вкладка **Периоды** позволяет вам указать точные дни и часы, когда обслуживание будет действовать. При нажатии на *Новый*, отобразится диалог гибкого *Период обслуживания*, где вы можете задать время - для ежедневного, еженедельного, ежемесячного или единовременного обслуживания.

Maintenance Periods [Hosts and groups](#)

* Periods	Period type	Schedule	Period	Action
	Weekly	At 15:00 Monday of every week	1h	Edit

Maintenance period

Period type	Weekly
* Every week(s)	1
* Day of week	<input checked="" type="checkbox"/> Monday <input type="checkbox"/> Tuesday <input type="checkbox"/> Wednesday <input type="checkbox"/> Thursday <input type="checkbox"/> Friday <input type="checkbox"/> Saturday <input type="checkbox"/> Sunday
At (hour:minute)	15 : 0
* Maintenance period length	0 Days 1 Hours 0 Minutes
Update Cancel	
Add	Cancel

Ежедневный и еженедельный периоды имеют параметр *Каждый день/Каждую неделю*, который по умолчанию равен 1. Изменив его значение на 2, обслуживание будет выполняться каждые два дня или каждые две недели и так далее. Исходным днем или неделей считается время, когда наступит время *Активно с*.

Например, имеется *Активно с* заданное как 2013-09-06 в 12:00 и длительностью в час с повтором каждые два дня в 23:00 - в результате первый период обслуживания начнется с 2013-09-06 в 23:00, тогда как второй период обслуживания начнется с 2013-09-08 в 23:00. Или, с тем же временем *Активно с* и длительностью в час с повтором каждые два дня в 01:00, первый период обслуживания начнется с 2013-09-08 в 01:00, и второй период обслуживания начнется с 2013-09-10 в 01:00.

Вкладка **Узлы сети и группы** позволяет вам выбрать группы узлов сети, узлы сети и теги проблем для обслуживания.

Maintenance Periods Hosts and groups

* At least one host group or host must be selected.

Host groups type here to search

Hosts type here to search

Tags And/Or Or

tag Contains Equals value

Add

Cancel

Параметр

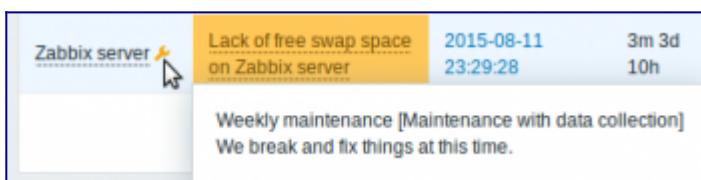
Описание

Группы узлов сети	Выберите группы узлов сети для которых обслуживание будет активировано. Обслуживание будет активировано по всем узлам сети из указанных групп(ы) узлов сети. Это поле с функцией автодополнения, таким образом после начала ввода имени группы, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню.
	Если указать родительскую группу, косвенным образом будут выбраны все вложенные группы узлов сети. Таким образом обслуживание будет также активировано на узлах сети из вложенных групп.
Узлы сети	Выберите узлы сети для которых обслуживание будет активировано. Это поле с функцией автодополнения, таким образом после начала ввода имени узла сети, будет автоматически предложен список совпадающих узлов сети в выпадающем меню.
	Если заданы теги, тогда обслуживание по выбранным узлам сети будет ограничено проблемами с соответствующими тегами.
Теги	В случае нескольких тегов поведение вычисляется следующим образом: И/Или - должны соответствовать все теги; однако, теги с одинаковыми именами вычисляются с условием Или Или - достаточно соответствия одного тега
	Имеется два способа сопоставления значения тега: Содержит - регистрозависимое совпадение подстроки Равно - регистрозависимое соответствие строке

Просмотр

Отображение узлов сети в обслуживании

Иконка с оранжевым ключом вблизи имени узла сети в разделах интерфейса Мониторинг → Панель, Мониторинг → Проблемы и Инвентаризация → Узлы сети → Инвентарные данные узла сети означает, что этот узел сети находится в обслуживании.



Детали обслуживания отображаются при наведении курсора мыши на иконку.

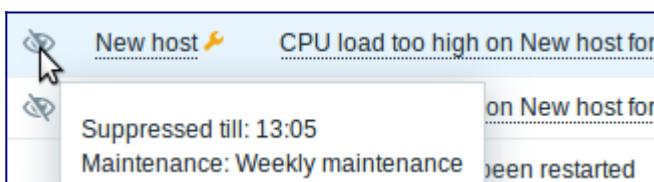
Кроме этого, у узлов сети появляется оранжевый фон в *Мониторинг* → *Карты сети и в Настройка* → *Узлы сети* их состояние отображается как в “обслуживании”.

Отображение подавленных проблем

Обычно проблемы с узлами сети в обслуживании подавляются, то есть не отображаются в веб-интерфейсе. Однако, также имеется возможность настроить таким образом, что подавленные проблемы будут отображены, выбрав опцию *Отображение подавленных проблем* в следующих местах:

- Мониторинг → Панель (при настройке виджетов *Узлы сети с проблемами*, *Проблемы*, *Проблемы по важности*, *Обзор триггеров*)
- Мониторинг → Проблемы (в фильтре)
- Мониторинг → Обзор (в фильтре; с 'Триггеры' в *Tun*)
- Мониторинг → Карты сети (при настройке карты)
- Глобальные [оповещения](#) (при настройке профиля пользователя)

При отображении подавленных проблем отображается следующая иконка:  При наведении курсора мыши отобразятся более детальные сведения:



12. Регулярные выражения

Обзор

Zabbix поддерживает [Perl совместимые регулярные выражения](#) (PCRE).

Имеется два варианта использования регулярных выражений в Zabbix:

- ввод регулярного выражения вручную
- использование глобальных регулярных выражений созданных в Zabbix

Регулярные выражения

Вы можете вручную ввести регулярное выражение в поддерживаемых местах. Обратите внимание, что в этом случае выражение может не начинаться с @, так как этот символ в Zabbix используется как ссылка на глобальные регулярные выражения.

Глобальные регулярные выражения

В Zabbix веб-интерфейсе имеется продвинутый редактор для создания и тестирования сложных регулярных выражений.

После того как регулярное выражение создано, его можно использовать в нескольких местах веб-интерфейса, используя ссылку на его имя с префиксом @, например, `@moe_пользовательское_выражение`.

Для создания глобального регулярного выражения:

- Перейдите в: Администрирование → Общие
- Выберите Регулярные выражения в выпадающем списке
- Нажмите на Новое регулярное выражение

Вкладка **Выражения** позволяет указать имя регулярного выражения и добавить подвыражения.

EXPRESSION TYPE	EXPRESSION	DELIMITER	CASE SENSITIVE
Result is FALSE	^lo\$		<input checked="" type="checkbox"/>
Result is FALSE	^Software Loopback Interface		<input checked="" type="checkbox"/>

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр

Имя

Укажите имя регулярного выражения. Разрешены любые Unicode символы.

Выражения

Нажмите на *Добавить* в Блоке выражений для того, чтобы добавить новое подвыражение.

Выберите тип выражения:

Строка символов включена - совпадение по подстроке

Любая строка символов включена - совпадение по любой из подстрок из списка с разделителями. Разделители в списке включают в себя запятую (,), точку (.) или прямую косую черту (/).

Строка символов не включена - совпадение по любой строке за исключением подстроки

Результат ПРАВДА - совпадение с регулярным выражением

Результат ЛОЖЬ - отсутствие совпадения с регулярным выражением

Укажите подстроку/регулярное выражение.

Запятая (,), точка (.) или прямая косая черта (/) для разделения текстовых строк в регулярном выражении. Этот параметр доступен только когда выбран режим типа выражения “*Любая строка символов включена*”.

Регистрозависимое

Отметьте эту опцию чтобы указать является ли регулярное выражение чувствительным к регистру букв.

Начиная с Zabbix 2.4.0, косая черта (/) в выражении воспринимается буквально, нежели как разделитель. Таким образом возможно сохранить выражения содержащие косую черту, тогда как ранее такая запись приводила к ошибке.



В Zabbix имя пользовательского регулярного выражения может содержать запятые, пробелы и т.п. В этих случаях, там где имеется вероятность неверной интерпретации при использовании ссылки (например, запятая в параметре ключа элемента данных) нужно всю ссылку заключить в кавычки, вот таким образом: "@Мое выражение для цели, цели2".

В других местах (например, в свойствах LLD правил) имена регулярных выражений нельзя заключать в кавычки.

Пример

Использование следующего регулярного выражения в LLD для обнаружения баз данных не будет принимать во внимание базу данных с заданным именем:

`^TESTDATABASE$`

Test string	TESTDATABASE
Test expressions	
Result	Expression type
	Result is FALSE
	Combined result
	Expression
	^TESTDATABASE
	Result
	FALSE
	Combined result
	FALSE

Выбран *Тип выражения*: "Результат ЛОЖЬ". Не соответствует имени, содержащему строку "TESTDATABASE".

Пример с использованием inline модификатора регулярного выражения

Использование следующего регулярного выражения, которое включает inline модификатор (?i) для поиска соответствия символов "error":

(?i)error

Test string	Sometexthere1345Error1357
Test expressions	
Result	Expression type
	Result is TRUE
	Combined result
	Expression
	(?i)error
	Result
	TRUE
	Combined result
	TRUE

Выбран *Тип выражения*: "Результат ПРАВДА". Символы "error" совпадают.

Другой пример с использованием inline модификатора регулярного выражения

Использование следующего регулярного выражения, которое включает несколько inline модификаторов для поиска соответствия символов после указанной строки:

(?<=match (?i)everything(?-i) after this line\n)(?sx).*# мы добавили s модификатор, чтобы разрешить . соответствие символам перехода на новую строку

Test string

```
Some text here for your consideration
1235kfd345
match eveRything after this line
Continuation
```

[Test expressions](#)

Result

Expression type Expression

Result is TRUE (?<=match (?i)everything(?-i) after this line\n)(?sx).*# we add s modifier to allow . match newline character

Combined result

Выбран Тип выражения: "Результат ПРАВДА". Символы после указанной строки совпадают.



Модификатор **g** нельзя указывать в строке. Список доступных модификаторов можно найти на [странице помощи по pcresyntax](#) [en]. Для получения более подробной информации о синтаксисе PCRE, пожалуйста, обратитесь к [PCRE HTML документации](#) [en].

Более сложный пример

Пользовательское регулярное выражение может состоять из нескольких подвыражений, и их можно протестировать на вкладке **Тест**, используя тестовую строку.

[Expressions](#) [Test](#)

Test string `lo`

[Test expressions](#)

Result	Expression type	Expression	Result
Result is FALSE	Expression	<code>^Software Loopback Interface</code>	TRUE
Result is FALSE	Expression	<code>^(In)?[Ll]oop[Bb]ack[0-9._]*\$</code>	TRUE
Result is FALSE	Expression	<code>^NULL[0-9.]*</code>	TRUE
Result is FALSE	Expression	<code>^[Ll]o[0-9.]*</code>	FALSE
Result is FALSE	Expression	<code>^[\$s]ystem\$</code>	TRUE
Result is FALSE	Expression	<code>^Nu[0-9.]*</code>	TRUE
Combined result			FALSE

[Update](#) [Clone](#) [Delete](#) [Cancel](#)

Результаты покажут состояние каждого подвыражения и суммарное состояние для всего пользовательского выражения.

Общее состояние пользовательского выражения определяется как *Комбинированный результат*. Если задано несколько подвыражений, для вычисления *Комбинированного результата* Zabbix использует логический оператор И. Такое поведение означает, если по крайней мере один Результат является ЛОЖЬЮ, тогда *Комбинированный результат* будет также иметь состояние Ложь.

Объяснение глобальных регулярных выражений

Глобальное регулярное выражение	Выражение	Описание
<i>File systems for discovery</i>	<code>^(btrfs ext2 ext3 ext4 jfs reiser xfs ffs ufs jfs jfs2 vxfs hfs refs ntfs fat32 zfs)\$</code>	Соответствие “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 “ntfs” или “fat32” или “zfs”
<i>Network interfaces for discovery</i>	<code>^Software Loopback Interface</code> <code>^lo\$</code>	Соответствие строкам, которые начинаются с “Software Loopback Interface” Соответствие “lo”
	<code>^(In)?[Ll]oop[Bb]ack[0-9_.]*\$</code>	Соответствие строк, которые начинаются с “In”, затем имеют “L” или “l”, затем “oop”, затем “B” или “b”, затем “ack”, за которыми могут следовать, необязательно, любое количество цифр, точек или подчёркиваний
	<code>^NULL[0-9.]*\$</code>	Соответствие строк, которые начинаются с “NULL” и затем, опционально, следует любое количество цифр или точек
	<code>^[Ll]o[0-9.]*\$</code>	Соответствие строк, которые начинаются с “Lo” или “lo” и затем, опционально следует любое количество цифр или точек
	<code>^[Ss]ystem\$</code>	Соответствие “System” или “system”
	<code>^Nu[0-9.]*\$</code>	Соответствие строк, которые начинаются с “Nu” и затем, опционально следует любое количество цифр или точек
<i>Storage devices for SNMP discovery</i>	<code>^(Physical memory Virtual memory Memory buffers Cached memory Swap space)\$</code>	Соответствие “Physical memory” или “Virtual memory” или “Memory buffers” или “Cached memory” или “Swap space”
<i>Windows service names for discovery</i>	<code>^(MMCSS gupdate SysmonLog clr_optimization_v2.0.50 727_32 clr_optimization_v4.0.30 319_32)\$</code>	Соответствие “MMCSS” или “gupdate” или “SysmonLog” или строк наподобии “clr_optimization_v2.0.50727_32” и “clr_optimization_v4.0.30319_32”, где вместо точек вы можете поместить любой символ за исключением символа новой строки.
<i>Windows service startup states for discovery</i>	<code>^(automatic automatic delayed)\$</code>	Соответствие “automatic” или “automatic delayed”.

Поддержка регулярных выражений по назначению

Назначение	Регулярное выражение	Глобальное регулярное выражение	Комментарии
Элементы данных агента			
eventlog[]	Да	Да	Параметры регулярное выражение, важность, источник, eventid
log[]			Параметр регулярное выражение
log.count[]			

Назначение	Регулярное выражение	Глобальное регулярное выражение	Комментарии
logrt[]			
logrt.count[]		Да/Нет	Параметр регулярное выражение поддерживает оба варианта, параметр <code>регулярное_выражение_файла</code> поддерживает только не глобальные выражения
proc.cpu.util[]			Параметр <code> cmdline</code>
proc.mem[]			
proc.num[]			
sensor[]			Параметры устройство и датчик в Linux 2.4
system.hw.macaddr[]			Параметр интерфейс
system.sw.packages[]			Параметр пакет
vfs.dir.count[]			Параметры <code>regex_вкл</code> и <code>regex_искл</code>
vfs.dir.size[]			Параметры <code>regex_вкл</code> и <code>regex_искл</code>
vfs.file.regexp[]			
vfs.file.regmatch[]			
web.page.regexp[]			Параметр регулярное выражение
SNMP трапы			
snmptrap[]	Да	Да	Параметр <code> regexp</code>
Предобработка значений элементов данных			
	Да	Нет	Параметр шаблон
Функции триггеров			
count()			Параметр шаблон, если параметр оператор равен <code>regexp</code> или <code>iregexp</code>
logeventid()			
logsource()	Да	Да	
iregexp()			Параметр шаблон
regexp()			
Низкоуровневое обнаружение			
	Да	Да	Поле <i>Фильтр</i>
Веб-мониторинг			
	Да	Нет	<i>Переменные с префиксом regex:</i> Поле <i>Требуемая строка</i>
Функции макросов			
regsub()			
iregsub()	Да	Нет	Параметр шаблон
Соответствие иконок			
	Да	Да	Поле <i>Выражение</i>

13. Подтверждение проблем

Обзор

В Zabbix события о проблемах могут подтверждаться пользователями.

Если пользователь получает оповещение о проблемном событии, он может зайти в веб-интерфейс Zabbix, перейти из списка проблем на экран обновления проблем и подтвердить проблему. При

подтверждении, он может ввести свой комментарий о проблеме, сказав, что он работает над проблемой или то, что пользователь думает над этой проблемой.

Таким образом, если другой пользователь замечает ту же самую проблему, он сразу же увидит, была ли проблема подтверждена и, конечно, сами комментарии.

Этот способ рабочего процесса решения проблем, при наличии более одного пользователя системы, может быть более скоординированным решением.

Состояние подтверждения также используется при определении [операций действия](#). Вы можете задать, например, что оповещение отправляется менеджеру более высокого уровня только, если событие не было подтверждено некоторое время.

Для подтверждения событий, пользователь должен иметь по крайней мере права на чтение соответствующего триггера.

Имеется **два** варианта доступа к экрану обновления проблем, который позволяет подтвердить проблему.

Первый способ, вы можете нажать на колонку *Подтверждено*, которая отображает состояние подтверждения проблемы, в:

- Мониторинг → ПАНЕЛЬ (виджеты *Последние 20 проблем* и *Проблемы по важности*)
- Мониторинг → Проблемы
- Мониторинг → Проблемы → Детали события
- Мониторинг → Комплексные экраны (в элементах *События в группах узлов сети*, *События у узла сети*, *Проблемы по важности*)

Колонка *Подтверждено* содержит либо ссылку 'Да', либо ссылку 'Нет', указывая на подтвержденную или неподтвержденную проблему соответственно. При нажатии на эти ссылки вы попадете на экран обновления проблем.

Второй способ, вы можете нажать на ячейку нерешенной проблемы в:

- Мониторинг → ПАНЕЛЬ (виджеты *Обзор данных* и *Обзор триггеров*)
- Мониторинг → Обзор
- Мониторинг → Обзор (элементы *Обзор данных* и *Обзор триггеров*)

Выпадающее меню содержит опцию *Подтверждение*, которая позволяет вам перейти на экран обновления проблем.

Обновление проблем

Экран обновления проблем позволяет:

- прокомментировать проблему
- просмотреть текущие комментарии и выполненные действия
- изменить важность проблемы
- подтвердить проблему
- закрыть проблему вручную

Update problem

Message
History

Time	User	User action	Message
2018-06-20 07:46:43	Admin (Zabbix Administrator)		Started working on it.

Scope Only selected problem
 Selected and all other problems of related triggers 1 event
Change severity Not classified Information Warning Average High Disaster
Acknowledge
Close problem
* At least one update operation or message must exist.

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Сообщение	Введите текст комментария к проблеме. Перечисляются предыдущие действия и комментарии к проблеме, вместе со временем и детальной информацией о пользователе.
История	Чтобы узнать значения иконок, которые используются к действиям пользователей, смотрите страницу деталей событий . Определение масштаба по таким действиям как изменение важности, подтверждению или закрытия проблем вручную: Только выбранная проблема - повлияет только на выбранное событие Выбранная и все остальные проблемы связанных триггеров - в случае подтверждения/закрытия проблемы, повлияет на выбранное событие и все остальные проблемы, которые не были ранее подтверждены/закрыты. Если область применения состоит из уже подтвержденных или закрытых проблем, такие проблемы не будут повторно подтверждены/закрыты. С другой стороны, количество сообщений и операций изменения важности не ограничены.
Область	Отметьте эту опцию и нажмите на кнопку важности, чтобы обновить важность проблемы.
Изменить важность	Отметьте эту опцию и нажмите на кнопку важности, чтобы обновить важность проблемы.
Подтвердить	Отметьте эту опцию, чтобы подтвердить проблему. Эта опция отключена по уже подтвержденным проблемам.
Закрыть проблему	Отметьте эту опцию, чтобы закрыть проблему вручную. Закрыть проблему таким способом можно, если в настройке триггера выбрана опция <i>Разрешить закрывать вручную</i> .

Просмотр

Основываясь на информации о подтверждении, имеется возможность настроить каким образом будет отображаться количество проблем на ПАНЕЛИ или картах сети. Чтобы это сделать, вам необходимо сделать выбор в опции *Отображение проблем*, доступной как при [настройке карт сетей](#), так и [Проблемы по важности виджете панели](#). Имеется возможность отображения количества всех проблем, количества неподтвержденных проблем, отдельно от общего количества, или только количество неподтвержденных проблем.

Основываясь на информации обновления проблем (подтверждения и т.д.), имеется возможность настроить операции обновления - отправить сообщение или выполнить удаленные команды.

14. Экспорт/импорт настроек

Обзор

Функциональность экспорта/импорта настроек в Zabbix делает возможным выполнять обмен объектами конфигурации между одной Zabbix системой и другой.

Типичные случаи применения этой функционала:

- обмен шаблонами и картами сети - Zabbix пользователи могут опубликовывать их параметры конфигурации
- обмен веб-сценариями на share.zabbix.com - экспортируйте шаблон с веб-сценариями и загрузите его на share.zabbix.com. Другие люди могут загрузить этот шаблон себе и импортировать XML в Zabbix.
- интеграция с прикладными инструментами - универсальный XML формат делает возможным интеграцию и импорт/экспорт данных в прикладные инструменты и приложения.

Что можно импортировать/экспортировать

Объекты, которые можно импортировать/экспортировать:

- [группы узлов сети](#) (только через Zabbix API)
- [шаблоны](#)
- [узлы сети](#)
- [карты сети](#)
- изображения
- [комплексные экраны](#)
- преобразования значений

Формат экспорта

Данные можно экспортировать при помощи веб-интерфейса Zabbix или [Zabbix API](#). Поддерживаемые форматы экспорта:

- XML - в веб-интерфейсе Zabbix
- XML или JSON - в Zabbix API

Дополнительная информация о экспорте

- Все поддерживаемые элементы экспортируются в один файл.
- Объекты на узлах сети и в шаблонах (элементы данных, триггеры, графики, правила обнаружения), которые унаследованы из шаблонов не экспортируются. Любые изменения выполненные над этими объектами на уровне узла сети (такие как, изменен интервал обновления, изменено регулярное выражение или добавлены прототипы к низкоуровневому

правилу обнаружения) при экспорте будут потеряны; при импорте все объекты из присоединенных шаблонов пересоздаются, такими как они есть в присоединенном шаблоне.

- Созданные низкоуровневым обнаружением элементы и любые объекты, которые связаны с ними, не экспортируются. Например, если триггер создан для элемента данных, генерированного правилом обнаружения, то такой триггер не будет экспортирован.

Дополнительная информация об импорте

- Импорт останавливается при первой же ошибке.
- При обновлении существующих изображений в процессе импорта изображения, поле "imagetype" игнорируется, т.е. невозможно изменить тип изображения с помощью функции импорта.
- При импорте узлов сети/шаблонов, при использовании опции "Удалить отсутствующие", макросы узла сети/шаблона, которых нет в импортируемом XML файле будут также удалены.
- Пустые теги элементов данных, триггеров, графиков, групп элементов данных узлов сети/шаблонов, правил обнаружения, прототипов элементов данных, прототипов триггеров, прототипов графиков бессмысленны, т.к. они имеют такой же смысл, как если бы были не указаны вовсе. Другие теги, например группы элементов данных, имеют смысл, например, пустой тег означает отсутствие группы элементов данных у элемента данных, пропущенный тег означает запрет обновления групп элементов данных.
- Импорт поддерживает как XML, так и JSON, импортируемый файл должен иметь корректное расширение файла: .xml для XML и .json для JSON
- Смотрите [информацию о совместимости](#) по поводу поддерживаемых версий XML.

Общий формат XML

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>4.0</version>
    <date>2016-10-04T06:20:11Z</date>
</zabbix_export>
```

<?xml version="1.0" encoding="UTF-8"?>

Заголовок XML документов по умолчанию.

```
<zabbix_export>
```

Корневой элемент при экспорте Zabbix XML.

```
<version>4.0</version>
```

Версия экспорта.

```
<date>2016-10-04T06:20:11Z</date>
```

Дата, когда был выполнен экспорт, в формате ISO 8601 long.

Остальные теги зависят от экспортируемых объектов.

14.1 Группы узлов сети

В веб-интерфейсе группы узлов сети можно [экспортировать](#) только вместе с экспортом узла сети или шаблона. Когда экспортируется узел сети или шаблон, все группы которым он принадлежит экспортируются вместе с ним автоматически.

API позволяет экспортить группы узлов сети независимо от узлов сети или шаблонов.

```
<groups>
  <group>
    <name>Zabbix servers</name>
  </group>
</groups>
```

groups/group

Параметр	Тип	Описание	Детали
name	строка	Имя группы.	

14.2 Шаблоны

Обзор

Шаблоны [экспортируются](#) вместе со множеством связанных объектов и со связями с объектами.

Экспорт шаблона содержит:

- присоединенные группы узлов сети
- данные шаблона
- связи с другими шаблонами
- связи с группами узлов сети
- напрямую присоединенные группы элементов данных
- напрямую присоединенные элементы данных
- напрямую присоединенные триггеры
- напрямую присоединенные графики
- напрямую присоединенные комплексные экраны
- напрямую присоединенные правила обнаружения со всеми прототипами
- напрямую присоединенные веб-сценарии
- преобразования значений

Экспорт

Для экспорта шаблонов сделайте следующее:

- Перейдите в: *Настройки* → *Шаблоны*
- Отметьте шаблоны, которые нужно экспортировать
- Нажмите на *Экспорт* ниже списка

<input type="checkbox"/>	Name	Applications
<input checked="" type="checkbox"/>	Template App MySQL	Applications 1

1 selected **Export** Delete Delete and clear

Выбранные шаблоны экспортируются в локальный XML файл с именем по умолчанию *zabbix_export_templates.xml*.

Импорт

Для импорта шаблонов сделайте следующее:

- Перейдите в: *Настройка* → *Шаблоны*
- Нажмите на *Импорт* справа
- Выберите импортируемый файл
- Отметьте необходимые опции в правилах импорта
- Нажмите на *Импорт*

The screenshot shows a dialog box titled 'Import file' with a 'Browse...' button and a message 'No file selected.' Below this is a table with two columns: 'Rules' and actions ('Update existing', 'Create new', 'Delete missing'). The 'Groups' row has checked boxes for all three actions. Other rows like 'Hosts', 'Template screens', and 'Template linkage' have mixed checked/unchecked status across the columns.

Rules	Update existing	Create new	Delete missing
Groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Hosts	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template screens	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template linkage	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Applications	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Graphs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Web scenarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Screens	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Maps	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Images	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Value mappings	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Import **Cancel**

Все обязательные поля ввода отмечены красной звёздочкой.

После импорта в веб-интерфейсе отобразится сообщение об успехе или об ошибке.

Правила импорта:

Правило	Описание
Обновить существующее	Существующие элементы будут обновлены данными взятыми с файла импорта. В противном случае они не будут обновляться.
Создать новое	Импорт добавит новые элементы, используя данные из файла импорта. В противном случае они не будут добавляться.
Удалить пропущенное	Импорт удалит существующие элементы, которые отсутствуют в файле импорта. В противном случае они не будут удаляться.

Формат экспорта

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>4.0</version>
    <date>2018-04-03T06:25:22Z</date>
    <groups>
        <group>
```

```

<name>Templates/Databases</name>
</group>
</groups>
<templates>
    <template>
        <template>Template DB MySQL</template>
        <name>Template DB MySQL</name>
        <description/>
        <groups>
            <group>
                <name>Templates/Databases</name>
            </group>
        </groups>
        <applications>
            <application>
                <name>MySQL</name>
            </application>
        </applications>
        <items>
            <item>
                <name>MySQL status</name>
                <type>0</type>
                <snmp_community/>
                <snmp_oid/>
                <key>mysql.ping</key>
                <delay>1m</delay>
                <history>1w</history>
                <trends>365d</trends>
                <status>0</status>
                <value_type>3</value_type>
                <allowed_hosts/>
                <units/>
                <snmpv3_contextname/>
                <snmpv3_securityname/>
                <snmpv3_securitylevel>0</snmpv3_securitylevel>
                <snmpv3_authprotocol>0</snmpv3_authprotocol>
                <snmpv3_authpassphrase/>
                <snmpv3_privprotocol>0</snmpv3_privprotocol>
                <snmpv3_privpassphrase/>
                <params/>
                <ipmi_sensor/>
                <authtype>0</authtype>
                <username/>
                <password/>
                <publickey/>
                <privatekey/>
                <port/>
                <description>It requires user parameter mysql.ping, which is
defined in userparameter_mysql.conf.&#13;
&#13;;
0 - MySQL server is down&#13;
1 - MySQL server is up</description>
                <inventory_link>0</inventory_link>
                <applications>
                    <application>
                        <name>MySQL</name>
                    </application>
                </applications>
                <valuemap>
                    <name>Service state</name>
                </valuemap>
                <logtimefmt/>
                <preprocessing/>
                <jmx_endpoint/>
                <timeout>3s</timeout>
                <url/>
                <query_fields/>
                <posts/>
                <status_codes>200</status_codes>

```

```

<follow_redirects>1</follow_redirects>
<post_type>0</post_type>
<http_proxy/>
<headers/>
<retrieve_mode>0</retrieve_mode>
<request_method>1</request_method>
<output_format>0</output_format>
<allow_traps>0</allow_traps>
<ssl_cert_file/>
<ssl_key_file/>
<ssl_key_password/>
<verify_peer>0</verify_peer>
<verify_host>0</verify_host>
<master_item/>
</item>
<item>
    <name>MySQL begin operations per second</name>
    <type>0</type>
    <snmp_community/>
    <snmp_oid/>
    <key>mysql.status[Com_begin]</key>
    <delay>1m</delay>
    <history>1w</history>
    <trends>365d</trends>
    <status>0</status>
    <value_type>0</value_type>
    <allowed_hosts/>
    <units>qps</units>
    <snmpv3_contextname/>
    <snmpv3_securityname/>
    <snmpv3_securitylevel>0</snmpv3_securitylevel>
    <snmpv3_authprotocol>0</snmpv3_authprotocol>
    <snmpv3_authpassphrase/>
    <snmpv3_privprotocol>0</snmpv3_privprotocol>
    <snmpv3_privpassphrase/>
    <params/>
    <ipmi_sensor/>
    <authtype>0</authtype>
    <username/>
    <password/>
    <publickey/>
    <privatekey/>
    <port/>
    <description>It requires user parameter mysql.status[*], which is
defined in userparameter_mysql.conf.</description>
    <inventory_link>0</inventory_link>
    <applications>
        <application>
            <name>MySQL</name>
        </application>
    </applications>
    <valuemap/>
    <logtimefmt/>
    <preprocessing>
        <step>
            <type>10</type>
            <params/>
        </step>
    </preprocessing>
    <jmx_endpoint/>
    <timeout>3s</timeout>
    <url/>
    <query_fields/>
    <posts/>
    <status_codes>200</status_codes>
    <follow_redirects>1</follow_redirects>
    <post_type>0</post_type>
    <http_proxy/>
    <headers/>

```

```

<retrieve_mode>0</retrieve_mode>
<request_method>1</request_method>
<output_format>0</output_format>
<allow_traps>0</allow_traps>
<ssl_cert_file/>
<ssl_key_file/>
<ssl_key_password/>
<verify_peer>0</verify_peer>
<verify_host>0</verify_host>
<master_item/>
</item>
<item>
    <name>MySQL queries per second</name>
    <type>0</type>
    <snmp_community/>
    <snmp_oid/>
    <key>mysql.status[Questions]</key>
    <delay>1m</delay>
    <history>1w</history>
    <trends>365d</trends>
    <status>0</status>
    <value_type>0</value_type>
    <allowed_hosts/>
    <units>qps</units>
    <snmpv3_contextname/>
    <snmpv3_securityname/>
    <snmpv3_securitylevel>0</snmpv3_securitylevel>
    <snmpv3_authprotocol>0</snmpv3_authprotocol>
    <snmpv3_authpassphrase/>
    <snmpv3_privprotocol>0</snmpv3_privprotocol>
    <snmpv3_privpassphrase/>
    <params/>
    <ipmi_sensor/>
    <authtype>0</authtype>
    <username/>
    <password/>
    <publickey/>
    <privatekey/>
    <port/>
    <description>It requires user parameter mysql.status[*], which is
defined in userparameter_mysql.conf.</description>
    <inventory_link>0</inventory_link>
    <applications>
        <application>
            <name>MySQL</name>
        </application>
    </applications>
    <valuemap/>
    <logtimefmt/>
    <preprocessing>
        <step>
            <type>10</type>
            <params/>
        </step>
    </preprocessing>
    <jmx_endpoint/>
    <timeout>3s</timeout>
    <url/>
    <query_fields/>
    <posts/>
    <status_codes>200</status_codes>
    <follow_redirects>1</follow_redirects>
    <post_type>0</post_type>
    <http_proxy/>
    <headers/>
    <retrieve_mode>0</retrieve_mode>
    <request_method>1</request_method>
    <output_format>0</output_format>
    <allow_traps>0</allow_traps>

```

```

<ssl_cert_file/>
<ssl_key_file/>
<ssl_key_password/>
<verify_peer>0</verify_peer>
<verify_host>0</verify_host>
<master_item/>
</item>
</items>
<discovery_rules/>
<httptests/>
<macros/>
<templates/>
<screens>
<screen>
<name>MySQL performance</name>
<hsize>2</hsize>
<vsize>1</vsize>
<screen_items>
<screen_item>
<resourcetype>0</resourcetype>
<width>500</width>
<height>200</height>
<x>0</x>
<y>0</y>
<colspan>1</colspan>
<rowspan>1</rowspan>
<elements>0</elements>
<valign>1</valign>
<halign>0</halign>
<style>0</style>
<url/>
<dynamic>0</dynamic>
<sort_triggers>0</sort_triggers>
<resource>
<name>MySQL operations</name>
<host>Template DB MySQL</host>
</resource>
<max_columns>3</max_columns>
<application/>
</screen_item>
</screen_items>
</screen>
</screens>
</template>
</templates>
<triggers>
<trigger>
<expression>{Template DB MySQL:mysql.ping.last(0)}=0</expression>
<recovery_mode>0</recovery_mode>
<recovery_expression/>
<name>MySQL is down</name>
<correlation_mode>0</correlation_mode>
<correlation_tag/>
<url/>
<status>0</status>
<priority>2</priority>
<description/>
<type>0</type>
<manual_close>0</manual_close>
<dependencies/>
<tags/>
</trigger>
</triggers>
<graphs>
<graph>
<name>MySQL operations</name>
<width>900</width>
<height>200</height>
<yaxismin>0.0000</yaxismin>

```

```

<yaxismax>100.0000</yaxismax>
<show_work_period>1</show_work_period>
<show_triggers>1</show_triggers>
<type>0</type>
<show_legend>1</show_legend>
<show_3d>0</show_3d>
<percent_left>0.0000</percent_left>
<percent_right>0.0000</percent_right>
<ymin_type_1>0</ymin_type_1>
<ymax_type_1>0</ymax_type_1>
<ymin_item_1>0</ymin_item_1>
<ymax_item_1>0</ymax_item_1>
<graph_items>
    <graph_item>
        <sortorder>0</sortorder>
        <drawtype>0</drawtype>
        <color>C8C800</color>
        <yaxisside>0</yaxisside>
        <calc_fnc>2</calc_fnc>
        <type>0</type>
        <item>
            <host>Template DB MySQL</host>
            <key>mysql.status[Com_begin]</key>
        </item>
    </graph_item>
</graph_items>
</graph>
</graphs>
<value_maps>
    <value_map>
        <name>Service state</name>
        <mappings>
            <mapping>
                <value>0</value>
                <newvalue>Down</newvalue>
            </mapping>
            <mapping>
                <value>1</value>
                <newvalue>Up</newvalue>
            </mapping>
        </mappings>
    </value_map>
</value_maps>
</zabbix_export>

```

Теги элементов

Значения тегов элементов описаны в таблице ниже.

Теги шаблона

Элемент	Свойство элемента	Тип	Диапазон	Описание
templates				Корневой элемент шаблонов.
template				Одиночный шаблон.
	template	string		Уникальное имя шаблона.
	name	string		Видимое имя шаблона.
	description	text		Описание шаблона.
groups				Корневой элемент групп узлов сети.
group				Одиночная группа узлов сети.
	name	строка		Уникальное имя группы.
applications				Корневой элемент групп элементов данных

Элемент	Свойство элемента	Тип	Диапазон	Описание
application	name			шаблона.
macros				Одиночная группа элементов данных шаблона.
macro				Имя группы элементов данных шаблона.
	name			Корневой элемент макросов шаблона.
	value			Одиночный макрос шаблона.
templates				Имя пользовательского макроса.
template				Значение пользовательского макроса.
	name	строка		Корневой элемент для присоединенных шаблонов.
				Одиночный шаблон.
				Имя шаблона.

Теги элемента данных шаблона

Элемент	Свойство элемента	Тип	Диапазон	Описание
items				Корневой элемент элементов данных.
item				Одиночный элемент данных.
	name	строка		Имя элемента данных.
	type	целое число	0 - Zabbix агент 1 - SNMPv1 агент 2 - Zabbix траппер 3 - простая проверка 4 - SNMPv2 агент 5 - внутренний 6 - SNMPv3 агент 7 - Zabbix агент (активный) 8 - агрегированный 9 - HTTP тест (шаг сценария веб-мониторинга) 10 - внешний 11 - монитор баз данных 12 - IPMI агент 13 - SSH агент 14 - Telnet агент 15 - вычисляемый 16 - JMX агент 17 - SNMP трап 18 - Зависимый элемент данных 19 - элемент данных HTTP агент	Тип элемента данных.
	snmp_community	строка		Имя SNMP community, если 'type' равен 1,4.
	snmp_oid	строка		ID SNMP объекта.
	key	строка		Ключ элемента данных.
	delay	строка		Интервал обновления элемента данных. Секунды, единицы времени с суффиксом, пользовательские интервалы,

Элемент	Свойство элемента	Тип	Диапазон	Описание
				пользовательские макросы или LLD макросы.
	history	строка		Единица времени как долго хранить данные истории.
	trends	строка		Единица времени с суффиксом, пользовательский макрос или LLD макросы.
	status	целое число	0 - активирован 1 - деактивирован	Единица времени как долго хранить данные динамики изменений. Единица времени с суффиксом, пользовательский макрос или LLD макросы.
	value_type	целое число	0 - числовое с плавающей точкой 1 - символ 2 - журнал (лог) 3 - числовое целое положительное 4 - текст	Состояние элемента данных.
	allowed_hosts	строка		Тип получаемого значения.
	units	строка		Список IP адресов (разделенных запятыми) узлов сети, с которых разрешена отправка данных на элемент данных, если 'type' равен 2 или 19.
	snmpv3_contextname	строка		Единицы измерений получаемых значений (bps, B).
	snmpv3_securityname	строка		Имя контекста SNMPv3.
	snmpv3_securitylevel	целое число	0 - noAuthNoPriv 1 - authNoPriv 2 - authPriv	Имя безопасности SNMPv3.
	snmpv3_authprotocol	целое число	0 - MD5 1 - SHA	Уровень безопасности SNMPv3.
	snmpv3_authpassphrase	строка		Протокол аутентификации SNMPv3.
	snmpv3_privprotocol	целое число	0 - DES 1 - AES	Фраза-пароль аутентификации SNMPv3.
	snmpv3_privpassphrase	строка		Протокол безопасности SNMPv3.
	params	текст		Фраза-пароль безопасности SNMPv3.
				Имя “Выполняемый скрипт”, если 'type' равен 13,14
				Поле “SQL запрос”, если 'type' равен 11
				Поле “Формула”, если 'type' равен 15

Элемент	Свойство элемента	Тип	Диапазон	Описание
	ipmi_sensor	строка		ID IPMI сенсора, если 'type' равен 12.
	authtype	целое число	Тип аутентификации для элементов данных типа SSH агент: 0 - пароль 1 - ключ	Тип аутентификации, если 'type' равен 13 или 19.
	username	строка		Имя пользователя, если 'type' равен 11,13,14,19.
	password	строка		Пароль , если 'type' равен 11,13,14,19.
	publickey	строка		Имя файла публичного ключа, если 'type' равен 13.
	privatekey	строка		Имя файла приватного ключа, если 'type' равен 13.
	port	строка		Пользовательский порт элемента данных.
	description	текст		Описание элемента данных.
	inventory_link	целое число	0 - ссылка отсутствует <i>номер</i> - номер поля в таблице 'host_inventory'	Использование значения элемента данных для заполнения указанного поля инвентарных данных.
	logtimefmt	строка		Формат времени в записях в журнале. Используется только журнал (лог) элементами данных.
	jmx_endpoint	string		JMX endpoint, если 'type' равен 16.
	url	string		Строка URL, 'type' равен 19.
	allow_traps	integer	0 - Не разрешать тралы. 1 - Разрешать тралы.	Свойство позволяет отправку данных на элемент данных, если 'type' равен 19.
	follow_redirects	integer	0 - Не следовать перенаправлениям. 1 - Следовать перенаправлениям.	Следование HTTP перенаправлениям, если 'type' равен 19.
	headers	object		Объект с заголовками HTTP(S) запроса, если 'type' равен 19.
	http_proxy	string		Строка подключения к HTTP(S) прокси, если 'type' равен 19.
	output_format	integer	0 - Записывать как есть. 1 - Конвертировать в JSON.	Каким образом обрабатывать ответ, если 'type' равен 19.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	post_type	integer	0 - Сырые данные. 2 - JSON данные. 3 - XML данные.	Тип тела запроса, если 'type' равен 19.
	posts	text		Тело запроса, если 'type' равен 19.
	query_fields	array		Массив объектов для запроса полей запроса, если 'type' равен 19.
	request_method	integer	0 - GET 1 - POST 2 - PUT 3 - HEAD	Метод запроса, если 'type' равен 19.
	retrieve_mode	integer	0 - Тело. 1 - Заголовки. 2 - Записывать как тело, так и заголовки.	Какую часть ответа нужно записывать, если 'type' равен 19.
	ssl_cert_file	string		Путь к файлу публичного ключа SSL, если 'type' равен 19.
	ssl_key_file	string		Путь к файлу приватного ключа SSL, если 'type' равен 19.
	ssl_key_password	string		Пароль к файлу SSL ключа, если 'type' равен 19.
	status_codes	string		Диапазоны требуемых кодов HTTP состояний, разделенные запятыми, если 'type' равен 19.
	timeout	string		Время ожидания при выполнении запроса данных элемента данных, если 'type' равен 19.
	verify_host	integer	0 - Не проверять. 1 - Проверять.	Проверка имени узла в URL в полях Common Name тлт Subject Alternate Name сертификата хоста, если 'type' равен 19.
	verify_peer	integer	0 - Не проверять. 1 - Проверять.	Проверка подлинности сертификата хоста, если 'type' равен 19.
value map				Преобразование значений.
	name	строка		Имя преобразования значений, которое используется элементом данных.
applications				Корневой элемент групп элементов данных.
application				Одиночная группа элементов данных.
	name			Имя группы элементов данных.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	preprocessing			Предварительная обработка элемента данных.
	step			Одиночный шаг предварительной обработки значений элемента данных.
	type	целое число	1 - пользовательский множитель 2 - обрезка справа 3 - обрезка слева 4 - обрезка с обеих сторон 5 - соответствие регулярному выражению 6 - двоичное в десятичное 7 - восьмеричное в десятичное 8 - шестнадцатеричное в десятичное 9 - простое изменение; вычисляется как (полученное значение-пред. значение) 10 - изменение в секунду; вычисляется как (полученное значение-пред. значение)/(время сейчас-время посл. проверки)	Тип шага предварительной обработки значений элемента данных.
	params	строка		Параметры шага предварительной обработки значений элемента данных.
master_item				Одиночный элемент данных мастер элемента данных.
	key	строка		Значение ключа мастер элемента данных зависимого элемента данных.

Теги правил низкоуровневого обнаружения шаблона

Элемент	Свойство элемента	Тип	Диапазон	Описание
discovery_rules				Корневой элемент правил низкоуровневого обнаружения.
discovery_rule				Одиночное правильно низкоуровневого обнаружения.

По большинству значений тегов элементов,смотрите значения тегов элементов обычных элементов данных. Ниже описаны только те теги, которые являются специфичными для правил низкоуровневого обнаружения.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	lifetime	строка		Период времени после которого элементы данных, которые более не обнаруживаются, будут удалены. Секунды, единицы времени с суффиксом или пользовательский макрос.
filter				Одиночный фильтр.
	evaltype	целое число	0 - логика И/ИЛИ 1 - логика И 2 - логика ИЛИ 3 - пользовательская формула	Логика использования условий фильтра для проверки правила низкоуровневого обнаружения.
	formula	строка		Пользовательская формула вычисления условий фильтрации.
	conditions			Корневой элемент условий фильтров.
condition				Одиночное условие фильтра.
	macro	строка		Имя макроса низкоуровневого обнаружения.
	value	строка		Значение фильтра: регулярное выражение или глобальное регулярное выражение.
	operator	целое число		
	formulaid	СИМВОЛ		ID условия фильтра. Используется в пользовательской формуле вычисления.
item_prototype				Корневой элемент item_prototypes.
pes				
item_prototype				Одиночный item_prototype.
pe				
	<i>По большинству значений тегов элементов,смотрите значения тегов элементов обычных элементов данных. Ниже описаны только те теги, которые являются специфичными для item_prototypes.</i>			
application_prototype				Корневой элемент прототипов групп элементов данных.
application_prototype				Одиночный прототип групп

Элемент	Свойство элемента	Тип	Диапазон	Описание
prototype	name			элементов данных. Имя прототипа групп элементов данных.
master_item	key	строка		Одиночный прототип элементов данных прототипа мастер элементов данных/прототипа элементов данных.

Теги триггеров шаблона

Элемент	Свойство элемента	Тип	Диапазон	Описание
triggers				Корневой элемент триггеров.
trigger	expression	строка		Одиночный триггер.
	recovery_mode	целое число	0 - выражение 1 - выражение восстановления 2 - нет	Выражение триггера.
	recovery_expression	строка		Основа генерации ОК событий.
	name	строка		Выражение восстановления триггера.
	correlation_mode	целое число	0 - без корреляции событий 1 - корреляция событий по тегу	Имя триггера.
	correlation_tag	строка		Режим корреляции.
	url	строка		Используемое имя тега для корреляции событий.
	status	целое число	0 - активирован 1 - деактивирован	URL триггера .
			0 - не классифицировано 1 - информация	Состояние триггера.
	priority	целое число	2 - предупреждение 3 - средняя 4 - высокая 5 - чрезвычайная	4 - высокая 5 - чрезвычайная
	description	текст		Важность триггера.
	type	целое число	0 - одно событие о проблеме 1 - множественные события о проблемах	Описание триггера.
	manual_close	целое число	0 - не разрешена 1 - разрешена	Тип генерации событий.
dependencies				Закрытие вручную событий о проблемах.
				Корневой элемент

Элемент	Свойство элемента	Тип	Диапазон	Описание
dependency	name	строка		зависимостей.
	expression	строка		Одиночная зависимость.
	recovery_expression	строка		Имя зависимого триггера.
tags				Выражение восстановления зависимого триггера.
tag	tag	строка		Выражение восстановления зависимого триггера.
	value	строка		Корневой элемент тегов событий.
tag				Одиночный тег событий.
				Имя тега.
				Значение тега.

Теги графиков шаблона

Элемент	Свойство элемента	Тип	Диапазон	Описание
graphs				Корневой элемент графиков.
graph				Одиночный график.
	name	строка		Имя графика.
	width	целое число		Ширина графика, в пикселях.
	height	целое число		Используется для предпросмотра и в круговом/расширенном графиках.
	yaxismin	дробное число		Высота графика, в пикселях.
	yaxismax	дробное число		Используется для предпросмотра и в круговом/расширенном графиках.
	show_work_period	целое число	0 - нет 1 - да	Минимальное значение оси Y, если 'ymin_type_1' равен 1.
	show_triggers	целое число	0 - нет 1 - да	Максимальное значение оси Y, если 'ymin_type_1' равен 1.
	type	целое число	0 - нормальный 1 - стэкируемый 2 - круговой 3 - расширенный 4 - 3D круговой 5 - 3D расширенный	Подсветка нерабочих часов, если 'type' равен 0,1.
	show_legend	целое число	0 - нет 1 - да	Отображение значений простых триггеров в виде линии, если 'type' равен 0,1.
	show_3d	целое число	0 - 2D 1 - 3D	Отображение значений простых триггеров в виде линии, если 'type' равен 0,1.
	percent_left	дробное число		Тип графика.
	percent_right	дробное число		
				Отображение легенды графика.
				Активация 3D стиля, если 'type' равен 2,3.
				Отображение процентной линии для левой стороны оси X, если 'type' равен 0.
				Отображение процентной линии для левой стороны оси X, если 'type'

Элемент	Свойство элемента	Тип	Диапазон	Описание
	уmin_type_1	целое число	0 - вычисляется 1 - фиксировано 2 - последнее значение выбранного элемента данных	равен 0. Минимальное значение оси Y, если 'type' равен 0,1.
	умакс_type_1	целое число	0 - вычисляется 1 - фиксировано 2 - последнее значение выбранного элемента данных	Максимальное значение оси Y, если 'type' равен 0,1.
	уmin_item_1	строка	null или детали элемента данных	Детали элемента данных, если 'уmin_type_1' равен 2.
	умакс_item_1	строка	null или детали элемента данных	Детали элемента данных, если 'умакс_type_1' равен 2.
graph_items				Корневой элемент данных графиков.
graph_item	sortorder	целое число		Одиночный элемент данных графика.
	drawtype	целое число	0 - одинарная линия 1 - заполнение региона 2 - жирная линия 3 - точечная линия 4 - пунктирная линия	Порядок отрисовки. Меньшее значение отрисовывается первым. Можно использовать для отрисовки линий или регионов позади (или спереди) других.
	color	строка		Стиль отрисовки, если 'type' графика равен 0.
	yaxisside	целое число	0 - ось слева 1 - ось справа	Цвет элемента (6 символов, hex). Позиция оси Y (слева или справа), которой принадлежит элемент, если 'type' графика равен 0,1.
	calc_fnc	целое число	1 - минимум 2 - среднее 4 - максимум 7 - все (минимум, среднее и максимум, если 'type' графика равен 0) 9 - последнее (если 'type' графика не равен 0,1)	Данные для отрисовки, если существует более одного значения по элементу данных.
	type	целое число	1 - значение элемента данных представляется пропорционально на круге 2 - значение элемента данных представляется целым кругом (сумма графика)	Тип отрисовки для круговых/расширенных графиков.
item				Одиночный элемент данных.
	host	строка		Узел сети элемента данных.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	key	строка		Ключ элемента данных.
Теги веб-сценария шаблона				
Элемент	Свойство элемента	Тип	Диапазон	Описание
http tests				Корневой элемент веб-сценариев.
http test				Одиночный веб-сценарий.
	name	строка		Имя веб-сценария.
	delay	строка		Частота выполнения веб-сценария. Секунды, единицы времени с суффиксом или пользовательский макрос.
	attempts	целое число	1-10	Количество попыток выполнения шагов веб-сценария.
	agent	строка		Агент клиента. Zabbix будет притворяться выбранным браузером. Эта опция полезна, когда веб-сайт возвращает разное содержимое для разных браузеров.
	http_proxy	строка		Укажите используемый HTTP прокси, использую формат: <code>http://[имя_пользователя[:пароль]@]прокси.моякомпания.ком[:порт]</code>
	variables	текст		Список переменных (макросов) уровня веб-сценария, которые можно затем использовать в шагах сценария.
	headers	текст		HTTP заголовки, которые необходимо отправлять при выполнении запроса.
	status	целое число	0 - активировано 1 - деактивировано	Состояние веб-сценария.
	authentication	целое число	0 - нет 1 - простая 2 - NTLM	Метод аутентификации.
	http_user	строка		Имя пользователя для аутентификации.
	http_password	строка		Пароль для аутентификации.
	verify_peer	целое число	0 - нет 1 - да	Проверка SSL сертификата веб-сервера.
	verify_host	целое число	0 - нет 1 - да	Проверка, что поле Общее Имя или поле Дополнительное Имя Субъекта сертификата веб-сервера совпадают.
	ssl_cert_file	строка		Имя используемого файла SSL сертификата для аутентификации клиента.
	ssl_key_file	строка		Имя используемого файла приватного ключа SSL для аутентификации клиента.
	ssl_key_password	строка		Пароль к файлу приватного ключа SSL.
steps				Корневой элемент шагов веб-сценариев.
step				Одиночный шаг веб-сценария.
	name	строка		Имя шага веб-сценария.
	url	строка		Наблюдаемый URL.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	posts	текст		Список 'Post' переменных.
	variables	текст		Список переменных (макросов) уровня шага, которые применяются после выполнения этого шага.
	headers	текст		Если значение переменной имеет префикс 'regex:', тогда ее значение извлекается из данных, которые получены на этом шаге, в соответствии с шаблоном регулярного выражения следующим за 'regex:' префиксом HTTP заголовки, которые необходимо отправлять при выполнении запроса.
	follow_redirects	целое число	0 - нет 1 - да	Следовать HTTP перенаправлениям.
	retrieve_mode	целое число	0 - содержимое 1 - только заголовки	Режим получения HTTP ответа.
	timeout	строка		Время выполнения шага. Секунды, единицы времени с суффиксом или пользовательский макрос.
	required	строка		Требуемая строка. Игнорируется, если пустая.
	status_codes	строка		Список допустимых кодов состояний разделенных запятыми. Игнорируется, если пустой. Например: 200-201,210-299

14.3 Узлы сети

Обзор

Узлы сети [экспортируются](#) вместе со множеством связанных объектов и со связями с объектами.

Экспорт узла сети содержит:

- присоединенные группы узлов сети
- данные узла сети
- связи с шаблонами
- связи с группами узлов сети
- интерфейсы узла сети
- напрямую присоединенные группы элементов данных
- напрямую присоединенные элементы данных
- напрямую присоединенные триггеры
- напрямую присоединенные графики
- напрямую присоединенные правила обнаружения со всеми прототипами
- напрямую присоединенные веб-сценарии
- макросы узла сети
- инвентарные данные узла сети
- преобразования значений

Когда узел сети импортирован и обновлен, он может быть присоединен только к дополнительным шаблонам и никогда не отсоединен от них.

Экспорт

Для экспорта узлов сети сделайте следующее:

- Перейдите в: *Настройки* → *Узлы сети*
- Отметьте узлы сети, которые нужно экспортировать
- Нажмите на *Экспорт* ниже списка

The screenshot shows the 'Hosts' section of the Zabbix interface. At the top, there's a header with columns: Name, Applications, Items, Triggers, Graphs, Discovery, and View. Below the header, two hosts are selected: 'Zabbix server' and 'Zabbix host'. Both hosts have their respective details listed under their names. At the bottom of the list, there are several buttons: 'Enable', 'Disable', 'Export' (which has a cursor over it), 'Mass update', and 'Delete'.

Выбранные узлы сети экспортируются в локальный XML файл с именем по умолчанию `zabbix_export_hosts.xml`.

Импорт

Для импорта узлов сети сделайте следующее:

- Перейдите в: *Настройка* → *Узлы сети*
- Нажмите на *Импорт* справа
- Выберите импортируемый файл
- Отметьте необходимые опции в правилах импорта
- Нажмите на *Импорт*

* Import file No file selected.

Rules	Update existing	Create new	Delete missing
Groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Hosts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Templates	<input type="checkbox"/>	<input type="checkbox"/>	
Template screens	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Template linkage	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applications	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Graphs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Web scenarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Screens	<input type="checkbox"/>	<input type="checkbox"/>	
Maps	<input type="checkbox"/>	<input type="checkbox"/>	
Images	<input type="checkbox"/>	<input type="checkbox"/>	
Media types	<input type="checkbox"/>	<input type="checkbox"/>	
Value mappings	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Все обязательные поля ввода отмечены красной звёздочкой.

После импорта в веб-интерфейсе отобразится сообщение об успехе или об ошибке.

Правила импорта:

Правило

Обновить существующее

Существующие элементы будут обновлены данными взятыми с файла импорта. В противном случае они не будут обновляться.

Создать новое

Импорт добавит новые элементы, используя данные из файла импорта. В противном случае они не будут добавляться.

Удалить пропущенное

Импорт удалит существующие элементы, которые отсутствуют в файле импорта. В противном случае они не будут удаляться.

Описание

Формат экспорта

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>4.0</version>
    <date>2018-04-03T08:11:42Z</date>
    <groups>
        <group>
            <name>Zabbix servers</name>
        </group>
    </groups>
    <hosts>
        <host>
            <host>Export host</host>
            <name>Export host</name>
        </host>
    </hosts>
```

```

<description/>
<proxy/>
<status>0</status>
<ipmi_authtype>-1</ipmi_authtype>
<ipmi_privilege>2</ipmi_privilege>
<ipmi_username/>
<ipmi_password/>
<tls_connect>1</tls_connect>
<tls_accept>1</tls_accept>
<tls_issuer/>
<tls_subject/>
<tls_psk_identity/>
<tls_psk/>
<templates/>
<groups>
    <group>
        <name>Zabbix servers</name>
    </group>
</groups>
<interfaces>
    <interface>
        <default>1</default>
        <type>1</type>
        <useip>1</useip>
        <ip>127.0.0.1</ip>
        <dns/>
        <port>10050</port>
        <bulk>1</bulk>
        <interface_ref>if1</interface_ref>
    </interface>
</interfaces>
<applications>
    <application>
        <name>Application</name>
    </application>
</applications>
<items>
    <item>
        <name>Item</name>
        <type>0</type>
        <snmp_community/>
        <snmp_oid/>
        <key>item.key</key>
        <delay>30s</delay>
        <history>90d</history>
        <trends>365d</trends>
        <status>0</status>
        <value_type>3</value_type>
        <allowed_hosts/>
        <units/>
        <snmpv3_contextname/>
        <snmpv3_securityname/>
        <snmpv3_securitylevel>0</snmpv3_securitylevel>
        <snmpv3_authprotocol>0</snmpv3_authprotocol>
        <snmpv3_authpassphrase/>
        <snmpv3_privprotocol>0</snmpv3_privprotocol>
        <snmpv3_privpassphrase/>
        <params/>
        <ipmi_sensor/>
        <authtype>0</authtype>
        <username/>
        <password/>
        <publickey/>
        <privatekey/>
        <port/>
        <description/>
        <inventory_link>0</inventory_link>
        <applications>
            <application>

```

```

        <name>Application</name>
    </application>
</applications>
<valuemap>
    <name>Host status</name>
</valuemap>
<logtimefmt/>
<preprocessing/>
<jmx_endpoint/>
<timeout>3s</timeout>
<url/>
<query_fields/>
<posts/>
<status_codes>200</status_codes>
<follow_redirects>1</follow_redirects>
<post_type>0</post_type>
<http_proxy/>
<headers/>
<retrieve_mode>0</retrieve_mode>
<request_method>1</request_method>
<output_format>0</output_format>
<allow_traps>0</allow_traps>
<ssl_cert_file/>
<ssl_key_file/>
<ssl_key_password/>
<verify_peer>0</verify_peer>
<verify_host>0</verify_host>
<master_item/>
    <interface_ref>if1</interface_ref>
</item>
</items>
<discovery_rules/>
<httptests/>
<macros/>
<inventory/>
</host>
</hosts>
<triggers>
    <trigger>
        <expression>{Export host:item.key.last()}</expression>
        <recovery_mode>0</recovery_mode>
        <recovery_expression/>
        <name>Item value not 0</name>
        <correlation_mode>0</correlation_mode>
        <correlation_tag/>
        <url/>
        <status>0</status>
        <priority>2</priority>
        <description/>
        <type>0</type>
        <manual_close>0</manual_close>
        <dependencies/>
        <tags/>
    </trigger>
</triggers>
<graphs>
    <graph>
        <name>Example graph</name>
        <width>900</width>
        <height>200</height>
        <yaxismin>0.0000</yaxismin>
        <yaxismax>100.0000</yaxismax>
        <show_work_period>1</show_work_period>
        <show_triggers>1</show_triggers>
        <type>0</type>
        <show_legend>1</show_legend>
        <show_3d>0</show_3d>
        <percent_left>0.0000</percent_left>
        <percent_right>0.0000</percent_right>
    </graph>
</graphs>

```

```

<ymin_type_1>0</ymin_type_1>
<ymax_type_1>0</ymax_type_1>
<ymin_item_1>0</ymin_item_1>
<ymax_item_1>0</ymax_item_1>
<graph_items>
    <graph_item>
        <sortorder>0</sortorder>
        <drawtype>0</drawtype>
        <color>1A7C11</color>
        <yaxisside>0</yaxisside>
        <calc_fnc>2</calc_fnc>
        <type>0</type>
        <item>
            <host>Export host</host>
            <key>item.key</key>
        </item>
    </graph_item>
</graph_items>
</graph>
</graphs>
<value_maps>
    <value_map>
        <name>Host status</name>
        <mappings>
            <mapping>
                <value>0</value>
                <newvalue>Up</newvalue>
            </mapping>
            <mapping>
                <value>2</value>
                <newvalue>Unreachable</newvalue>
            </mapping>
        </mappings>
    </value_map>
</value_maps>
</zabbix_export>

```

Теги элементов

Значения тегов элементов описаны в таблице ниже.

Теги узлов сети

Элемент	Свойство элемента	Тип	Диапазон	Описание
groups				Корневой элемент для групп узлов сети.
group				Одиночная группа узлов сети.
	name	строка		Уникальное имя группы узлов сети.
hosts				Корневой элемент для узлов сети.
host				Одиночный узел сети.
	host	строка		Уникальное имя узла сети.
	name	строка		Видимое имя узла сети.
	description	текст		Описание узла сети.
	status	целое число	0 - под наблюдением 1 - без наблюдения	Состояние узла сети.
	ipmi_authtype	целое число	-1 - по умолчанию 0 - none 1 - MD2 2 - MD5	Алгоритм аутентификации IPMI.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	ipmi_privilege	целое число	4 - straight 5 - OEM 6 - RMCP+ 1 - callback 2 - пользователь 3 - оператор 4 - админ 5 - OEM	Уровень привилегий IPMI.
	ipmi_username	строка		Имя пользователя для IPMI проверок.
	ipmi_password	строка		Пароль для IPMI проверок.
	tls_connect	целое число	1 - без шифрования 2 - TLS с PSK 4 - TLS с сертификатом 1 - без шифрования 2 - TLS с PSK 3 - без шифрования и TLS с PSK 4 - TLS с сертификатом 5 - без шифрования и TLS с сертификатом 6 - TLS с PSK или сертификат 7 - без шифрования и TLS с PSK или сертификат	Тип исходящих соединений.
	tls_accept	целое число		Тип входящих соединений.
	tls_issuer	строка		Разрешенный эмитент сертификата агента/прокси.
	tls_subject	строка		Разрешенный субъект сертификата агента/прокси.
	tls_psk_identity	строка		Строка идентификатор PSK.
	tls_psk	строка		Строка значения PSK.
proxy				Прокси.
	name	строка		Имя прокси (если имеется), которое наблюдает узел сети.
templates				Корневой элемент для присоединенных шаблонов.
template				Одиночный шаблон.
	name	строка		Имя шаблона.
interfaces				Корневой элемент интерфейсов узла сети.
interface				Одиночный интерфейс.
	default	целое число	0 - вторичный 1 - первичный (по умолчанию)	Состояние интерфейса. У узла сети может быть только один первичный интерфейс каждого типа.
	type	целое число	0 - неизвестно 1 - Zabbix агент	Тип интерфейса.

Элемент	Свойство элемента	Тип	Диапазон	Описание
			2 - SNMP 3 - IPMI 4 - JMX	
useip	целое число	0 - использовать DNS имя 1 - использовать IP адрес		Используемый вариант подключения к узлу сети.
ip	строка			IP адрес, может быть как IPv4, так и IPv6.
dns	строка			DNS имя.
port	строка			Номер порта.
bulk	целое число	0 - деактивировано 1 - активировано		Необходимо ли использовать массовые запросы SNMP
interface_ref	строка			Имя ссылки на интерфейс, которое используется в элементах данных.
applications				Корневой элемент групп элементов данных.
application				Одиночная группа элементов данных.
	name			Имя группы элементов данных.
macros				Корневой элемент макросов.
macro				Одиночный макрос.
	name			Имя пользовательского макроса.
	value			Значение пользовательского макроса.

Теги элементов данных узла сети

Элемент	Свойство элемента	Тип	Диапазон	Описание
items				Корневой элемент элементов данных.
item				Одиночный элемент данных.
	name	строка		Имя элемента данных.
	type	целое число	0 - Zabbix агент 1 - SNMPv1 агент 2 - Zabbix траппер 3 - простая проверка 4 - SNMPv2 агент 5 - внутренний 6 - SNMPv3 агент 7 - Zabbix агент (активный) 8 - агрегированный 9 - HTTP тест (шаг сценария веб-мониторинга) 10 - внешний 11 - монитор баз данных 12 - IPMI агент 13 - SSH агент 14 - Telnet агент 15 - вычисляемый 16 - JMX агент 17 - SNMP трап	Тип элемента данных.

Элемент	Свойство элемента	Тип	Диапазон	Описание
			18 - Зависимый элемент данных 19 - элемент данных HTTP агент	
	snmp_community	строка		Имя SNMP community, если 'type' равен 1,4.
	snmp_oid	строка		ID SNMP объекта.
	key	строка		Ключ элемента данных.
	delay	строка		Интервал обновления элемента данных. Секунды, единицы времени с суффиком, пользовательские интервалы или пользовательские макросы.
	history	строка		Единица времени как долго хранить данные истории. Единица времени с суффиксом или пользовательский макрос.
	trends	строка		Единица времени как долго хранить данные динамики изменений. Единица времени с суффиксом или пользовательский макрос.
	status	целое число	0 - активирован 1 - деактивирован	Состояние элемента данных.
			0 - числовое с плавающей точкой	
	value_type	целое число	1 - символ 2 - журнал (лог) 3 - числовое целое положительное 4 - текст	Тип получаемого значения.
	allowed_hosts	строка		Список IP адресов (разделенных запятыми) узлов сети, с которых разрешена отправка данных на элемент данных, если 'type' равен 2 или 19.
	units	строка		Единицы измерений получаемых значений (bps, B).
	snmpv3_contextname	строка		Имя контекста SNMPv3.
	snmpv3_securityname	строка		Имя безопасности SNMPv3.
	snmpv3_securitylevel	целое число	0 - noAuthNoPriv 1 - authNoPriv 2 - authPriv	Уровень безопасности SNMPv3.
	snmpv3_authprotocol	целое число	0 - MD5 1 - SHA	Протокол аутентификации SNMPv3.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	snmpv3_authpa ssphrase	строка		Фраза-пароль аутентификации SNMPv3.
	snmpv3_privpr otocol	целое число	0 - DES 1 - AES	Протокол безопасности SNMPv3.
	snmpv3_privpa ssphrase	строка		Фраза-пароль безопасности SNMPv3.
	params	текст		Имя “Выполнляемый скрипт”, если 'type' равен 13,14 Поле “SQL запрос”, если 'type' равен 11 Поле “Формула”, если 'type' равен 15.
	ipmi_sensor	строка		ID IPMI сенсора, если 'type' равен 12.
	authtype	целое число	Тип аутентификации для элементов данных типа SSH агент: 0 - пароль 1 - ключ	Тип аутентификации, если 'type' равен 13 или 19.
	username	строка	Тип аутентификации для элементов данных HTTP агента: 0 - отсутствует 1 - простой 2 - NTLM	Имя пользователя, если 'type' равен 11,13,14,19.
	password	строка		Пароль , если 'type' равен 11,13,14,19.
	publickey	строка		Имя файла публичного ключа, если 'type' равен 13.
	privatekey	строка		Имя файла приватного ключа, если 'type' равен 13.
	port	строка		Пользовательский порт элемента данных.
	description	текст		Описание элемента данных.
	inventory_link	целое число	0 - ссылка отсутствует номер - номер поля в таблице 'host_inventory'	Использование значения элемента данных для заполнения указанного поля инвентарных данных.
	logtimefmt	строка		Формат времени в записях в журнале. Используется только журнал (лог) элементами данных.
	interface_ref	строка		Ссылка на интерфейс узла сети.
	jmx_endpoint	string		JMX endpoint, если 'type' равен 16.
	url	string		Строка URL, 'type' равен 19.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	allow_traps	integer	0 - Не разрешать трэпы. 1 - Разрешать трэпы.	Свойство позволяет отправку данных на элемент данных, если 'type' равен 19.
	follow_redirects	integer	0 - Не следовать перенаправлениям. 1 - Следовать перенаправлениям.	Следование HTTP перенаправлениям, если 'type' равен 19.
	headers	object		Объект с заголовками HTTP(S) запроса, если 'type' равен 19.
	http_proxy	string		Строка подключения к HTTP(S) прокси, если 'type' равен 19.
	output_format	integer	0 - Записывать как есть. 1 - Конвертировать в JSON.	Каким образом обрабатывать ответ, если 'type' равен 19.
	post_type	integer	0 - Сырые данные. 2 - JSON данные. 3 - XML данные.	Тип тела запроса, если 'type' равен 19.
	posts	text		Тело запроса, если 'type' равен 19.
	query_fields	array		Массив объектов для запроса полей запроса, если 'type' равен 19.
	request_method	integer	0 - GET 1 - POST 2 - PUT 3 - HEAD	Метод запроса, если 'type' равен 19.
	retrieve_mode	integer	0 - Тело. 1 - Заголовки. 2 - Записывать как тело, так и заголовки.	Какую часть ответа нужно записывать, если 'type' равен 19.
	ssl_cert_file	string		Путь к файлу публичного ключа SSL, если 'type' равен 19.
	ssl_key_file	string		Путь к файлу приватного ключа SSL, если 'type' равен 19.
	ssl_key_password	string		Пароль к файлу SSL ключа, если 'type' равен 19.
	status_codes	string		Диапазоны требуемых кодов HTTP состояний, разделенные запятыми, если 'type' равен 19.
	timeout	string		Время ожидания при выполнении запроса данных элемента данных, если 'type' равен 19.
	verify_host	integer	0 - Не проверять. 1 - Проверять.	Проверка имени узла в URL в полях Common Name или Subject Alternate Name сертификата хоста, если 'type'

Элемент	Свойство элемента	Тип	Диапазон	Описание
	verify_peer	integer	0 - Не проверять. 1 - Проверять.	равен 19. Проверка подлинности сертификата хоста, если 'type' равен 19.
value map				Преобразование значений.
	name	строка		Имя преобразования значений, которое используется элементом данных.
applications				Корневой элемент групп элементов данных.
application				Одиночная группа элементов данных.
	name			Имя группы элементов данных.
preprocessing				Предварительная обработка элемента данных.
step				Одиночный шаг предварительной обработки значений элемента данных.
	type	целое число	1 - пользовательский множитель 2 - обрезка справа 3 - обрезка слева 4 - обрезка с обеих сторон 5 - соответствие регулярному выражению 6 - двоичное в десятичное 7 - восьмеричное в десятичное 8 - шестнадцатеричное в десятичное 9 - простое изменение; вычисляется как (полученное значение-пред. значение) 10 - изменение в секунду; вычисляется как (полученное значение-пред. значение)/(время сейчас-время посл. проверки)	Тип шага предварительной обработки значений элемента данных.
	params	строка		Параметры шага предварительной обработки значений элемента данных.
master_item				Одиночный элемент данных мастер элемента данных.
	key	строка		Значение ключа мастер элемента данных зависимого элемента данных.

Теги правил низкоуровневого обнаружения узла сети

Элемент	Свойство элемента	Тип	Диапазон	Описание
discovery_rules				Корневой элемент правил низкоуровневого обнаружения.
discovery_rule				Одиночное правило низкоуровневого обнаружения.
	<i>По большинству значений тегов элементов, смотрите значения тегов элементов обычных элементов данных. Ниже описаны только те теги, которые являются специфичными для правил низкоуровневого обнаружения.</i>			
	lifetime	строка		Период времени после которого элементы данных, которые более не обнаруживаются, будут удалены. Секунды, единицы времени с суффиксом или пользовательский макрос.
filter				Одиночный фильтр.
	evaltype	целое число	0 - логика И/ИЛИ 1 - логика И 2 - логика ИЛИ 3 - пользовательская формула	Логика использования условий фильтра для проверки правила низкоуровневого обнаружения.
	formula	строка		Пользовательская формула вычисления условий фильтрации.
	conditions			Корневой элемент условий фильтров.
condition				Одиночное условие фильтра.
	macro	строка		Имя макроса низкоуровневого обнаружения.
	value	строка		Значение фильтра: регулярное выражение или глобальное регулярное выражение.
	operator	целое число		
	formulaid	символ		ID условия фильтра. Используется в пользовательской формуле вычисления.

Элемент	Свойство элемента	Тип	Диапазон	Описание
item_prototypes				Корневой элемент item_prototypes.
item_prototype				Одиночный item_prototype.
	По большинству значений тегов элементов,смотрите значения тегов элементов обычных элементов данных. Ниже описаны только те теги, которые являются специфичными для item_prototypes.			
application_prototypes				Корневой элемент прототипов групп элементов данных.
application_prototype				Одиночный прототип групп элементов данных.
	name			Имя прототипа групп элементов данных.
master_item				Одиночный прототип элементов данных/прототипа мастер элементов данных/прототипа элементов данных.
	key	строка		Значение ключа прототипа мастер элементов данных/прототипа элементов данных зависимого прототипа элементов данных.

Теги триггеров узла сети

Элемент	Свойство элемента	Тип	Диапазон	Описание
triggers				Корневой элемент триггеров.
trigger				Одиночный триггер.
	expression	строка		Выражение триггера.
	recovery_mode	целое число	0 - выражение 1 - выражение восстановления 2 - нет	Основа генерации ОК событий.
	recovery_expression	строка		Выражение восстановления триггера.
	name	строка		Имя триггера.
	correlation_mode	целое число	0 - без корреляции событий 1 - корреляция событий по тегу	Режим корреляции.
	correlation_tag	строка		Используемое имя тега для корреляции событий.
	url	строка		URL триггера .

Элемент	Свойство элемента	Тип	Диапазон	Описание
	status	целое число	0 - активирован 1 - деактивирован 0 - не классифицировано 1 - информация 2 - предупреждение 3 - средняя 4 - высокая 5 - чрезвычайная	Состояние триггера.
	priority	целое число	2 - предупреждение 3 - средняя 4 - высокая 5 - чрезвычайная	Важность триггера.
	description	текст		Описание триггера.
	type	целое число	0 - одно событие о проблеме 1 - множественные события о проблемах	Тип генерации событий.
	manual_close	целое число	0 - не разрешена 1 - разрешена	Закрытие вручную событий о проблемах.
dependencies				Корневой элемент зависимостей.
dependency				Одиночная зависимость.
	name	строка		Имя зависимого триггера.
	expression	строка		Выражение зависимого триггера.
	recovery_expression	строка		Выражение восстановления зависимого триггера.
tags				Корневой элемент тегов событий.
tag				Одиночный тег событий.
	tag	строка		Имя тега.
	value	строка		Значение тега.

Теги графиков узла сети

Элемент	Свойство элемента	Тип	Диапазон	Описание
graphs				Корневой элемент графиков.
graph				Одиночный график.
	name	строка		Имя графика.
	width	целое число		Ширина графика, в пикселях. Используется для предпросмотра и в круговом/расширенном графиках.
	height	целое число		Высота графика, в пикселях. Используется для предпросмотра и в круговом/расширенном графиках.
	yaxismin	дробное число		Минимальное значение оси Y, если 'ymin_type_1' равен 1.
	yaxismax	дробное число		Максимальное значение оси Y, если 'ymin_type_1' равен 1.
	show_work_period	целое число	0 - нет 1 - да	Подсветка нерабочих часов, если 'type' равен 0,1.
	show_triggers	целое число	0 - нет 1 - да	Отображение значений простых триггеров в виде линии, если 'type'

Элемент	Свойство элемента	Тип	Диапазон	Описание
				равен 0,1.
type	целое число		0 - нормальный 1 - стэкируемый 2 - круговой 3 - расширенный 4 - 3D круговой 5 - 3D расширенный	Тип графика.
show_legend	целое число		0 - нет 1 - да	Отображение легенды графика.
show_3d	целое число		0 - 2D 1 - 3D	Активация 3D стиля, если 'type' равен 2,3.
percent_left	дробное число			Отображение процентной линии для левой стороны оси X, если 'type' равен 0.
percent_right	дробное число			Отображение процентной линии для левой стороны оси X, если 'type' равен 0.
ymin_type_1	целое число		0 - вычисляется 1 - фиксировано 2 - последнее значение выбранного элемента данных	Минимальное значение оси Y, если 'type' равен 0,1.
ymax_type_1	целое число		0 - вычисляется 1 - фиксировано 2 - последнее значение выбранного элемента данных	Максимальное значение оси Y, если 'type' равен 0,1.
ymin_item_1	строка		null или детали элемента данных	Детали элемента данных, если 'ymin_type_1' равен 2.
ymax_item_1	строка		null или детали элемента данных	Детали элемента данных, если 'ymax_type_1' равен 2.
graph_items				Корневой элемент данных графиков.
graph_item				Одиночный элемент данных графика.
sortorder	целое число			Порядок отрисовки. Меньшее значение отрисовывается первым. Можно использовать для отрисовки линий или регионов позади (или спереди) других.
drawtype	целое число		0 - одинарная линия 1 - заполнение региона 2 - жирная линия 3 - точечная линия 4 - пунктирная линия	Стиль отрисовки, если 'type' графика равен 0.
color	строка			Цвет элемента (6 символов, hex).
yaxisside	целое число		0 - ось слева 1 - ось справа	Позиция оси Y (слева или справа), которой принадлежит элемент, если 'type' графика равен 0,1.
calc_fnc	целое		1 - минимум	Данные для отрисовки, если

Элемент	Свойство элемента	Тип	Диапазон	Описание
		число	2 - среднее 4 - максимум 7 - все (минимум, среднее и максимум, если 'type' графика равен 0) 9 - последнее (если 'type' графика не равен 0,1)	
	type	целое число	1 - значение элемента данных представляется пропорционально на круге 2 - значение элемента данных представляется целым кругом (сумма графика)	существует более одного значения по элементу данных.
item				Тип отрисовки для круговых/расширенных графиков.
host		строка		Одиночный элемент данных.
key		строка		Узел сети элемента данных.
Ключ элемента данных.				

Теги веб-сценария узла сети

Элемент	Свойство элемента	Тип	Диапазон	Описание
httptests				Корневой элемент веб-сценариев.
httptest				Одиночный веб-сценарий.
name		строка		Имя веб-сценария.
delay		строка		Частота выполнения веб-сценария. Секунды, единицы времени с суффиксом или пользовательский макрос.
attempts		целое число	1-10	Количество попыток выполнения шагов веб-сценария.
agent		строка		Агент клиента. Zabbix будет притворяться выбранным браузером. Эта опция полезна, когда веб-сайт возвращает разное содержимое для разных браузеров.
http_proxy		строка		Укажите используемый HTTP прокси, использую формат: <code>http://[имя_пользователя[:пароль]@]прокси.моякомпания.ком[:порт]</code>
variables		текст		Список переменных (макросов) уровня веб-сценария, которые можно затем использовать в шагах сценария.
headers		текст		HTTP заголовки, которые необходимо отправлять при выполнении запроса.
status		целое число	0 - активировано 1 - деактивировано	Состояние веб-сценария.
authentication		целое число	0 - нет 1 - простая 2 - NTLM	Метод аутентификации.
http_user		строка		Имя пользователя для аутентификации.
http_password		строка		Пароль для аутентификации.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	verify_peer	целое число	0 - нет 1 - да	Проверка SSL сертификата веб-сервера.
	verify_host	целое число	0 - нет 1 - да	Проверка, что поле Общее Имя или поле Дополнительное Имя Субъекта сертификата веб-сервера совпадают.
	ssl_cert_file	строка		Имя используемого файла SSL сертификата для аутентификации клиента.
	ssl_key_file	строка		Имя используемого файла приватного ключа SSL для аутентификации клиента.
	ssl_key_password	строка		Пароль к файлу приватного ключа SSL.
steps				Корневой элемент шагов веб-сценариев.
step				Одиночный шаг веб-сценария.
	name	строка		Имя шага веб-сценария.
	url	строка		Наблюдаемый URL.
	posts	текст		Список 'Post' переменных.
	variables	текст		Список переменных (макросов) уровня шага, которые применяются после выполнения этого шага.
	headers	текст		Если значение переменной имеет префикс 'regex:', тогда ее значение извлекается из данных, которые получены на этом шаге, в соответствии с шаблоном регулярного выражения следующим за 'regex:' префиксом HTTP заголовки, которые необходимо отправлять при выполнении запроса.
	follow_redirects	целое число	0 - нет 1 - да	Следовать HTTP перенаправлениям.
	retrieve_mode	целое число	0 - содержимое 1 - только заголовки	Режим получения HTTP ответа.
	timeout	строка		Время выполнения шага. Секунды, единицы времени с суффиксом или пользовательский макрос.
	required	строка		Требуемая строка. Игнорируется, если пустая.
	status_codes	строка		Список допустимых кодов состояний разделенных запятыми. Игнорируется, если пустой. Например: 200-201,210-299

14.4 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

Не экспортируются группы узлов сети, узлы сети, триггеры, другие карты и любые другие элементы, которые могут быть связаны с экспортируемой картой. Таким образом, если хотя бы один из элементов, на который ссылается карта, отсутствует, импорт завершится с ошибкой.

Экспорт/импорт карт сетей поддерживается начиная с Zabbix 1.8.2.

Экспорт

Для экспорта карт сети сделайте следующее:

- Перейдите в: *Мониторинг → Карты сети*
- Отметьте карты сети, которые нужно экспортировать
- Нажмите на *Экспорт* ниже списка

<input type="checkbox"/>	Name	Width	Height
<input checked="" type="checkbox"/>	Network	590	400
<input type="checkbox"/>	Offices	700	550
<input type="checkbox"/>	User map	800	600

1 selected **Export** **Delete**

Выбранные карты сети экспортируются в локальный XML файл с именем по умолчанию *zabbix_export_maps.xml*.

Импорт

Для импорта карт сети сделайте следующее:

- Перейдите в: *Мониторинг → Карты сети*
- Нажмите на *Импорт* справа
- Выберите импортируемый файл
- Отметьте необходимые опции в правилах импорта
- Нажмите на *Импорт*

* Import file zbx_export_maps.xml

Rules	Update existing	Create new	Delete missing
Groups	<input type="checkbox"/>		
Hosts	<input type="checkbox"/>	<input type="checkbox"/>	
Templates	<input type="checkbox"/>	<input type="checkbox"/>	
Template screens	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Template linkage		<input type="checkbox"/>	
Applications		<input type="checkbox"/>	<input type="checkbox"/>
Items	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Triggers	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Graphs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Web scenarios			
Screens	<input type="checkbox"/>	<input type="checkbox"/>	
Maps	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Images	<input type="checkbox"/>	<input type="checkbox"/>	
Value mappings	<input type="checkbox"/>	<input type="checkbox"/>	

Все обязательные поля ввода отмечены красной звёздочкой.

После импорта в веб-интерфейсе отобразится сообщение об успехе или об ошибке.

Правила импорта:

Правило

Обновить существующее

Существующие элементы будут обновлены данными взятыми с файла импорта. В противном случае они не будут обновляться.

Создать новое

Импорт добавит новые элементы, используя данные из файла импорта. В противном случае они не будут добавляться.

Если вы не отметите обе опции для карты сети и выберите соответствующие опции для изображений, только изображения будут импортированы. Импорт изображений доступен только пользователям Zabbix Супер Администраторам.



Если заменить существующее изображение, это повлияет на все карты, которые используют это изображение.

Описание

Формат экспорта

Экспорт маленькой карты сети с тремя элементами, их изображениями и некоторыми связями между ними. Обратите внимание, что изображения обрезаны, чтобы сэкономить место.

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>4.0</version>
    <date>2016-10-05T08:16:20Z</date>
    <images>
```

```

<image>
    <name>Server_(64)</name>
    <imagetype>1</imagetype>
    <encodedImage>iVBOR...SuQmCC</encodedImage>
</image>
<image>
    <name>Workstation_(64)</name>
    <imagetype>1</imagetype>
    <encodedImage>iVBOR...SuQmCC</encodedImage>
</image>
<image>
    <name>Zabbix_server_3D_(96)</name>
    <imagetype>1</imagetype>
    <encodedImage>iVBOR...ggg==</encodedImage>
</image>
</images>
<maps>
    <map>
        <name>Network</name>
        <width>590</width>
        <height>400</height>
        <label_type>0</label_type>
        <label_location>0</label_location>
        <highlight>1</highlight>
        <expandproblem>0</expandproblem>
        <markelements>1</markelements>
        <show_unack>0</show_unack>
        <severity_min>2</severity_min>
        <show_suppressed>0</show_suppressed>
        <grid_size>40</grid_size>
        <grid_show>1</grid_show>
        <grid_align>1</grid_align>
        <label_format>0</label_format>
        <label_type_host>2</label_type_host>
        <label_type_hostgroup>2</label_type_hostgroup>
        <label_type_trigger>2</label_type_trigger>
        <label_type_map>2</label_type_map>
        <label_type_image>2</label_type_image>
        <label_string_host/>
        <label_string_hostgroup/>
        <label_string_trigger/>
        <label_string_map/>
        <label_string_image/>
        <expand_macros>0</expand_macros>
        <background/>
        <iconmap/>
        <urls/>
        <selements>
            <selement>
                <elementtype>0</elementtype>
                <label>Host 1</label>
                <label_location>-1</label_location>
                <x>476</x>
                <y>28</y>
                <elementsubtype>0</elementsubtype>
                <areatype>0</areatype>
                <width>200</width>
                <height>200</height>
                <viewtype>0</viewtype>
                <use_iconmap>0</use_iconmap>
                <selementid>8</selementid>
                <elements>
                    <element>
                        <host>Discovered host</host>
                    </element>
                </elements>
                <icon_off>
                    <name>Server_(64)</name>
                </icon_off>
            </selement>
        </selements>
    </map>
</maps>

```

```

<icon_on/>
<icon_disabled/>
<icon_maintenance/>
<application/>
<urls/>
</selement>
<selement>
    <elementtype>0</elementtype>
    <label>Zabbix server</label>
    <label_location>-1</label_location>
    <x>252</x>
    <y>50</y>
    <elements subtype>0</elements subtype>
    <areatype>0</areatype>
    <width>200</width>
    <height>200</height>
    <viewtype>0</viewtype>
    <use_iconmap>0</use_iconmap>
    <selementid>6</selementid>
    <elements>
        <element>
            <host>Zabbix server</host>
        </element>
    </elements>
    <icon_off>
        <name>Zabbix_server_3D_(96)</name>
    </icon_off>
    <icon_on/>
    <icon_disabled/>
    <icon_maintenance/>
    <application/>
    <urls/>
</selement>
<selement>
    <elementtype>0</elementtype>
    <label>New host</label>
    <label_location>-1</label_location>
    <x>308</x>
    <y>230</y>
    <elements subtype>0</elements subtype>
    <areatype>0</areatype>
    <width>200</width>
    <height>200</height>
    <viewtype>0</viewtype>
    <use_iconmap>0</use_iconmap>
    <selementid>7</selementid>
    <elements>
        <element>
            <host>Zabbix host</host>
        </element>
    </elements>
    <icon_off>
        <name>Workstation_(64)</name>
    </icon_off>
    <icon_on/>
    <icon_disabled/>
    <icon_maintenance/>
    <application/>
    <urls/>
</selement>
</selements>
<links>
    <link>
        <drawtype>0</drawtype>
        <color>008800</color>
        <label/>
        <selementid1>6</selementid1>
        <selementid2>8</selementid2>
        <linktriggers/>

```

```

</link>
<link>
    <drawtype>2</drawtype>
    <color>00CC00</color>
    <label>100MBps</label>
    <selementid1>7</selementid1>
    <selementid2>6</selementid2>
    <linktriggers>
        <linktrigger>
            <drawtype>0</drawtype>
            <color>DD0000</color>
            <trigger>
                <description>Zabbix agent on {HOST.NAME} is unreachable
for 5 minutes</description>
                <expression>{Zabbix
host:agent.ping.nodata(5m)}=1</expression>
                <recovery_expression/>
            </trigger>
        </linktrigger>
    </linktriggers>
</link>
</links>
</map>
</maps>
</zabbix_export>

```

Теги элементов

Значения тегов элементов описаны в таблице ниже.

Элемент	Свойство элемента	Тип	Диапазон	Описание
images				Корневой элемент изображений.
image				Одиночное изображение.
	name	строка		Уникальное имя изображения.
	imagetype	целое число	1 - иконка 2 - фон	Тип изображения.
	encodedImage			Изображение закодированное в Base64.
maps				Корневой элемент карт сети.
map				Одиночная карта.
	name	строка		Уникальное имя карты.
	width	целое		Ширина карты, в пикселях.
	height	целое		Высота карты, в пикселях.
	label_type	целое число	0 - подпись 1 - IP адрес узла сети 2 - имя элемента 3 - только состояние 4 - ничего	Тип подписи к элементам карты.
	label_location	целое число	0 - снизу 1 - слева 2 - справа 3 - сверху	Расположение подписи к элементам карты по умолчанию.
	highlight	целое число	0 - нет 1 - да	Включение подсветки иконок для активных триггеров и состояний узлов сети.

Элемент	Свойство элемента	Тип	Диапазон	Описание
	expandproblem	целое число	0 - нет 1 - да	Отображение триггера с проблемой у элементах с одной проблемой.
	markelements	целое число	0 - нет 1 - да	Подсветка элементов карты, которые недавно изменили свое состояние.
	show_unack	целое число	0 - количество всех проблем 1 - количество неподтвержденных проблем 2 - количество подтвержденных и неподтвержденных проблем раздельно 0 - не классифицировано 1 - информация 2 - предупреждение 3 - средняя 4 - высокая 5 - чрезвычайная	Отображение проблем.
	severity_min	целое число	0 - нет 1 - да	Минимальная важность триггеров по умолчанию, которая отображается на карте.
	show_supresse d	целое число	0 - нет 1 - да	Отображение проблем, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.
	grid_size	целое число	20, 40, 50, 75 или 100	Размер ячейки сетки карты в пикселях, если "grid_show=1"
	grid_show	целое число	0 - да 1 - нет	Отображение сетки в настройке карты.
	grid_align	целое число	0 - да 1 - нет	Автоматическое выравнивание иконок в настройке карты.
	label_format	целое число	0 - нет 1 - да 0 - подпись 1 - IP адрес узла сети 2 - имя элемента 3 - только состояние 4 - ничего 5 - пользовательская подпись	Использование расширенной конфигурации подписей.
	label_type_host	целое число	0 - подпись 2 - имя элемента 3 - только состояние 4 - ничего 5 - пользовательская подпись	Метод отображения подписи к узлам сети, если "label_format=1"
	label_type_host group	целое число	0 - подпись 2 - имя элемента 3 - только состояние 4 - ничего 5 - пользовательская подпись	Метод отображения подписи к группам узлов сети, если "label_format=1"
	label_type_trigg er	целое число	0 - подпись 2 - имя элемента 3 - только состояние 4 - ничего 5 - пользовательская подпись	Метод отображения подписи к триггеров, если "label_format=1"

Элемент	Свойство элемента	Тип	Диапазон	Описание
			0 - подпись 2 - имя элемента 3 - только состояние 4 - ничего 5 - пользовательская подпись	
label_type_map	label_type_map	целое число	0 - подпись 2 - имя элемента 4 - ничего 5 - пользовательская подпись	Метод отображения подписи к карты сети, если “label_format=1”
			0 - подпись 2 - имя элемента 4 - ничего 5 - пользовательская подпись	
label_type_imag	label_type_imag	целое число	0 - подпись 2 - имя элемента 4 - ничего 5 - пользовательская подпись	Метод отображения подписи к изображений, если “label_format=1”
			0 - подпись 2 - имя элемента 4 - ничего 5 - пользовательская подпись	
label_string_host	label_string_host	строка		Пользовательская подпись к элементам узлов сети, если “label_type_host=5”
label_string_hostgroup	label_string_hostgroup	строка		Пользовательская подпись к элементам групп узлов сети, если “label_type_hostgroup=5”
label_string_trigger	label_string_trigger	строка		Пользовательская подпись к элементам триггеров, если “label_type_trigger=5”
label_string_map	label_string_map	строка		Пользовательская подпись к элементам карт сети, если “label_type_map=5”
label_string_image	label_string_image	строка		Пользовательская подпись к элементам изображениям, если “label_type_image=5”
expand_macros	expand_macros	целое число	0 - no 1 - yes	Раскрытие макросов в подписях при настройке карты.
background	background	id		ID изображения фона (если имеется), если “imagetype=2”
iconmap	iconmap	id		ID соответствия иконок (если имеется).
urls				
url				
name	name	строка		Одиночный URL.
url	url	строка		Имя ссылки.
elementtype	elementtype	целое число	0 - узел сети 1 - карта 2 - триггер 3 - группа узлов сети 4 - изображение	Сама ссылка URL.
elementtype	elementtype	целое число	0 - узел сети 1 - карта сети 2 - триггер 3 - группа узлов сети 4 - изображение	Принадлежность ссылки к типу элемента карты.
selements				
selement				
label	label	строка		Одиночный элемент карты.
labeltype	labeltype	целое число	0 - узел сети 1 - карта сети 2 - триггер 3 - группа узлов сети 4 - изображение	Тип элемента карты.
label	label	строка		Подпись к иконке.

Элемент	Свойство элемента	Тип	Диапазон	Описание
label_location		целое число	-1 - использование умолчаний карты 0 - снизу 1 - слева 2 - справа 3 - сверху	
x		целое число		Расположение на оси X.
y		целое число		Расположение на оси Y.
elements subtype		целое число	0 - одна группа узлов сети 1 - все узлы сети группы	Подтип элемента, если "elementtype=3"
areatype		целое число	0 - вписать во всю карту 1 - пользовательский размер	Размер области, если "elements subtype=1"
width		целое число		Ширина области, если "areatype=1"
height		целое число		Высота области, если "areatype=1"
viewtype		целое число	0 - располагать равномерно в области	Алгоритм расположения в области, если "elements subtype=1"
use_ iconmap		целое число	0 - нет 1 - да	Использование соответствия иконок для этого элемента. Уместно только, если соответствие иконок активировано на уровне карты.
selementid		id		Уникальный ID записи элемента.
elements	application	строка		
element				Один Zabbix объект, который представлен на карте (карта, группа узлов сети, узел сети и т.д.).
host				
icon_off				Используемое изображение, когда элемент в состоянии 'OK'.
icon_on				Используемое изображение, когда элемент в состоянии 'Проблема'.
icon_disable	d			Используемое изображение, когда элемент деактивирован.
icon_maintenance				Используемое изображение, когда элемент в обслуживании.
name		строка		Уникальное имя изображения.
links				

Элемент	Свойство элемента	Тип	Диапазон	Описание
link				Одиночная связь между элементами карты.
	drawtype	целое число	0 - линия 2 - жирная линия 3 - точечная линия 4 - пунктирная линия	Стиль связи.
	color	строка		Цвет связи (6 символов, hex).
	label	строка		Подпись к связи.
	selementid1	id		ID первого элемента, с которым соединена связь.
	selementid2	id		ID второго элемента, с которым соединена связь.
linktriggers				
linktrigger				Одиночный индикатор состояния связи.
	drawtype	целое число	0 - линия 2 - жирная линия 3 - точечная линия 4 - пунктирная линия	Стиль связи, когда триггер в состоянии 'Проблема'.
	color	строка		Цвет связи (6 символов, hex), когда триггер в состоянии 'Проблема'.
trigger				Используемый триггер для индикации состояния связи.
	description	строка		Имя триггера.
	expression	строка		Выражение триггера.
	recovery_expression	строка		Выражение восстановления триггера.

14.5 Комплексные экраны

Обзор

Экспорт комплексного экрана содержит структуру комплексного экрана - все настройки комплексного экрана и элементы комплексного экрана с их настройками.

Все, что добавлено на сам комплексный экран (например, узел сети, группа узлов сети или любые другие данные) не экспортируется. Таким образом, если хотя бы один из элементов, на который ссылается комплексный экран, отсутствует, импорт завершится с ошибкой.

Экспорт

Для экспорта комплексного экрана сделайте следующее:

- Перейдите в: Мониторинг → Комплексные экраны
- Отметьте комплексные экраны, которые нужно экспортировать
- Нажмите на Экспорт ниже списка

Name	Dimension (cols x rows)
<input type="checkbox"/> Servers	2 x 3
<input checked="" type="checkbox"/> Zabbix server	2 x 3
<input type="checkbox"/> Zabbix server2	3 x 3

1 selected **Export** **Delete**

Выбранные комплексные экраны экспортируются в локальный XML файл с именем по умолчанию `zabbix_export_screens.xml`.

Импорт

Для импорта комплексного экрана сделайте следующее:

- Перейдите в: *Мониторинг → Комплексные экраны*
- Нажмите на *Импорт* справа
- Выберите импортируемый файл
- Отметьте необходимые опции в правилах импорта
- Нажмите на *Импорт*

* Import file	<input type="button" value="Browse..."/>	zbx_export_screens.xml
Rules		
Groups	<input type="checkbox"/>	<input type="checkbox"/>
Hosts	<input type="checkbox"/>	<input type="checkbox"/>
Templates	<input type="checkbox"/>	<input type="checkbox"/>
Template screens	<input type="checkbox"/>	<input type="checkbox"/>
Template linkage	<input type="checkbox"/>	<input type="checkbox"/>
Applications	<input type="checkbox"/>	<input type="checkbox"/>
Items	<input type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input type="checkbox"/>	<input type="checkbox"/>
Triggers	<input type="checkbox"/>	<input type="checkbox"/>
Graphs	<input type="checkbox"/>	<input type="checkbox"/>
Web scenarios		
Screens	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Maps	<input type="checkbox"/>	<input type="checkbox"/>
Images	<input type="checkbox"/>	<input type="checkbox"/>
Value mappings	<input type="checkbox"/>	<input type="checkbox"/>
Import Cancel		

Все обязательные поля ввода отмечены красной звёздочкой.

После импорта в веб-интерфейсе отобразится сообщение об успехе или об ошибке.

Правила импорта:

Правило	Описание
Обновить существующее	Существующие элементы будут обновлены данными взятыми с файла импорта. В противном случае они не будут обновляться.
Создать новое	Импорт добавит новые элементы, используя данные из файла импорта. В противном случае они не будут добавляться.

Формат экспорта

Экспорт маленького комплексного экрана с двумя графиками, занимающими первую строку этого комплексного экрана.

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>4.0</version>
    <date>2016-10-07T08:02:40Z</date>
    <screens>
        <screen>
            <name>Zabbix server</name>
            <hsize>2</hsize>
            <vsize>3</vsize>
            <screen_items>
                <screen_item>
                    <resourcetype>0</resourcetype>
                    <width>300</width>
                    <height>80</height>
                    <x>0</x>
                    <y>0</y>
                    <colspan>1</colspan>
                    <rowspan>1</rowspan>
                    <elements>0</elements>
                    <valign>0</valign>
                    <halign>0</halign>
                    <style>0</style>
                    <url/>
                    <dynamic>1</dynamic>
                    <sort_triggers>0</sort_triggers>
                    <resource>
                        <name>CPU load</name>
                        <host>Zabbix host</host>
                    </resource>
                    <max_columns>3</max_columns>
                    <application/>
                </screen_item>
                <screen_item>
                    <resourcetype>0</resourcetype>
                    <width>300</width>
                    <height>80</height>
                    <x>1</x>
                    <y>0</y>
                    <colspan>1</colspan>
                    <rowspan>1</rowspan>
                    <elements>0</elements>
                    <valign>0</valign>
                    <halign>0</halign>
                    <style>0</style>
                    <url/>
                    <dynamic>1</dynamic>
                    <sort_triggers>0</sort_triggers>
                    <resource>
                        <name>CPU utilization</name>
                        <host>Zabbix host</host>
                    </resource>
                    <max_columns>3</max_columns>
```

```
<application/>
</screen_item>
</screen_items>
</screen>
</screens>
</zabbix_export>
```

Теги элементов

Значения тегов элементов описаны в таблице ниже.

Элемент	Свойство элемента	Тип	Диапазон	Описание
screens				
screen				
	name	строка		Уникальное имя комплексного экрана.
	hsize	целое число		Размер по горизонтали, количество колонок.
	vsizer	целое число		Размер по вертикали, количество строк.
screen_ite				
ms				
screen_ite				
m				
			0 - график 1 - простой график 2 - карта сети 3 - простая текстовая информация 4 - информация о узле сети 5 - информация о триггере 6 - информация о сервере 7 - часы 8 - комплексный экран 9 - обзор триггеров 10 - обзор данных 11 - URL 12 - история действий 13 - история событий 14 - события у группы узлов сети 15 - проблемы по важности 16 - события у узла сети 19 - прототип простого графика 20 - прототип графика	
	resourcetype	целое число		Тип ресурса.
	width	целое число		Ширина элемента комплексного экрана (в пикселях), если 'resourcetype' равен 0, 1, 7, 11, 19 или 20.
	height	целое число		Высота элемента комплексного экрана (в пикселях), если 'resourcetype' равен 0, 1, 7, 11, 19 или 20.
	x	целое число		X-координаты элемента комплексного экрана на комплексном экране, слева

Элемент	Свойство элемента	Тип	Диапазон	Описание
	y	целое число		направо. '0' означает начинать с первой колонки.
	colspan	целое число		Y-координаты элемента комплексного экрана на комплексном экране, сверху вниз. '0' означает начинать с первой строки.
	rowspan	целое число		Количество колонок, в которых будет отображаться элемент комплексного экрана.
	elements	целое число		Количество строк, в которых будет отображаться элемент комплексного экрана.
	valign	целое число	0 - по центру (по умолчанию) 1 - сверху 2 - снизу	Отображаемое количество строк в элементе комплексного экрана, если 'resourcetype' равен 3, 12, 13, 14 или 16.
	halign	целое число	0 - по центру (по умолчанию) 1 - слева 2 - справа	Выравнивание по вертикали.
		целое число	0 - простая текстовая информация 1 - HTML	Выравнивание по горизонтали.
	style	целое число	0 - локальное время 1 - время сервера 2 - время узла сети	Опция отображения элемента комплексного экрана, если 'resourcetype' равен 3.
		целое число	0 - по горизонтали 1 - по вертикали	Опция отображения элемента комплексного экрана, если 'resourcetype' равен 7.
		целое число	0 - с левой стороны 1 - сверху	Опция отображения элемента комплексного экрана, если 'resourcetype' равен 4,5.
url		строка		Опция отображения элемента комплексного экрана, если 'resourcetype' равен 9,10.
	dynamic	целое число	0 - нет 1 - да	URL ссылки, если 'resourcetype' равен 11.
	sort_triggers		0 - последнее изменение (по убыванию) 1 - важность (по убыванию) 2 - узел сети (по возрастанию)	Создание элемента комплексного экрана динамическим, если 'resourcetype' равен 0, 1, 3, 19 или 20.
		целое число		Опция сортировки триггеров, если 'resourcetype' равен 14,16.
		целое число	3 - время (по возрастанию) 4 - время (по убыванию) 5 - тип (по возрастанию) 6 - тип (по	Опция сортировки триггеров, если 'resourcetype' равен 12.

Элемент	Свойство элемента	Тип	Диапазон	Описание
			убыванию) 7 - состояние (по возрастанию) 8 - состояние (по убыванию) 9 - осталось попыток (по возрастанию) 10 - осталось попыток (по убыванию) 11 - получатель (по возрастанию) 12 - получатель (по убыванию)	
	max_columns	целое число		Как много колонок должно отображаться в ячейке комплексного экрана по сгенерированным графикам, если 'resourcetype' равен 19 или 20. Полезно, когда имеется много сгенерированных LLD графиков.
	application	строка		Фильтр по имени группы элементов данных, если 'resourcetype' равен 9 или 10.
resource				
	name	строка		Имя ресурса.
	host	строка		Имя узла сети ресурса.

15. Обнаружение

Пожалуйста, воспользуйтесь боковым меню для доступа к содержимому раздела Обнаружение.

15.1 Обнаружение сети

Обзор

Zabbix предлагает эффективное и очень гибкое автоматическое обнаружение сетевых устройств.

При наличии настроенного должным образом сетевого обнаружения вы сможете:

- ускорить развертывание Zabbix
- более простое администрирование
- использовать Zabbix в быстро меняющихся средах без излишнего администрирования

Обнаружение сетевых устройств в Zabbix базируется на основе следующей информации:

- Диапазонов IP
- Доступности внешних сервисов (FTP, SSH, WEB, POP3, IMAP, TCP и других)
- Полученной информации от Zabbix агента
- Полученной информации от SNMP агента

Обнаружение НЕ предусматривает:

- Обнаружение топологии сети

В основном обнаружение устройств сети состоит из двух этапов: обнаружение и действия.

Обнаружение

Zabbix периодически сканирует указанные в [правилах сетевого обнаружения](#) диапазоны IP. Частота сканирования настраивается индивидуально каждому правилу.

Обратите внимание, что одно правило обнаружения всегда обрабатывается одним процессом обнаружения. Диапазон IP адресов не разбивается между несколькими процессами обнаружения.

Каждое правило имеет заданный набор проверок сервисов, которые будут выполняться по диапазону IP.



Проверки обнаружения обрабатываются независимо от остальных проверок. Если какие либо проверки не находят сервис (или завершаются с ошибкой), то другие проверки все равно выполняются.

Каждая проверка сервиса и хоста (IP) выполняется модулем сетевого обнаружения, который порождает создание события обнаружения.

Событие

Результат проверки сервиса

<i>Сервис обнаружен</i>	Сервис 'доступен' после того, как был 'недоступен' или когда обнаружен первый раз.
<i>Сервис доступен</i>	Сервис 'доступен', подряд.
<i>Сервис потерян</i>	Сервис 'недоступен' после того, как был 'доступен'.
<i>Сервис недоступен</i>	Сервис 'недоступен', подряд.
<i>Узел сети обнаружен</i>	По крайней мере один сервис на узле сети 'доступен' после того, как все сервисы на этом узле сети были 'недоступны'.
<i>Узел сети доступен</i>	По крайней мере один сервис на узле сети 'доступен', подряд.
<i>Узел сети потерян</i>	Все сервисы на узле сети 'недоступны' после того, как по крайней мере один был 'доступен'.
<i>Узел сети недоступен</i>	Все сервисы на узле сети 'недоступны', подряд.

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
- Присоединение/отсоединение узлов сети к/от шаблона
- Выполнение удаленных скриптов

Эти действия можно задать с учетом 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 на страницах [операций](#) действий и [условий](#) действий.

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. Такие операции, как:

- активация узла сети
- деактивация узла сети
- добавление узла сети в группу узлов сети
- присоединение шаблона к узлу сети

При добавлении узлов сети, имя узла сети является результатом обратного преобразования DNS или IP адресом, в случае, если обратное преобразование неудачно. Преобразование выполняется на строне Zabbix сервера или Zabbix прокси, в зависимости от того, кто выполняет обнаружение. Если преобразование неудачно на прокси, оно не повторяется на стороне сервера. Если узел сети с таким же именем уже существует, следующий узел сети получит приставку к имени **_2**, затем **_3** и так далее.

Созданные узлы сети добавляются в группу *Discovered hosts* (по умолчанию, настраивается в **Администрирование → Общие → Прочие**). 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 группы узлов сети* (укажите другую группу узлов сети), так как узел сети должен принадлежать по крайней мере одной группе узлов сети.

Если узел сети уже существует с обнаруженным IP адресом, новый узел сети не создается. Однако, если действие на обнаружение содержит операции (присоединение шаблона, добавление в группу узлов сети и так далее), они будут выполнены над существующим узлом сети.

Удаление узлов сети

Начиная с Zabbix 2.4.0, узлы сети, которые созданы правилом сетевого обнаружения, удаляются автоматически, 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:

- обнаружены 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 интерфейс, который будет создан для узла сети будет интерфейсом по умолчанию. Другие IP адреса добавляются дополнительными интерфейсами.
- если узел сети отвечает только на проверки агентов, для него будет создан только агентский интерфейс. Если в дальнейшем он начнет отвечать на SNMP, дополнительно будет создан SNMP интерфейс.
- если были созданы 3 различных узла сети, будучи обнаруженными по критерию уникальности “IP”, и затем правило обнаружения изменяется, таким образом, что узлы сети А, В и С имеют одинаковый результат критерия уникальности, В и С будут созданы, как дополнительные интерфейсы к А, первому узлу сети. Узлы сети В и С останутся неизменными. В **Мониторинг → Обнаружение** добавленные интерфейсы будут отображаться в колонке “Обнаруженное устройство”, черным шрифтом и с отступом, однако колонка “Наблюдаемый узел сети” будет

отображать только А, первый созданный узел сети. “Доступен/Недоступен” не измеряется по отношению к IP адресам, которые считаются дополнительными интерфейсами.

15.1.1 Настройка правила сетевого обнаружения

Обзор

Для того чтобы настроить правило обнаружения сети, используемое в Zabbix для обнаружения узлов сети и сервисов:

- Перейдите в *Настройка → Обнаружение*
- Нажмите на *Создать правило* (или на имя существующего правила, чтобы изменить его)
- Измените атрибуты правила обнаружения

Атрибуты правил

The screenshot shows the 'Discovery rule' configuration dialog. The 'Name' field is set to 'Local network'. The 'Discovery by proxy' dropdown is set to 'No proxy'. The 'IP range' is specified as '192.168.0.1-254'. The 'Update interval' is set to '1h'. Under the 'Checks' section, several items are listed: 'HTTP' (Edit, Remove), 'HTTPS' (Edit, Remove), 'ICMP ping' (Edit, Remove), 'Zabbix agent "system.uname"' (Edit, Remove), and 'SNMPv2 agent "1.3.6.1.2.1.1.1.0"' (Edit, Remove). A 'New' button is also present. In the 'Device uniqueness criteria' section, 'IP address' is selected as the criterion. The 'Enabled' checkbox is checked. At the bottom are 'Add' and 'Cancel' buttons.

Параметр

Описание

Имя

Уникальное имя правила. Например, “Локальная сеть”.

Обнаружение
через прокси

Что осуществляет обнаружение:

без прокси - обнаружение выполняет Zabbix сервер

<имя прокси> - обнаружение осуществляется указанным прокси

Параметр	Описание
Диапазон IP адресов	<p>Диапазон IP адресов обнаружения. Может принимать следующие форматы:</p> <p>Один IP: 192.168.1.33</p> <p>Диапазон IP адресов: 192.168.1-10.1-255. Диапазон ограничен общим количеством покрываемых адресов (менее чем 64К).</p> <p>Маска IP: 192.168.4.0/24</p> <p>поддерживаемые маски IP: /16 - /30 для IPv4 адресов /112 - /128 для IPv6 адресов</p> <p>Список: 192.168.1.1-255, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24</p> <p>Начиная с Zabbix 3.0.0 это поле поддерживает пробелы, символы табуляции и многострочность.</p>
Интервал обновления	<p>Этот параметр определяет как часто Zabbix будет выполнять это правило.</p> <p>Задержка отсчитывается после того, как завершится выполнение предыдущего процесса обнаружения, таким образом перекрытия не произойдет.</p> <p>Начиная с 3.4.0, поддерживаются суффиксы времени, например, 30s, 1m, 2h, 1d.</p> <p>Пользовательские макросы поддерживаются начиная с Zabbix 3.4.0.</p> <p><i>Обратите внимание что, если используется пользовательский макрос и его значение изменилось (к примеру, 1w → 1h), следующая проверка будет выполнена в соответствии с предыдущим значением (в далеком будущем с примерами значений).</i></p> <p>Zabbix будет использовать этот список проверок для обнаружения.</p> <p>Поддерживаемые проверки: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Telnet, Zabbix агент, SNMPv1 агент, SNMPv2 агент, SNMPv3 агент, ICMP ping.</p> <p>Обнаружение, основанное на протоколах, использует функционал net.tcp.service[] для тестирования каждого узла сети, исключая SNMP, который выполняет запрос, используя SNMP OID. Zabbix агент тестируется запросом элемента данных. Пожалуйста,смотрите элементы данных агентов для получения подробностей.</p> <p>Параметр 'Порты' может принимать следующие значения:</p> <p>Один порт: 22</p> <p>Диапазон портов: 22-45</p> <p>Список: 22-45,55,60-70</p> <p>Критерием уникальности может быть:</p> <p>IP адрес - без обработки нескольких устройств с одним IP адресом. Если устройство с таким же IP существует, то оно будет считаться уже обнаруженным и новый узел сети не будет добавлен.</p> <p>Тип проверки обнаружения - либо SNMP, либо Zabbix агента проверка.</p> <p>При отмеченном правиле, оно является активным и обрабатывается Zabbix сервером.</p> <p>Если правило не отмечено, оно неактивно. Таким образом не обрабатывается.</p>
Проверки	
Критерий уникальности устройства	
Активировано	

Изменения настройки прокси

Начиная с Zabbix 2.2.0 узлы сети обнаруженные разными zabbix прокси всегда воспринимаются как разные узлы сети. До тех пор пока разрешено осуществлять поиск по совпадающим IP диапазонам в разных подсетях, изменение прокси для подсети уже находящейся под наблюдением усложняется тем, что все изменения необходимо также применить ко всем уже обнаруженным узлам сети.

Примерный план для замены прокси в правиле обнаружения:

1. деактивировать правило обнаружения
2. синхронизировать конфигурацию прокси
3. заменить Zabbix прокси в соответствующем правиле обнаружения
4. заменить zabbix прокси на всех узлах, обнаруженных соответствующим правилом

5. активировать правило обнаружения

Сценарий из реальной жизни

Допустим, мы хотим настроить обнаружение для локальной сети, имеющей диапазон IP адресов 192.168.1.1-192.168.1.254.

В нашем сценарии мы хотим получить:

- обнаружение тех узлов сети, на которых имеется Zabbix агент
- выполнение обнаружения каждые 10 минут
- добавление узла сети для наблюдения, если время работы узла сети более 1 часа
- удаление узла сети, если узел сети недоступен на протяжении более 24 часов
- добавление узлов сети Linux в группу “Linux servers”
- добавление узлов сети Windows в группу “Windows servers”
- использование *Template OS Linux* для хостов Linux
- использование *Template OS Windows* для хостов Windows

Шаг 1

Добавим правило обнаружения в сети для нашего диапазона IP адресов.

The screenshot shows the configuration of a new discovery rule. The fields are as follows:

- Name:** Local network
- Discovery by proxy:** No proxy
- IP range:** 192.168.1.1-254
- Update interval:** 10m
- Checks:** Zabbix agent "system.uname" (with [Edit](#) and [New](#) buttons)
- Device uniqueness criteria:** IP address (radio button selected)
- Enabled:** checked

Zabbix будет пытаться обнаружить узлы сети в диапазоне IP адресов 192.168.1.1-192.168.1.254, пытаясь подключиться к Zabbix агенту и получить значение ключа **system.uname**. Полученное значение от агента можно использовать для выполнения различных действий для разных операционных систем. Например, присоединение шаблона Template OS Windows к Windows серверам, шаблона Template OS Linux к Linux серверам.

Правило будет выполняться каждые 10 минут (600 секунд).

Когда правило будет добавлено, Zabbix автоматически запустит обнаружение и порождение событий, основанных на обнаружении, для дальнейшей их обработки.

Шаг 2

Определим действие для добавления обнаруженных Linux серверов в соответствующие группы/шаблоны.

The screenshot shows the 'Operations' tab of a configuration interface. A new action is being created:

- Name:** Add discovered Linux servers
- Type of calculation:** And/Or (selected), resulting in A and B and C and D
- Conditions:** A, B, C, D defined as follows:
 - A: Received value like *Linux*
 - B: Discovery status = *Up*
 - C: Service type = *Zabbix agent*
 - D: Uptime/Downtime >= 3600
- New condition:** Uptime/Downtime >= 600 (with an 'Add' button below)

Это действие активируется, если:

- сервис “Zabbix агента” в состоянии “доступен”
- значение system.uname (ключ Zabbix агента, который мы использовали при создании правила) содержит “Linux”
- Время работы более одного 1 часа (3600 секунд)

The screenshot shows the 'Operations' tab of a configuration interface. A new discovery rule is being created:

- Default subject:** Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IP/
- Default message:** Discovery rule: {DISCOVERY.RULE.NAME}
Device IP:{DISCOVERY.DEVICE.IPADDRESS}
Device DNS: {DISCOVERY.DEVICE.DNS}
Device status: {DISCOVERY.DEVICE.STATUS}
Device uptime: {DISCOVERY.DEVICE.UPTIME}
Device service name: {DISCOVERY.SERVICE.NAME}
- Operations:** Details
Add to host groups: Linux servers
Link to templates: Template OS Linux
New

Это действие будет выполнять следующие операции:

- добавление обнаруженного узла сети в группу “Linux servers” (также добавление узла сети, если он не был добавлен ранее)
- присоединение к шаблону “Template OS Linux”. Zabbix автоматически запустит наблюдение за узлом сети, используя элементы данных и триггеры из шаблона “Template OS Linux”.

Шаг 3

Зададим действие для добавления обнаруженных Windows серверов в соответствующие группы/шаблоны.

Action Operations

* Name: Add discovered Windows servers

Type of calculation: And/Or A and B and C and D

Conditions:

Label	Name
A	Received value like Windows
B	Discovery status = Up
C	Service type = Zabbix agent
D	Uptime/Downtime >= 3600

New condition: Uptime/Downtime >= 600

Add

Action Operations

Default subject: Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPA}

Default message: Discovery rule: {DISCOVERY.RULE.NAME}
Device IP: {DISCOVERY.DEVICE.IPADDRESS}
Device DNS: {DISCOVERY.DEVICE.DNS}
Device status: {DISCOVERY.DEVICE.STATUS}
Device uptime: {DISCOVERY.DEVICE.UPTIME}
Device service name: {DISCOVERY.SERVICE.NAME}

Operations:

Details
Add to host groups: Windows servers
Link to templates: Template OS Windows
New

Шаг 4

Зададим действие для удаления потерянных серверов.

Action Operations

* Name	Remove lost servers								
Type of calculation	And/Or A and B and C								
Conditions	<table border="1"> <tr> <td>Label</td> <td>Name</td> </tr> <tr> <td>A</td> <td>Uptime/Downtime >= 86400</td> </tr> <tr> <td>B</td> <td>Discovery status = Down</td> </tr> <tr> <td>C</td> <td>Service type = Zabbix agent</td> </tr> </table>	Label	Name	A	Uptime/Downtime >= 86400	B	Discovery status = Down	C	Service type = Zabbix agent
Label	Name								
A	Uptime/Downtime >= 86400								
B	Discovery status = Down								
C	Service type = Zabbix agent								
New condition	<table border="1"> <tr> <td>Service type</td> <td>=</td> <td>FTP</td> </tr> </table>	Service type	=	FTP					
Service type	=	FTP							
	Add								

Action Operations

Default subject	Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPA}						
Default message	<p>Discovery rule: {DISCOVERY.RULE.NAME}</p> <p>Device IP:{DISCOVERY.DEVICE.IPADDRESS} Device DNS: {DISCOVERY.DEVICE.DNS} Device status: {DISCOVERY.DEVICE.STATUS} Device uptime: {DISCOVERY.DEVICE.UPTIME}</p> <p>Device service name: {DISCOVERY.SERVICE.NAME}</p>						
Operations	<table border="1"> <tr> <td>Details</td> <td>Action</td> </tr> <tr> <td>Remove host</td> <td>Edit Remove</td> </tr> <tr> <td>New</td> <td></td> </tr> </table>	Details	Action	Remove host	Edit Remove	New	
Details	Action						
Remove host	Edit Remove						
New							

Сервер будет удален из конфигурации, если сервис “Zabbix агент” ‘Недоступен’ на протяжении более чем 24 часов (86400 секунд).

15.2 Авторегистрация активных агентов

Обзор

Имеется возможность разрешить автоматическую регистрацию активных Zabbix агентов, после чего сервер может начать наблюдать их. Таким образом, новые узлы сети могут быть добавлены для наблюдения без ручной настройки этих узлов сети на сервере.

Автоматическая регистрация может выполниться, когда ранее неизвестный активный агент запрашивает проверки.

Эта функция может быть очень удобной для автоматического мониторинга новых узлов из облака. Как только у вас появляется новый узел в облаке, Zabbix автоматически начинает собирать данные о производительности и доступности этого узла сети.

Авторегистрация активных агентов также поддерживает наблюдение за добавленными узлами сети с пассивными проверками. Когда активный агент запрашивает проверки, при условии, что параметры

конфигурации 'ListenIP' или 'ListenPort' указаны в файле конфигурации, агент отправляет их на сервер. (Если указано несколько IP адресов, то на сервер отправляется первый адрес.)

Сервер при добавлении нового автоматически зарегистрированного узла сети, использует полученные IP адрес и порт для настройки агента. Если значение IP адреса не получено, будет использоваться адрес с которого пришло входящее подключение. Если значение порта не получено, то используется 10050 порт.

Авторегистрация выполняется повторно:

- если меняется информация [метаданных](#) узла сети
- для вручную созданных узлов сети с отсутствующими метаданными
- если узел сети вручную изменён для мониторинга через другой Zabbix прокси
- если авторегистрация по одному хосту поступает с нового Zabbix прокси

Настройка

Задайте сервер

Убедитесь, что у вас в [файле конфигурации](#) агента указан Zabbix сервер - zabbix_agentd.conf

`ServerActive=10.0.0.1`

Если вы не указали параметр *Hostname* в zabbix_agentd.conf, то будет использоваться системное имя узла сети на котором находится агент, в качестве имени узла сети. Например, имя узла системы в Linux можно получить, выполнив команду 'hostname'.

Перезапустите агента после сделанных изменений в файле конфигурации.

Действия на авторегистрацию активных агентов

Когда сервер получает запрос на авторегистрацию от активного агента он вызывает [действие](#). Для авторегистрации необходимо настроить действие с ресурсом события “Авторегистрация”.



Настройка [сетевого обнаружения](#) не требуется для автоматической регистрации активных агентов.

Перейдите в *Настройка* → *Действия*, выберите ресурс события *Авторегистрация* и нажмите на *Создать действие*:

- В закладке Действие укажите имя действия
- Опционально задайте условия. Если вы собираетесь использовать условие “Метаданных узла сети” (Host metadata), просмотрите следующий раздел.
- В закладке Операции добавьте соответствующие операции, такие как - 'Добавить узел сети', 'Добавить в группу узлов сети' (например, *Discovered hosts*), 'Присоединить к шаблонам' и др.



Если узлы сети будут автоматически регистрироваться, они, скорее всего, будут поддерживать только активный мониторинг (такие как узлы сети, к которым доступ закрыт брандмауэром с вашего Zabbix сервера), в этом случае вы возможно захотите создать специфичный шаблон, такой как *Template_Linux-active*, и присоединить его.

Использование метаданных узлов сети

Когда агент отправляет запрос на авторегистрацию серверу он отправляет свое имя узла сети (*hostname*). В некоторых случаях (к примеру, ноды облака Amazon) имени узла сети недостаточно для Zabbix сервера для отличия обнаруженных узлов сети. Метаданные узла сети могут дополнитель но использовать для отправки дополнительной информации от агента к серверу.

Метаданные узла сети настраиваются в [файле конфигурации](#) агента - zabbix_agentd.conf. Имеется 2 способа указать метаданные узла сети в файле конфигурации:

HostMetadata
HostMetadataItem

Смотрите описание параметров по ссылке выше.



Попытка авторегистрации происходит каждый раз, как только активный агент отправляет запрос на обновление своих активных проверок серверу. Задержка между запросами определяется параметром [RefreshActiveChecks](#) агента. Первый запрос отправляется сразу же после перезапуска агента.

Пример 1

Использование метаданных узла сети для отделения узлов сети Linux от Windows.

Скажем, вы бы хотели, чтобы узлы сети автоматически регистрировались Zabbix сервером. У Вас имеются агенты Zabbix (смотрите раздел “Настройка” выше) в вашей сети. В вашей сети имеются узлы сети Windows и узлы сети Linux и у вас имеются в наличии “Template OS Linux” и “Template OS Windows” шаблоны в веб-интерфейсе Zabbix. Поэтому при регистрации узлов сети вы бы хотели, чтобы соответствующие Linux/Windows шаблоны назначались зарегистрированным узлам сети. По умолчанию, при авторегистрации серверу отправляется только имя узла сети, которого может быть не достаточно. Для того, чтобы убедиться, что будет назначен узлу сети правильный шаблон, вам следует использовать метаданные узла сети.

Настройка в веб-интерфейсе

Первым делом вам нужно произвести настройки в веб-интерфейсе. Создайте 2 действия. Первое действие:

- Имя: Авторегистрация Linux хостов
- Условия: Метаданные узлов сети содержат *Linux*
- Операции: Присоединить к шаблонам: Template OS Linux



Для этого случая вы можете пропустить операцию “Добавить узел сети”. Присоединение к шаблону требует изначально добавления узла сети, поэтому сервер сделает это автоматически.

Второе действие:

- Имя: Авторегистрация Windows хостов
- Условия: Метаданные узлов сети содержат *Windows*
- Операции: Присоединить к шаблонам: Template OS Windows

Настройка агента

Сейчас настройте агенты. Добавьте следующую строку в файлы конфигурации агентов:

HostMetadataItem=system.uname

Таким образом, вы убедитесь, что метаданные узла сети будут содержать “Linux” или “Windows” в зависимости хоста на котором запущен 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 сделанных изменений в файле конфигурации.

Пример 2

Шаг 1

Использование метаданных узла сети для получения некоторой базовой защиты от регистрации нежелательных узлов сети.

Настройка в веб-интерфейсе

Создайте действие в веб-интерфейсе, используя приведенный выше трудно-угадываемый секретный код для запрета нежелательных узлов сети:

- Имя: Действие авторегистрации Linux хостов
- 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: Template OS Linux

Пожалуйста, примите во внимание, что использование только этого метода не сможет обеспечить надежную защиту потому, что данные передаются как простой текст.

Настройка агента

Добавьте следующую строку в файл конфигурации агента:

```
HostMetadata: Linux
21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

Где “Linux” является платформой, а остальная строка является трудно-угадываемым секретным текстом.

Не забудьте перезапустить агента после внесения изменений в файл конфигурации.

Шаг 2

Имеется возможность добавления дополнительного мониторинга уже зарегистрированному узлу сети.

Настройка в веб-интерфейсе

В веб-интерфейсе обновите действие:

- Имя: Действие авторегистрации Linux хостов
- 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: Template OS Linux
 - Присоединить к шаблонам: Template DB MySQL

Настройка агента

Обновите следующую строку в файле конфигурации агента:

```
HostMetadata=MySQL on Linux  
21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

Не забудьте перезапустить агента после внесения любых изменений в файл конфигурации.

15.3 Низкоуровневое обнаружение

Обзор

Низкоуровневое обнаружение (LLD) даёт возможность автоматического создания элементов данных, триггеров и графиков для различных объектов на компьютере. Например, Zabbix может автоматически начать мониторить файловые системы или сетевые интерфейсы с вашего устройства, без необходимости создания вручную элементов данных для каждой файловой системы или сетевого интерфейса. Кроме того, в Zabbix имеется возможность настроить удаление ненужных объектов, основываясь на фактических результатах периодически выполняемого обнаружения.

Пользователь имеет возможность определить свои собственные типы обнаружения, обеспечив их функционирование согласно спецификации JSON протокола.

Общая архитектура процессов обнаружения заключается в следующем.

Сначала, пользователь создает правило обнаружения в “Настройка” → “Шаблоны” → колонка “Обнаружение”. Правило обнаружения состоит из (1) элемента данных, который осуществляет обнаружение необходимых объектов (например, файловые системы или сетевые интерфейсы) и (2) прототипов элементов данных, триггеров и графиков, которые должны быть созданы на основании полученных значений этого элемента данных.

Элемент данных, который осуществляет обнаружение необходимых объектов, подобен обычным элементам данных, которые видны в других местах: Zabbix сервер запрашивает у Zabbix агента (или любой другой указанный тип элемента данных) значение этого элемента данных, и агент отвечает текстовым значением. Разница в том, что значение, которое возвращает агент, должно содержать список обнаруженных объектов в специальном JSON формате. Хотя детали этого формата важны только для создателей собственных проверок обнаружения, всё же всем необходимо знать, что возвращаемое значение содержит список из пар: макрос → значение. Например, элемент данных “net.if.discovery” может вернуть две пары: “{#IFNAME}” → “lo” и “{#IFNAME}” → “eth0”.

Эти макросы затем используются в именах, ключах и в других полях прототипов, которые являются основой для создания реальных элементов данных, триггеров и графиков каждому обнаруженному объекту. Смотрите полный список [опций](#) по использованию макросов в низкоуровневом обнаружении.

Когда сервер получает значение элемента данных обнаружения, он смотрит на пару макрос → значение и для каждой пары создает реальные элементы данных, триггеров и графиков, основанных на их прототипах. В приведенном выше примере с “net.if.discovery”, сервер будет создавать один набор элементов данных, триггеров и графиков для локального интерфейса “lo” и другой набор для интерфейса “eth0”.

Настройка низкоуровневого обнаружения

Мы проиллюстрируем низкоуровневое обнаружение на примере обнаружения файловых систем.

Для настройки обнаружения, выполните следующее:

- Перейдите в: *Настройка* → *Шаблоны*

- Нажмите на *Обнаружение* в строке с соответствующим шаблоном

The screenshot shows the 'Templates' section of the Zabbix interface. At the top, there's a header with tabs: Applications, Items, Triggers, Graphs, Screens, and Discovery. Below the header, a table lists templates. One row is highlighted in blue, representing 'Template OS Linux'. This row also includes links for Applications (10), Items (32), Triggers (15), Graphs (5), Screens (1), and Discovery (2).

- Нажмите на *Создать правило обнаружения* в верхнем правом углу экрана
- Заполните диалог правила обнаружение необходимыми деталями

Правило обнаружения

Вкладка **Правило обнаружения** содержит общие атрибуты правила обнаружения:

The screenshot shows the 'Discovery rule' configuration dialog. It has tabs for 'Discovery rule' and 'Filters', with 'Discovery rule' selected. The form contains the following fields:

- Name:** Mounted filesystem discovery
- Type:** Zabbix agent
- Key:** vfs.fs.discovery
- Host interface:** 192.168.3.31 : 10050
- Update interval:** 1h
- Custom intervals:** A table with one row showing 'Flexible' type, '50s' interval, and '1-7,00:00-2' period.
- Keep lost resources period:** 30d
- Description:** Discovery of file systems of different types as defined in global regular expression "File systems for discovery".
- Enabled:** checked

At the bottom are 'Add' and 'Cancel' buttons.

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	Имя правила обнаружения.
Тип	Тип проверки выполняемого обнаружения; должен быть <i>Zabbix агент</i> или <i>Zabbix агент (активный)</i> при обнаружении файловых систем.
Ключ	Элемент данных "vfs.fs.discovery" уже встроен в Zabbix агент начиная с версии 2.0 для многих платформах (для получения более детальных сведений

Параметр	Описание
	смотрите список поддерживаемых ключей элементов данных), который возвращает список файловых систем, присутствующих в компьютере, и их типы в формате JSON.
	Это поле задает как часто Zabbix выполняет обнаружение. В начале, 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.
Интервал обновления	Поддерживаются суффиксы времени , например 30s, 1m, 2h, 1d, начиная с Zabbix 3.4.0.
	Пользовательские макросы поддерживаются начиная с Zabbix 3.4.0.
	<i>Обратите внимание:</i> Если укажите значение равное '0', элемент данных не будет обрабатываться. Однако, если также существует переменный интервал с ненулевым значением, элемент данных будет обрабатываться в течении действия переменного интервала.
	<i>Обратите внимание,</i> что уже созданное правило обнаружение можно выполнить незамедлительно нажатием кнопки Проверить сейчас .
Пользовательские интервалы	Вы можете создавать пользовательские правила проверки элемента данных: Гибкий - создание исключений из Интервала обновления (интервал с другой частотой обновления)
	По расписанию - создание пользовательского расписания проверки. Для получения более подробной информации смите Пользовательские интервалы . Проверка по расписанию поддерживается начиная с Zabbix 3.0.0. Это поле позволяет вам указать как много дней обнаруженный объект будет храниться (не будет удален), как только его состояние обнаружения станет "Не обнаруживается более" (мин 1 час, макс 25 лет).
Период сохранения потерянных ресурсов	Поддерживаются суффиксы времени , например 30s, 1m, 2h, 1d, начиная с Zabbix 3.4.0.
	Пользовательские макросы поддерживаются начиная с Zabbix 3.4.0.
	<i>Обратите внимание:</i> Если значение равно "0", объекты будут удалены сразу. Использование значения "0" не рекомендуется, так как простое ошибочное изменение фильтра может закончиться тем, что объект будет удален вместе со всеми данными историй.
Описание	Введите описание.
Состояние	Если отмечено, правило будет обрабатываться.

Фильтр правила обнаружения

Вкладка **Фильтры** содержит определения фильтрации правила обнаружения:

Type of calculation	Custom expression	A or (B and C) ...										
Filters	<table border="1"> <thead> <tr> <th>Label Macro</th> <th>Regular expression</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>{#FSTYPE}</td> <td>matches</td> <td>@File systems for discovery</td> </tr> <tr> <td>B</td> <td>{#MACRO}</td> <td>does not match</td> <td>regular expression</td> </tr> </tbody> </table>	Label Macro	Regular expression	A	{#FSTYPE}	matches	@File systems for discovery	B	{#MACRO}	does not match	regular expression	<input type="button" value="Add"/> <input type="button" value="Cancel"/>
Label Macro	Regular expression											
A	{#FSTYPE}	matches	@File systems for discovery									
B	{#MACRO}	does not match	regular expression									

Параметр	Описание
Тип	Доступны следующие опции расчета фильтров:

Параметр

Описание

вычисления	И - должны выполниться все фильтры;
	Или - достаточно выполнения одного фильтра;
	И/Или - используется <i>И</i> для разных имен макросов и <i>Или</i> с одинаковым именем макроса;
	Пользовательское выражение - появляется возможность указать пользовательское выражение. Формула должна включать в себя все фильтры из списка. Ограничено 255 символами.
Фильтры	Фильтр можно использовать только для генерирования реальных элементов данных, триггеров конкретных файловых систем. Ожидается использование Perl Compatible Regular Expression если вы заинтересованы только в файловых системах C:, D: и E:, вы можете поместить {#FS} "Макрос" и регулярное выражение "^C ^D ^E" в текстовые поля "Регулярное выражение". Функция возможна по типам файловых систем, при использовании макроса {#FSTYPE} (например, "%") типу диска (поддерживается только Windows агентов), используя макрос {#FSDRIVETYPE}. Вы можете ввести в поле "Регулярное выражение" регулярное выражение или ссылку на глообальную выражение . Для проверки регулярного выражения вы можете использовать "grep -E", например:
	<pre>for f in ext2 nfs reiserfs smbfs; do echo \$f grep -E '^ext ^reiserfs' echo</pre>
	Макрос {#FSDRIVETYPE} на Windows поддерживается начиная с Zabbix 3.0.0 . Определение нескольких фильтров поддерживается начиная с 2.4.0 . Обратите внимание, что если какой-то макрос из фильтра пропущен в ответе, найденный обработчик будет игнорироваться. Выпадающее меню в фильтре представлены два значения задать, которые можно использовать в регулярному выражению или наоборот, отсутствию соответствия.
	Чтобы обнаружение сработало корректно, база данных Zabbix в MySQL должна быть создана чувствительной к регистру, если имена файловых систем различаются только по регистру.
	Ошибка или опечатка в регулярном выражении, которое используется в LLD правиле, может привести к удалению тысяч элементов конфигурации, данных истории и событий на большом количестве узлов сети. Например, некорректное регулярное выражение "File systems for discovery" может привести к удалению тысяч элементов данных, триггеров, данных истории и событий.
	История правил обнаружения не сохраняется.
Кнопки диалога	
Кнопки в нижней части диалога позволяют выполнить несколько видов операций.	
Add	Добавление правила обнаружения. Эта кнопка доступна только для новых правил обнаружения.
Update	Обновление свойств правила обнаружения. Эта кнопка доступна только для уже существующих правил обнаружения.
Clone	Создание другого правила обнаружения на основе свойств текущего правила обнаружения.
Check now	Выполнение немедленного обнаружения на основе правила обнаружения. Правило обнаружения должно существовать. Смотрите более подробную информацию. <i>Обратите внимание</i> , что когда обнаружение выполняется немедленно, кэш конфигурации не обновляется, поэтому на результат не повлияют совсем недавние изменения настроек правила обнаружения.
Delete	Удаление правила обнаружения.
Cancel	Отмена изменения свойств правила обнаружения.

Прототипы элементов данных

Как только правило будет создано, перейдем к элементам данных этого правила и нажмем “Создать прототип”, чтобы создать прототип элементов данных. Обратите внимание на то, как используется макрос {#FSNAME}, где требуется указать имя файловой системы. Когда правило будет обрабатываться, этот макрос будет заменен обнаруженной файловой системой.

* Name Free disk space on {#FSNAME} (percentage)

Type Zabbix agent

* Key vfs.fs.size[{#FSNAME},pfree]

Type of information Numeric (float)

Units %

* Update interval 1m

Custom intervals	Type	Interval	Period
	Flexible	Scheduling	50s
Add			

* History storage period 1w

* Trend storage period 365d

Show value As is [show value mappings](#)

New application

Applications

- None-
- CPU
- Filesystems
- General
- Memory
- Network interfaces
- OS
- Performance
- Processes
- Security
- ...

New application prototype Application_{#FSNAME}

Application prototypes

- None-

Description

Create enabled

[Add](#)

[Cancel](#)

Можно использовать [макросы](#) низкоуровневого обнаружения и пользовательские [макросы](#) в настройках прототипа элементов данных и в [параметрах](#) предварительной обработки значений элемента данных.



Контекстное экранирование макросов низкоуровневого обнаружения для безопасного их использования в регулярных выражениях и параметрах предварительной обработки XPath.

Специфичные для прототипов элементов данных атрибуты:

Параметр

Описание

Новый прототип группы элементов данных

Вы можете задать новый прототип группы элементов данных. В свойствах группы элементов данных вы можете использовать макросы низкоуровневого обнаружения, которые, после выполнения обнаружения, будут заменены реальными значениями при создании групп элементов данных, которые специфичны для обнаруженного объекта. Смотрите также [заметки по обнаружению групп элементов данных](#) для получения более подробной информации.

Прототипы групп элементов данных

Выберите из существующих прототипов групп элементов данных.

Создать активированным

Если выбрано, элемент данных будет создан в активированном состоянии. Если не выбрано, элемент данных будет добавлен как обнаруженный объект, но в деактивированном состоянии.

Мы можем создать несколько прототипов элементов данных для каждой интересующей нас характеристики файловой системы:

Item prototypes		
All templates / Template OS Linux Discovery list / Mounted filesystem discovery Item prototypes		
<input type="checkbox"/> NAME ▲	KEY	INTERVAL
<input type="checkbox"/> Free disk space on {#FSNAME}	vfs.fs.size[{#FSNAME},free]	1m
<input type="checkbox"/> Free disk space on {#FSNAME} (percentage)	vfs.fs.size[{#FSNAME},pfree]	1m
<input type="checkbox"/> Free inodes on {#FSNAME} (percentage)	vfs.fs.inode[{#FSNAME},pfree]	1m
<input type="checkbox"/> Total disk space on {#FSNAME}	vfs.fs.size[{#FSNAME},total]	1h
<input type="checkbox"/> Used disk space on {#FSNAME}	vfs.fs.size[{#FSNAME},used]	1m

Прототипы триггеров

Мы создадим прототипы триггеров похожим способом как и прототипы элементов данных:

Trigger prototype Dependencies

* Name	Free disk space is less than 20% on volume {#FSNAME}				
Severity	<input type="radio"/> Not classified <input type="radio"/> Information <input checked="" type="radio"/> Warning <input type="radio"/> Average <input type="radio"/> High <input type="radio"/>				
* Expression	{Template OS Linux:vfs.fs.size[{#FSNAME},pfree].last(0)}<20				
Expression constructor					
OK event generation	<input type="radio"/> Expression <input type="radio"/> Recovery expression <input type="radio"/> None				
PROBLEM event generation mode	<input type="radio"/> Single <input checked="" type="radio"/> Multiple				
OK event closes	<input type="radio"/> All problems <input checked="" type="radio"/> All problems if tag values match				
Tags	<input type="text"/> tag <input type="text"/> value <input type="button" value="Add"/>				
Allow manual close	<input type="checkbox"/>				
URL	<input type="text"/>				
Description	<input type="text"/>				
Create enabled	<input checked="" type="checkbox"/>				

Специфичные для прототипов триггеров атрибуты:

Параметр

Создать активированным

Описание

Если выбрано, триггер будет создан в активированном состоянии.
Если не выбрано, триггер будет добавлен как обнаруженный объект, но в деактивированном состоянии.

Когда будут созданы реальные триггера из их прототипов, возможно потребуется большая гибкость чем использованная константа ('20' в нашем примере) для сравнения в выражении. Смотрите каким образом [пользовательские макросы с контекстом](#) могут быть полезны для получения подобной гибкости.

Также вы можете задать [зависимости](#) между прототипами триггеров (поддерживается начиная с Zabbix 3.0). Чтобы это сделать, перейдите на вкладку *Зависимости*. Прототип триггеров может зависеть от другого прототипа триггеров из этого же правила низкоуровневого обнаружения (LLD) или от обычного триггера. Прототип триггеров не может зависеть от прототипа триггеров из другого правила LLD и от триггера созданного другим прототипом триггеров. Прототип триггеров узла сети не может зависеть от триггера из шаблона.

Trigger prototypes

All templates / Template OS Linux Discovery list / Mounted filesystem discovery Item prototypes 5

SEVERITY	NAME	EXPRESSION
Warning	Free disk space is less than 20% on volume {#FSNAME}	{Template OS
Warning	Free inodes is less than 20% on volume {#FSNAME}	{Template OS

Прототипы графиков

Мы также можем создать прототипы графиков:

Graph prototype Preview

* Name Disk space usage {#FSNAME}

* Width 600

* Height 340

Graph type Pie

Show legend

3D view

* Items

Name	Type
1: Template OS Linux: Total disk space on {#FSNAME}	Graph
2: Template OS Linux: Free disk space on {#FSNAME}	Simple

Add Add prototype

Graph prototypes

All templates / Template OS Linux Discovery list / Mounted filesystem discovery Item prototypes 5

NAME	WIDTH
Disk space usage {#FSNAME}	600

В конце концов, мы создали правило обнаружения, которое выглядит как видно ниже. Оно имеет пять прототипов элементов данных, два прототипа триггеров и один прототип графиков.

Discovery rules

All templates / Template OS Linux Applications 10 Items 32 Triggers 15 Graphs 5 Screens 1

NAME	ITEMS	TRIGGERS	GRAPHS	H
Mounted filesystem discovery	Item prototypes 5 Trigger prototypes 2 Graph prototypes 1 H			

Обратите внимание: Для получения информации по настройке прототипов узлов сети, смотрите в разделе мониторинга виртуальных машин о [настройке прототипов узлов сети](#).

Обнаруженные объекты

Представленные снимки экрана ниже иллюстрируют как выглядят уже обнаруженные элементы данных, триггера и графики в настройке узла сети. Обнаруженные объекты имеют префикс ссылку золотистого цвета, которая ведет к правилу обнаружения, создавшего эти объекты.

The screenshot shows the 'Items' list in the Zabbix interface. The top navigation bar includes 'All hosts / Remote proxy: New host' (Enabled), tabs for 'ZBX', 'SNMP', 'JMX', and 'IPMI', and links for 'Applications 11' and 'Items 41'. Below the navigation is a table with columns: Wizard, Name, Triggers, and Key. Five items are listed:

Wizard	Name	Triggers	Key
<input type="checkbox"/>	... Mounted filesystem discovery: Free disk space on / (percentage)	Triggers 1	vfs.fs.size[/,free]
<input type="checkbox"/>	... Mounted filesystem discovery: Used disk space on /		vfs.fs.size[/,used]
<input type="checkbox"/>	... Mounted filesystem discovery: Free disk space on /		vfs.fs.size[/,free]
<input type="checkbox"/>	... Mounted filesystem discovery: Free inodes on / (percentage)	Triggers 1	vfs.fs.inode[/,free]

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 ключем или график с таким же именем.

Элементы данных (а также, триггеры и графики) созданные с помощью низкоуровневого правила обнаружения невозможно удалить вручную. Тем не менее, они будут удалены автоматически, если обнаруженный объект (файловая система, интерфейс и т.д.) более не обнаруживается (или более не попадает под фильтр). В этом случае они будут удалены спустя некоторое количество дней указанное в поле *Период сохранения потерянных ресурсов*.

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.

The screenshot shows the details of a deleted item. At the top, there are time filters: '1m', '7d', '1y', and 'Zabbix agent'. The status is 'Enabled' with a yellow warning icon. Below the filters, a message states: 'The item is not discovered anymore and will be deleted in 29d 23h 44m (on 2015-08-31 at 23:27)'.

Если объекты помечены на удаление, но не были удалены в назначенное время (деактивировано правило обнаружения или элемент данных узла сети), 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.

Triggers

All hosts / Remote proxy: New host Enabled ZBX SNMP JMX IPMI Applications 11 Items 41 T

<input type="checkbox"/>	Severity	Name ▲
<input type="checkbox"/>	Warning	Mounted filesystem discovery: Free disk space is less than 20% on volume /
<input type="checkbox"/>	Warning	Mounted filesystem discovery: Free inodes is less than 20% on volume /

Graphs

All hosts / Remote proxy: New host Enabled ZBX SNMP JMX IPMI Applications 11 Items 41 T

<input type="checkbox"/>	Name ▲
<input type="checkbox"/>	Template OS Linux: CPU jumps
<input type="checkbox"/>	Template OS Linux: CPU load
<input type="checkbox"/>	Template OS Linux: CPU utilization
<input type="checkbox"/>	Mounted filesystem discovery: Disk space usage /

Другие типы обнаружения

Для получения более детальных сведений и инструкций по остальным типам доступных обнаружений сморите следующие разделы:

- обнаружение [сетевых интерфейсов](#);
- обнаружение [CPU и ядер CPU](#);
- обнаружение [SNMP OID'ов](#);
- обнаружение [JMX объектов](#);
- обнаружение с использованием [ODBC SQL запросов](#);
- обнаружение [Windows служб](#);
- обнаружение [интерфейсов хостов](#) 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 LLD правил](#).

Ограничения данных для возвращаемых значений

Ограничения для JSON данных низкоуровневого правила обнаружения отсутствуют, если эти данные получены напрямую Zabbix сервером, так как полученные значения обрабатываются без сохранения в базу данных. Также ограничения отсутствуют и для пользовательских правил низкоуровневого обнаружения, однако, если предполагается получение пользовательских LLD данных при помощи пользовательского параметра, тогда накладывается ограничение по размеру значения (512 КБ) на сам пользовательский параметр.

Если данные поступают от Zabbix прокси, этот прокси вынужден сначала записать их в базу данных. В таком случае накладываются [ограничения к базе данных](#), например, 2048 байт для Zabbix прокси, который работает с IBM DB2 базой данных.

Несколько LLD правил по одному и тому же элементу данных

Начиная с Zabbix агента версии 3.2, имеется возможность задать несколько правил низкоуровневого обнаружения по одному и тому же элементу данных обнаружения.

Чтобы это сделать, вам необходимо указать [параметр](#) агента Alias, разрешив использование измененных ключей элемента данных обнаружения в разных правилах обнаружения, например `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` и так далее.

1 Создание пользовательских LLD правил

Также имеется возможность создать полностью пользовательское правило низкоуровневого обнаружения, для обнаружения любого типа объектов - к примеру, баз данных на сервере баз данных.

Чтобы это сделать, необходимо создать пользовательский элемент данных, который будет возвращать JSON, определяющий найденные объекты и дополнительно - некоторые свойства этих объектов.

Количество макросов на объект не ограничено - в то время как встроенные правила обнаружения возвращают либо один, либо два макроса (например, два в случае обнаружения файловых систем), имеется возможность возвращать больше.

Требуемый JSON формат лучше всего иллюстрируется в примере. Предположим, что мы оставим старый Zabbix агент версии 1.8 (который не поддерживает "vfs.fs.discovery"), но нам также нужно обнаруживать файловые системы. Вот простой Perl скрипт для Linux, который обнаруживает примонтированные файловые системы и выдает на выходе данные JSON, в которых включено и имя, и тип файловой системы. Одним из способов его использования является UserParameter с ключом "vfs.fs.discovery_perl":

```
#!/usr/bin/perl

$first = 1;

print "{\n";
print "\t\"data\":["\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\t]\n";
print "}\n";
```

Допустимыми символами в именах макросов низкоуровневых правил обнаружения являются **0-9**, **A-Z**, **,**, **-**, **,**.



Буквы в нижнем регистре в именах не поддерживаются.

Пример его вывода (переформатирован для наглядности) представлен ниже. JSON данные от пользовательской проверки обнаружения следуют такому же формату.

```
{
  "data": [
```

```

{
  "{#FSNAME}":"/",
  "{#FSNAME}":"/sys",
  "{#FSNAME}":"/proc",
  "{#FSNAME}":"/dev",
  "{#FSNAME}":"/dev/pts",
  "{#FSNAME}":"/lib/init/rw",
  "{#FSNAME}":"/dev/shm",
  "{#FSNAME}":"/home",
  "{#FSNAME}":"/tmp",
  "{#FSNAME}":"/usr",
  "{#FSNAME}":"/var",
  "{#FSNAME}":"/sys/fs/fuse/connections",
    "#FSTYPE":"rootfs" },
    "#FSTYPE":"sysfs" },
    "#FSTYPE":"proc" },
    "#FSTYPE":"devtmpfs" },
    "#FSTYPE":"devpts" },
    "#FSTYPE":"tmpfs" },
    "#FSTYPE":"tmpfs" },
    "#FSTYPE":"ext3" },
    "#FSTYPE":"ext3" },
    "#FSTYPE":"ext3" },
    "#FSTYPE":"ext3" },
    "#FSTYPE":"fusectl" }
}

]
}

```

Тогда, в правилах обнаружения в поле “Фильтр” мы можем указать “{#FSTYPE}”, как макрос, и “rootfs|ext3”, как регулярное выражение.



Вы не обязаны использовать имена макросов FSNAME/FSTYPE в пользовательских правилах низкоуровневого обнаружения, вы можете использовать любые другие имена, которые вам нравятся.

Обратите внимание на то, что при использовании пользовательского параметра, возвращаемые данные ограничены 512 КБ. Для получения более подробных сведенийсмотрите [ограничения данных для возвращаемых значений LLD](#).

2 Использование макросов LLD в контекстах пользовательских макросов

Пользовательские макросы [с контекстом](#) можно использовать для получения более гибких порогов в выражениях триггеров. Разные пороги можно задать на уровне пользовательского макроса и затем их можно использовать в константах триггеров, в зависимости от обнаруженного контекста.

Обнаруженный контекст появляется, когда используемые [макросы низкоуровневого обнаружения](#) в макросах раскрываются в реальные значения.

Для иллюстрации мы можем использовать данные из приведенного примера выше, предположим, что будут обнаружены следующие файловые системы: /, /home, /tmp, /usr, /var.

Мы можем задать узлу сети прототип триггера на свободное место на диске, где порог выражается при помощи пользовательского макроса с контекстом:

```
{host:vfs.fs.size[{#FSNAME}, pfree].last()}<{$LOW_SPACE_LIMIT:"{#FSNAME}"}
```

Затем добавим пользовательские макросы:

- {\$LOW_SPACE_LIMIT} 10
- {\$LOW_SPACE_LIMIT:/home} 20
- {\$LOW_SPACE_LIMIT:/tmp} 50

Тогда события сгенерируются, когда на файловых системах /, /usr и /var станет свободного места на диске меньше чем 10%, файловой системе /tmp станет свободного места на диске менее чем 50% или на файловой системе /home станет свободного места на диске менее чем 20%.

15.3.1 Обнаружение сетевых интерфейсов

Аналогично обнаружению [файловых систем](#), имеется возможность также обнаружения и сетевых интерфейсов.

Ключ элемента данных

Ключом элемента данных, который используется в [правиле обнаружения](#) является `net.if.discovery`

Этот ключ поддерживается начиная с Zabbix агента 2.0.

Поддерживаемые макросы

Вы можете использовать `{#IFNAME}` макрос в [фильтре](#) правила обнаружения и в прототипах элементов данных, триггеров и графиков.

Примеры прототипов элементов данных, которые вы возможно захотите создать на основе “`net.if.discovery`”:

- “`net.if.in[{#IFNAME},bytes]`”,
- “`net.if.out[{#IFNAME},bytes]`”.

15.3.2 Обнаружение CPU и ядер CPU

Аналогично обнаружению [файловых систем](#), имеется возможность также обнаружения CPU и ядер CPU.

Ключ элемента данных

Ключом элемента данных, который используется в [правиле обнаружения](#) является `system.cpu.discovery`

Этот ключ поддерживается начиная с Zabbix агента 2.4.

Поддерживаемые макросы

Этот ключ обнаружения возвращает два макроса - `{#CPU.NUMBER}` и `{#CPU.STATUS}`, идентифицирующие порядковый номер CPU и состояние соответственно. Отметим, нельзя сделать четкого различия между действительными, физическими процессорами, ядрами и hyperthread. `{#CPU.STATUS}` на Linux, UNIX и BSD системах возвращают состояние процессора, которое может быть как “online”, так и “offline”. На Windows системах, этот же макрос может представлять собой третье значение - “unknown” - которое указывает на то, что процессор был обнаружен, но информация по нему еще не собрана.

Обнаружение CPU основано на процессе коллектора агента, чтобы поддерживать соответствие с данными, которые поставляются коллектором и сохранить ресурсы на получение данных. Такое поведение дает эффект, что этот ключ элемента данных не работает с флагом командой строки тестирования (-t) бинарного файла, который возвращает состояние NOT_SUPPORTED и сопутствующее сообщение о том, что процесс коллектора не запущен.

Примеры прототипов элементов данных, которые вы возможно захотите создать на основе обнаружения CPU включают, например:

- “`system.cpu.util[{#CPU.NUMBER}, <тип>, <режим>]`”
- “`system.hw.cpu[{#CPU.NUMBER}, <инфо>]`”.

15.3.3 Обнаружение SNMP OID'ов

Обзор

В этом разделе мы выполним [обнаружение](#) SNMP на коммутаторе.

Ключ элемента данных

В отличие от обнаружения файловых систем и сетевых интерфейсов - этот элемент данных не требует наличия ключа “snmp.discovery”, достаточно указать, что типом элемента данных является SNMP агент.

Обнаружение SNMP OID'ов поддерживается начиная с Zabbix сервера/прокси 2.0.

Для настройки правила обнаружения, выполните следующее:

- Перейдите в: *Настройка → Шаблоны*
- Нажмите на *Обнаружение* в строке с соответствующим шаблоном

The screenshot shows the Zabbix 'Templates' page. At the top, there is a navigation bar with tabs: APPLICATIONS, ITEMS, TRIGGERS, GRAPHS, SCREENS, and DISCOVERY. Below the navigation bar, there is a table with one row. The first column contains a checkbox and the text 'Template SNMP Interfaces'. The second column contains links: Applications 1, Items 1, Triggers, Graphs, Screens, and Discovery 1. The 'Discovery' link is highlighted in blue, indicating it is selected.

- Нажмите на *Создать правило обнаружения* в верхнем правом углу экрана
- Заполните диалог правила обнаружение необходимыми деталями как показано на скриншоте ниже

The screenshot shows the 'Discovery rule' configuration dialog. The 'Discovery rule' tab is selected. The form fields are as follows:

- * Name: Network interfaces
- Type: SNMPv2 agent
- * Key: ifDescr
- * SNMP OID: discovery[{#IFDESCR},IF-MIB::ifDescr]
- * SNMP community: {\$SNMP_COMMUNITY}
- Port: (empty field)
- * Update interval: 1h
- Custom intervals:
 - Type: Flexible
 - Interval: 50s
 - Period: 1-7,00:00-24:00
 - Add: (button)
- * Keep lost resources period: 30d
- Description:

You may also consider using IF-MIB::ifType or IF-MIB::ifAlias for discovery depending on your filtering needs.
{\$SNMP_COMMUNITY} is a global macro.
- Enabled:

At the bottom of the dialog are two buttons: 'Add' and 'Cancel'.

Все обязательные поля ввода отмечены красной звёздочкой.

OID'ы для обнаружения добавляются в поле SNMP OID в следующем формате:

discovery[{#MAKPOC1}, oid1, {#MAKPOC2}, oid2, ...,]

где {#MAKPOC1}, {#MAKPOC2} ... допустимые имена низкоуровневых макросов и oid1, oid2... являются OID'ами способными сгенерировать осмысленные значения для этих макросов.

Встроенный макрос {#SNMPINDEX} содержит индекс обнаруженного OID, который применяется к обнаруженным объектам. Обнаруженные объекты группируются по значению макроса {#SNMPINDEX}.

Для понимания того, что мы имеем в виду, давайте выполним несколько раз snmpwalk на нашем коммутаторе:

```
$ 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
```

И зададим SNMP OID равным: discovery[{#IFDESCR}, ifDescr, {#IFPHYSADDRESS}, ifPhysAddress]

Теперь это правило будет обнаруживать объекты с макросом {#IFDESCR} равным **WAN**, **LAN1** и **LAN2**, макросом {#IFPHYSADDRESS} равным **8:0:27:90:7a:75**, **8:0:27:90:7a:76**, и **8:0:27:2b:af:9e**, макросом {#SNMPINDEX} равным индексам обнаруженных OID 1, 2 и 3:

```
{
  "data": [
    {
      "{#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 обнаруженный объект не имеет указанный OID, тогда по этому объекту соответствующий макрос пропускается. 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 случае SNMP обнаружения discovery[{#IFDESCR}, ifDescr, {#IFALIAS}, ifAlias] вернется следующая структура:

```
{  
    "data": [  
        {  
            "#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 скриншот иллюстрирует каким образом мы можем использовать эти макросы в прототипах элементов данных:

Item prototype Preprocessing

* Name	Incoming traffic on interface {#IFDESCR}		
Type	SNMPv2 agent		
* Key	ifInOctets[{#IFDESCR}]		
* SNMP OID	IF-MIB::ifInOctets.{#IFDESCR}		
* SNMP community	{\$SNMP_COMMUNITY}		
Port			
Type of information	Numeric (unsigned)		
Units	bps		
* Update interval	1m		
Custom intervals	Type	Interval	Period
	Flexible	Scheduling	50s
			1-7,00:00-
	Add		
* History storage period	1w		
* Trend storage period	365d		
Show value	As is		show value mappings
New application			

Опять же, вы можете создать столько прототипов элементов данных, сколько необходимо:

Item prototypes

All templates / Template SNMP Interfaces Discovery list / Network interfaces Item prototypes 8

<input type="checkbox"/> NAME ▲	KEY	INTERVAL	HI
<input type="checkbox"/> Admin status of interface {#IFDESCR}	ifAdminStatus[{#IFDESCR}]	1m	7d
<input type="checkbox"/> Alias of interface {#IFDESCR}	ifAlias[{#IFDESCR}]	1h	7d
<input type="checkbox"/> Description of interface {#IFDESCR}	ifDescr[{#IFDESCR}]	1h	7d
<input type="checkbox"/> Inbound errors on interface {#IFDESCR}	ifInErrors[{#IFDESCR}]	1m	7d
<input type="checkbox"/> Incoming traffic on interface {#IFDESCR}	ifInOctets[{#IFDESCR}]	1m	7d
<input type="checkbox"/> Operational status of interface {#IFDESCR}	ifOperStatus[{#IFDESCR}]	1m	7d
<input type="checkbox"/> Outbound errors on interface {#IFDESCR}	ifOutErrors[{#IFDESCR}]	1m	7d
<input type="checkbox"/> Outgoing traffic on interface {#IFDESCR}	ifOutOctets[{#IFDESCR}]	1m	7d

Прототипы триггеров

Следующий скриншот иллюстрирует каким образом мы можем использовать эти макросы в прототипах триггеров:

Trigger prototype Dependencies

* Name Operational status was changed on {HOST.NAME} int

Severity Not classified Information Warning Average High D

* Expression {Template SNMP Interfaces:ifOperStatus[#{IFDESCR}].diff(0)}=1 Add

Expression constructor

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Tags tag value Remove Add

Allow manual close

URL

Description

Create enabled

Trigger prototypes

All templates / Template SNMP Interfaces Discovery list / Network interfaces Item prototypes 8

<input type="checkbox"/> SEVERITY	NAME ▲	EXPR
<input type="checkbox"/> Information	Operational status was changed on {HOST.NAME} interface {#IFDESCR}	{Temp}

Прототипы графиков

Следующий скриншот иллюстрирует каким образом мы можем использовать эти макросы в прототипах графиков:

Graph prototype Preview

* Name	Traffic on interface {#IFDESCR}									
* Width	900									
* Height	200									
Graph type	Normal ▾									
Show legend	<input checked="" type="checkbox"/>									
Show working time	<input checked="" type="checkbox"/>									
Show triggers	<input checked="" type="checkbox"/>									
Percentile line (left)	<input type="checkbox"/>									
Percentile line (right)	<input type="checkbox"/>									
Y axis MIN value	Calculated ▾									
Y axis MAX value	Calculated ▾									
* Items	<table border="1"> <thead> <tr> <th>Name</th> <th>Function</th> <th>Draw st</th> </tr> </thead> <tbody> <tr> <td>1: Template SNMP Interfaces: Incoming traffic on interface {#IFDESCR}</td> <td>avg ▾</td> <td>Gradie</td> </tr> <tr> <td>2: Template SNMP Interfaces: Outgoing traffic on interface {#IFDESCR}</td> <td>avg ▾</td> <td>Gradie</td> </tr> </tbody> </table>	Name	Function	Draw st	1: Template SNMP Interfaces: Incoming traffic on interface {#IFDESCR}	avg ▾	Gradie	2: Template SNMP Interfaces: Outgoing traffic on interface {#IFDESCR}	avg ▾	Gradie
Name	Function	Draw st								
1: Template SNMP Interfaces: Incoming traffic on interface {#IFDESCR}	avg ▾	Gradie								
2: Template SNMP Interfaces: Outgoing traffic on interface {#IFDESCR}	avg ▾	Gradie								
Add	Add prototype									

Graph prototypes

All templates / Template SNMP Interfaces	Discovery list / Network interfaces	Item prototypes 8	Templates 1
<input type="checkbox"/> NAME ▾		WIDTH	
<input type="checkbox"/> Traffic on interface {#SNMPVALUE}		900	

Результат нашего правила обнаружения:

Discovery rules				
All templates / Template SNMP Interfaces		Applications 1	Items 1	Triggers
<input type="checkbox"/> NAME ▾		ITEMS	TRIGGERS	GRAPHS
<input type="checkbox"/> Network interfaces	Item prototypes 8	Trigger prototypes 1	Graph prototypes 1	Host prototypes 1

Обнаруженные объекты

Когда сервер выполнит правило обнаружения, будут созданы реальные элементы данных, триггеры и графики на основе значений, которые вернет правило обнаружения SNMP. В настройках узла эти

объекты будут иметь префикс с оранжевой ссылкой на правило обнаружения, с которого пришли эти объекты.

Items

All hosts / Switch1 Enabled ZBX SNMP JMX IPMI Applications 1 Items 241 Triggers 30 Groups 1 Filter ▾

<input type="checkbox"/>	Wizard Name	Triggers Key ▲
<input type="checkbox"/>	Network interfaces: Admin status of interface 1	ifAdminStatus[1]
<input type="checkbox"/>	Network interfaces: Admin status of interface 2	ifAdminStatus[2]
<input type="checkbox"/>	Network interfaces: Admin status of interface 3	ifAdminStatus[3]
<input type="checkbox"/>	Network interfaces: Admin status of interface 4	ifAdminStatus[4]

Triggers

All hosts / Switch1 Enabled ZBX SNMP JMX IPMI Applications 1 Items 241 Triggers 30 Groups 1 Filter ▾

<input type="checkbox"/>	Severity Name ▲	Exp
<input type="checkbox"/>	Information Network interfaces: Operational status was changed on {HOST.NAME} interface 1	{pro}
<input type="checkbox"/>	Information Network interfaces: Operational status was changed on {HOST.NAME} interface 2	{pro}
<input type="checkbox"/>	Information Network interfaces: Operational status was changed on {HOST.NAME} interface 3	{pro}
<input type="checkbox"/>	Information Network interfaces: Operational status was changed on {HOST.NAME} interface 4	{pro}

Graphs

All hosts / Switch1 Enabled ZBX SNMP JMX IPMI Applications 1 Items 241 Triggers 30 Groups 1 Filter ▾

<input type="checkbox"/>	Name ▲
<input type="checkbox"/>	Network interfaces: Traffic on interface 1
<input type="checkbox"/>	Network interfaces: Traffic on interface 2
<input type="checkbox"/>	Network interfaces: Traffic on interface 3
<input type="checkbox"/>	Network interfaces: Traffic on interface 4

15.3.4 Обнаружение JMX объектов

Обзор

Имеется возможность обнаружения всех атрибутов JMX MBeans или MBean, а также можно указать шаблон для обнаружения этих объектов.

Для настройки правила обнаружения обязательно понимать разницу между Mbean и атрибутами Mbean. MBean является объектом, который может представлять собой устройство, приложение или любой другой ресурс, которым необходимо управлять. Например, имеется Mbean, который представляет собой веб-сервер. Его атрибутами являются количество подключений, количество потоков, время ожидания запросов, кэш файлов http, использование памяти и так далее. Выражая эту мысль человеческим языком, мы можем определить кофе машину как Mbean, у которого имеются следующие атрибуты для мониторинга: количество воды на каждую кружку, средний расход воды за определенный период времени, количество кофейных зерен требующихся на каждую кружку, кофейные зерна и время наполнения водой и так далее.

Ключ элемента данных

В настройках [правила обнаружения](#) выберите **JMX агент** в поле *Tip*.

Используемым ключом элемента данных, является

`jmx.discovery[<режим обнаружения>, <имя объекта>]`

где

- режим обнаружения - один из следующих: *attributes* (получение JMX MBean атрибутов, по умолчанию) или *beans* (получение JMX MBeans)
- имя объекта - шаблон имени объекта, который определяет получаемые имена MBean (по умолчанию пусто, получение всех зарегистрированных компонентов).

Вы можете обратиться к ObjectName [документации](#) для получения опций по указанию шаблона имени объекта.

Если параметры не переданы, с JMX запрашиваются все MBean атрибуты.



Отсутствие заданных параметров в случае с JMX обнаружением или попытка получения всех атрибутов широкого диапазона, таких как `*:type=*, name=*`, может привести к потенциальным проблемам производительности.

Этот ключ поддерживается начиная с Zabbix Java gateway 3.4.

Примеры ключей элементов данных:

```
jmx.discovery #Получение всех JMX MBean атрибутов
jmx.discovery[beans] #Получение всех JMX MBeans
jmx.discovery[attributes,"*:type=GarbageCollector,name=*"] #Получение всех атрибутов
сборщика мусора
jmx.discovery[beans,"*:type=GarbageCollector,name=*"] #Получение всех сборщиков мусора
```

Этот элемент данных возвращает JSON объект. Например, в обнаружении MBean атрибутов (переформатировано для наглядности):

```
{
  "data": [
    {
      "{#JMXTYPE}": "java.lang.Long",
      "{#JMXTYPE}": "java.lang:type=GarbageCollector,name=PS Scavenge",
      "{#JMDESC}": "java.lang:type=GarbageCollector,name=PS
Scavenge,CollectionCount",
      "{#JMATTR}": "CollectionCount"
    },
    {
      "{#JMXTYPE}": "java.lang.Long",
      "{#JMXTYPE}": "java.lang:type=GarbageCollector,name=PS Scavenge",
      "{#JMDESC}": "java.lang:type=GarbageCollector,name=PS
Scavenge,CollectionCount"
    }
  ]
}
```

```

        "{#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"
    }
}
]
}

```

В обнаружении MBeans (переформатировано для наглядности):

```
{
  "data": [
    {
      "{#JMXDOMAIN}":"java.lang",
      "{#JMXTYPE}":"GarbageCollector",
      "{#JMXOBJ}":"java.lang:type=GarbageCollector,name=PS Scavenge",
      "{#JMXNAME}":"PS Scavenge"
    }
  ]
}
```

Поддерживаемые макросы

Следующие макросы поддерживаются для использования в [фильтре](#) правила обнаружения и прототипах элементов данных, триггеров и графиков:

Макрос	Описание
Обнаружение MBean атрибутов	

{#JMXVALUE}	Значение атрибута.
{#JMXTYPE}	Тип атрибута.
{#JMXOBJ}	Имя объекта.
{#JMXDESC}	Имя объекта, включая имя атрибута.
{#JMXATTR}	Имя атрибуты.

Обнаружение MBeans

{#JMXDOMAIN}	Домен MBean. (<i>зарезервированное имя в Zabbix</i>)
{#JMXOBJ}	Имя объекта. (<i>зарезервированное имя в Zabbix</i>)
{#JMX<ключ свойства>}	MBean свойства (такие как {#JMXTYPE}, {#JMXNAME}). Необходимо сделать важные замечания на которые следует обратить внимания при определении имени MBean атрибута, которое создано из имени MBean свойства по следующему алгоритму:

Макрос

Описание

- * регистр имени атрибута сменился на верхний регистр;
- * регистр имени атрибута игнорируется (макрос LLD не генерируется) в случае, если он содержит неподдерживаемые символы. Поддерживаемые символы можно описать при помощи следующего регулярного выражения: "A-Z0-9_.";
- * если имя атрибута называется "obj" или "domain", оно будет заменено значениями свойств Zabbix {#JMXOBJ} и {#JMXDOMAIN} (поддерживается начиная с Zabbix 3.4.3.).

Пожалуйста, обратите внимание на этот пример с jmx.discovery (с "beans" режимом). У MBean имеются следующие свойства:

```
name=test
тип=Type
attributes []=1, 2, 3
Name=NameOfTheTest
domAin=some
```

В результате JMX обнаружения сгенерируются следующие LLD макросы:

- {#JMXDOMAIN} - Zabbix внутренний, описывающий домен MBean
- {#JMXOBJ} - Zabbix внутренний, описывающий объект MBean
- {#JMXNAME} - создается из свойства "name"

Игнорируемые свойства:

- тип : это имя содержит неподдерживаемые символы (не-ASCII)
- attributes[] : это имя содержит неподдерживаемые символы (квадратные скобки не поддерживаются)
- Name : уже задан (name=test)
- domAin : является зарезервированным именем в Zabbix

Давайте рассмотрим еще два практических примера создания LLD правила с использованием Mbean. Для понимания разницы между LLD правилом, которое собирает Mbeans, и LLD правилом, которое собирает Mbean атрибуты, лучше взгляните на следующую таблицу:

MBean1	MBean2	MBean3
MBean1Attribute1	MBean2Attribute1	MBean3Attribute1
MBean1Attribute2	MBean2Attribute2	MBean3Attribute2
MBean1Attribute3	MBean2Attribute3	MBean3Attribute3

LLD правило, собирающее Mbeans

Это правило вернет 3 объекта: колонки верхней строки: MBean1, MBean2, MBean3.

Для получения более подробной информации об объектах, пожалуйста, обратитесь к таблице [поддерживаемых макросов](#), раздел *Обнаружение MBeans*.

Настройки правила обнаружения, собирающего Mbeans (без атрибутов), выглядят следующим образом:

Discovery rules

The screenshot shows the 'Discovery rules' configuration page. A new discovery rule is being created with the following settings:

- Name:** JMX garbage collectors
- Type:** JMX agent
- Key:** jmx.discovery[beans,"*:type=GarbageCollector,name=*"]
- Host interface:** 127.0.0.1 : 12340
- JMX endpoint:** service:jmx:rmi://jndi/rmi://[HOST.CONN]:[HOST.PORT]/jmxrmi

Используемый ключ:

```
jmx.discovery[beans, "":type=GarbageCollector, name=""]
```

Обнаружатся все сборщики мусора без атрибутов. Так как сборщики мусора имеют одинаковый набор атрибутов, мы можем использовать желаемые атрибуты в прототипах элементов данных следующим образом:

The screenshot shows the 'Item prototypes' configuration page. Three item prototypes are defined:

Name	Key
GC {#JMXNAME} CollectionCount	jmx[{#JMXOBJ},CollectionCount]
GC {#JMXNAME} CollectionTime	jmx[{#JMXOBJ},CollectionTime]
GC {#JMXNAME} Valid	jmx[{#JMXOBJ},Valid]

Используемые ключи:

```
jmx[{#JMXOBJ},CollectionCount]
jmx[{#JMXOBJ},CollectionTime]
jmx[{#JMXOBJ},Valid]
```

Результатом LLD правила обнаружения будет что-то близкое к этому (элементы данных обнаружились по двум сборщикам мусора):

The screenshot shows the 'Discovery list' page. It lists several triggers corresponding to the discovered Garbage Collector Mbeans:

Name	Triggers	Key
JMX garbage collectors: GC PS MarkSweep CollectionCount		jmx["java.lang:type=GarbageCollector,name=PS MarkSweep",CollectionCount]
JMX garbage collectors: GC PS MarkSweep CollectionTime		jmx["java.lang:type=GarbageCollector,name=PS MarkSweep",CollectionTime]
*** JMX garbage collectors: GC PS MarkSweep Valid		jmx["java.lang:type=GarbageCollector,name=PS MarkSweep",Valid]
JMX garbage collectors: GC PS Scavenge CollectionCount		jmx["java.lang:type=GarbageCollector,name=PS Scavenge",CollectionCount]
JMX garbage collectors: GC PS Scavenge CollectionTime		jmx["java.lang:type=GarbageCollector,name=PS Scavenge",CollectionTime]
*** JMX garbage collectors: GC PS Scavenge Valid		jmx["java.lang:type=GarbageCollector,name=PS Scavenge",Valid]

LLD правило, собирающее атрибуты Mbean

Это правило вернет 9 объектов со следующими полями: MBean1Attribute1, MBean2Attribute1, Mbean3Attribute1, MBean1Attribute2, MBean2Attribute2, Mbean3Attribute2, MBean1Attribute3, MBean2Attribute3, Mbean3Attribute3.

Для получения более подробной информации об объектах, пожалуйста, обратитесь к таблице [поддерживаемых макросов](#), раздел *Обнаружение MBean атрибутов*.

Настройки правила обнаружения, собирающего Mbean атрибуты, выглядят следующим образом:

Discovery rules

All hosts / JMX Enabled ZBX SNMP JMX IPMI Discovery list / JMX garbage collectors Item prototypes 1 Trigger prototypes

Discovery rule Filters

Name: JMX garbage collectors
Type: JMX agent
Key: jmx.discovery[attributes, "":type=GarbageCollector,name=*"]
Host interface: 127.0.0.1 : 12340
JMX endpoint: service:jmx:rmi://jndi/rmi://[HOST.CONN]:[HOST.PORT]/jmxrmi

Используемый ключ:

```
jmx.discovery[attributes, "":type=GarbageCollector,name=*"]
```

Обнаружатся все сборщики мусора с одним элементом атрибута.

Item prototypes

All hosts / JMX Enabled ZBX SNMP JMX IPMI Discovery list / JMX garbage collectors Item prototypes 1

Name	Key
{#JMXOBJ} {#JMXATTR}	jmx[#{JMXOBJ},#{JMXATTR}]

В этом конкретном случае элемент данных создается из прототипа по каждому MBean атрибуту. Основным недостатком подобной конфигурации является то, что невозможно создать триггер из прототипов триггеров, так как имеется только один прототип элементов данных по всем атрибутам. Таким образом, подобную установку можно использовать для сбора данных, но не рекомендуется использовать для автоматического мониторинга.

15.3.5 Обнаружение с использованием ODBC SQL запросов

Обзор

Этот тип низкоуровневого [обнаружения](#) осуществляется с использованием SQL запросов, полученные результаты которых автоматически преобразуются в объект JSON, пригодный для низкоуровневого обнаружения.

Ключ элемента данных

SQL запросы выполняются при помощи элементов данных типа “Монитор баз данных”. Так что, большая часть указаний со страницы [ODBC мониторинга](#) применима к получению работающего “Монитора баз данных” правила обнаружения, единственная разница лишь в том, что необходимо использовать ключ

```
db.odbc.discovery[<описание>, <dsn>]
```

вместо “db.odbc.select[<описание>, <dsn>]”.

Обнаружение с использованием SQL запросов поддерживается начиная с Zabbix сервера/прокси 3.0.

В качестве практического примера, иллюстрирующего как SQL запрос трансформируется в JSON, рассмотрим низкоуровневое обнаружения Zabbix прокси, выполнив ODBC запрос в Zabbix базу данных. Он может быть полезен для автоматического создания [внутренних элементов данных](#) “zabbix[proxy,<имя>,lastaccess]”, чтобы наблюдать какие прокси живы.

Давайте начнем с настройки правила обнаружения:

Discovery rule Filters

* Name Proxy discovery

Type Database monitor

* Key db.odbc.discovery[proxies,{\\$DSN}]

User name

Password

* SQL query

```
SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid WHERE h1.status IN (5, 6) GROUP BY h1.host;
```

* Update interval 1h

Custom intervals

Type	Interval	Period
Flexible	Scheduling	50s 1-7,00:00-24:00

Add

* Keep lost resources period 30d

Description

Enabled

Add Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Здесь используется следующий прямой запрос в базу данных Zabbix для выборки всех Zabbix прокси вместе с количеством узлов сети, за которыми эти прокси наблюдают. Количество узлов сети можно использовать, например, для фильтрации пустых прокси:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid WHERE h1.status IN (5, 6) GROUP BY h1.host;
+-----+-----+
| host | count |
+-----+-----+
| Japan 1 |      5 |
| Japan 2 |     12 |
| Latvia |      3 |
+-----+-----+
```

```
3 rows in set (0.01 sec)
```

Благодаря внутреннему механизму обработки элемента данных “db.odbc.discovery[]”, результат этого запроса автоматически преобразуется в следующий JSON:

```
{  
    "data": [  
        {  
            "{#HOST}": "Japan 1",  
            "{#COUNT}": "5"  
        },  
        {  
            "{#HOST}": "Japan 2",  
            "{#COUNT}": "12"  
        },  
        {  
            "{#HOST}": "Latvia",  
            "{#COUNT}": "3"  
        }  
    ]  
}
```

Видно, что имена колонок становятся именами макросов и выбранные строки становятся значениями этих макросов.

Если результат преобразования имени колонки в имя макроса неочевиден, предлагается использовать алиасы к именам колонок, так же как “COUNT(h2.host) AS count” в примере выше.

В случае, если имя колонки не удается сконвертировать в допустимое имя макроса, правило обнаружения становится неподдерживаемым с детальным сообщением об ошибке какой номер колонки не удалось преобразовать. Если желательна дополнительная помощь, полученные имена колонки отражаются при DebugLevel=4 в файле журнала Zabbix сервера:

\$ 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 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid  
WHERE h1.status IN (5, 6) GROUP BY h1.host;'
```

```
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 column #2 name to macro.
```

Теперь, когда мы понимаем как SQL запрос трансформируется в JSON объект, мы можем использовать макрос {#HOST} в прототипах элементов данных:

Item prototype **Preprocessing**

* Name	Last acces time of proxy {#HOST}						
Type	Zabbix internal						
* Key	zabbix[proxy,{#HOST}.lastaccess]						
Type of information	Numeric (unsigned)						
Units	unixtime						
* Update interval	60s						
Custom intervals	<table border="1"> <thead> <tr> <th>Type</th> <th>Interval</th> <th>Period</th> </tr> </thead> <tbody> <tr> <td>Flexible</td> <td>50s</td> <td>1-7:00:00-24:00</td> </tr> </tbody> </table> Add	Type	Interval	Period	Flexible	50s	1-7:00:00-24:00
Type	Interval	Period					
Flexible	50s	1-7:00:00-24:00					
* History storage period	90d						
* Trend storage period	365d						
Show value	<input type="button" value="As is"/> <input type="button" value="show value mappings"/>						

Как только обнаружение будет выполнено, элемент данных будет создан по каждому прокси:

<input type="checkbox"/>	Name	Triggers	Key ▲
<input type="checkbox"/>	Proxy discovery: Last access time of proxy Japan1		zabbix[proxy,Japan1,lastaccess]
<input type="checkbox"/>	Proxy discovery: Last access time of proxy Japan2		zabbix[proxy,Japan2,lastaccess]
<input type="checkbox"/>	Proxy discovery: Last access time of proxy Latvia		zabbix[proxy,Latvia,lastaccess]

15.3.6 Обнаружение служб Windows

Overview

Аналогично обнаружению [файловых систем](#), имеется возможность также обнаружения и служб Windows.

Ключ элемента данных

Ключом элемента данных, который используется в [правиле обнаружения](#) является `service.discovery`

Этот ключ поддерживается начиная с Zabbix Windows агента 3.0.

Поддерживаемые макросы

Следующие макросы поддерживаются для использования в [фильтре](#) правила обнаружения и прототипах элементов данных, триггеров и графиков:

Макрос	Описание
{#SERVICE.NAME}	Имя службы.

Макрос	Описание
{#SERVICE.DISPLAYNAME}	Отображаемое имя службы.
{#SERVICE.DESCRIPTION}	Описание службы.
{#SERVICE.STATE}	Числовое значение состояния службы: 0 - Запущена 1 - Пауза 2 - Ожидание старта 3 - Ожидание паузы 4 - Ожидание продолжения 5 - Ожидание остановки 6 - Остановлена 7 - Неизвестно
{#SERVICE.STATENAME}	Имя состояния службы (<i>Запущена, Пауза, Ожидание старта, Ожидание паузы, Ожидание продолжения, Ожидание остановки, Остановлена или Неизвестно</i>).
{#SERVICE.PATH}	Путь к службе.
{#SERVICE.USER}	Пользователь службы.
{#SERVICE.STARTUP}	Числовое значение типа запуска службы: 0 - Автоматически 1 - Автоматически (отложенный запуск) 2 - Вручную 3 - Отключена 4 - Неизвестно
{#SERVICE.STARTUPNAME}	Имя типа запуска службы (<i>Автоматически, Автоматически (отложенный запуск), Вручную, Отключена, Неизвестно</i>).
{#SERVICE.STARTUPTRIGGER}	Числовое значение, указывающее, тип запуска службы: 0 - не запускается по триггерам 1 - запускается по триггерам
}	Этот макрос поддерживается начиная с Zabbix 3.4.4. Он полезен для обнаружения таких типов запуска служб как <i>Автоматический запуск (по триггеру)</i> , <i>Автоматический отложенный запуск (по триггеру)</i> и <i>Ручной запуск (по триггеру)</i> .

На основе обнаружения служб Windows вы можете создать прототип [элементов данных](#), к примеру:

```
service.info[{#SERVICE.NAME},<параметр>]
```

где *параметр* принимает следующие значения: *state, displayname, path, user, startup* или *description*.

Например, чтобы получить отображаемое имя службы вам необходимо использовать элемент данных “`service.info[{#SERVICE.NAME},displayname]`”. Если значение *параметр* не указано (“`service.info[{#SERVICE.NAME}]`”), будет использоваться параметр *state* по умолчанию.

15.3.7 Обнаружение интерфейсов хостов в Zabbix

Обзор

Имеется возможность [обнаружения](#) всех интерфейсов добавленных к узлу сети в веб-интерфейсе Zabbix.

Ключ элемента данных

Элемент данных, который используется в [правиле обнаружения](#) является `zabbix[host,discovery,interfaces]`

внутренний элемент данных. Этот элемент данных поддерживается начиная с Zabbix сервера 3.4.

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:

```
{"data": [{"#{IF.CONN}": "192.168.3.1", "#{IF.IP}": "192.168.3.1", "#{IF.DNS}": "", "#{IF.PORT}": "10050", "#{IF.TYPE}": "AGENT", "#{IF.DEFAULT}": 1}]} 
```

При наличии нескольких интерфейсов, их записи сортируются в JSON в порядке:

- Тип интерфейса,
- Умолчание - интерфейс по умолчанию помещается до интерфейсов не по умолчанию,
- 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: 0 - не является интерфейсом по умолчанию 1 - интерфейс по умолчанию
{#IF.SNMP.BULK}	Состояние массовой обработки SNMP у интерфейса: 0 - деактивировано 1 - активировано Этот макрос возвращается только, если типом интерфейса является “SNMP”.

Заметки по низкоуровневому обнаружению

Обнаружение групп элементов данных

Прототипы групп элементов данных поддерживают LLD макросы.

Один прототип групп элементов данных может использоваться несколькими прототипами элементов данных одного правила обнаружения.

Как и другие, обнаруженные объекты групп элементов данных следуют времени жизни, указанному в правиле обнаружения (настройка 'период сохранения потерянных ресурсов') - они будут удалены после того как не обнаруживаются указанное количество дней.

Если группа элементов данных более не обнаруживается, все обнаруженные элементы данных автоматически из неё удаляются, даже если сама группа элементов данных еще не удалена из-за настройки 'периода сохранения потерянных ресурсов'.

Прототипы групп элементов данных, указанные в одном правиле обнаружения не могут обнаружить такую же группу элементов данных. В этой ситуации будет успешным первое обнаружение

прототипа, остальные сообщат соответствующую ошибку LLD. Только прототипы групп элементов данных, указанных в разных правилах обнаружения, могут вызвать обнаружение такой же группы элементов данных.

16. Распределенный мониторинг

Обзор

Zabbix обеспечивает эффективные и надежные пути мониторинга распределенной ИТ инфраструктуры, используя [прокси](#).

Прокси можно использовать для сбора данных в локальном масштабе от имени централизованного Zabbix сервера, а затем отправлять данные на сервер.

Возможности прокси

Делая выбор между использованием прокси или нет, во внимание должны быть приняты несколько соображений.

	Прокси
<i>Легкость</i>	Да
<i>Веб-интерфейс</i>	Нет
<i>Независимость работы</i>	Да
<i>Легкость обслуживания</i>	Да
<i>Автоматическое создание БД</i> ¹	Да
<i>Локальное администрирование</i>	Нет
<i>Готовность к встроенному аппаратному обеспечению</i>	Да
<i>Однонаправленные TCP соединения</i>	Да
<i>Централизованная конфигурация</i>	Да
<i>Генерирование оповещений</i>	Нет



[1] Функция автоматического создания БД работает только с SQLite. Остальные базы данных требуют [ручного создания](#).

16.1 Прокси

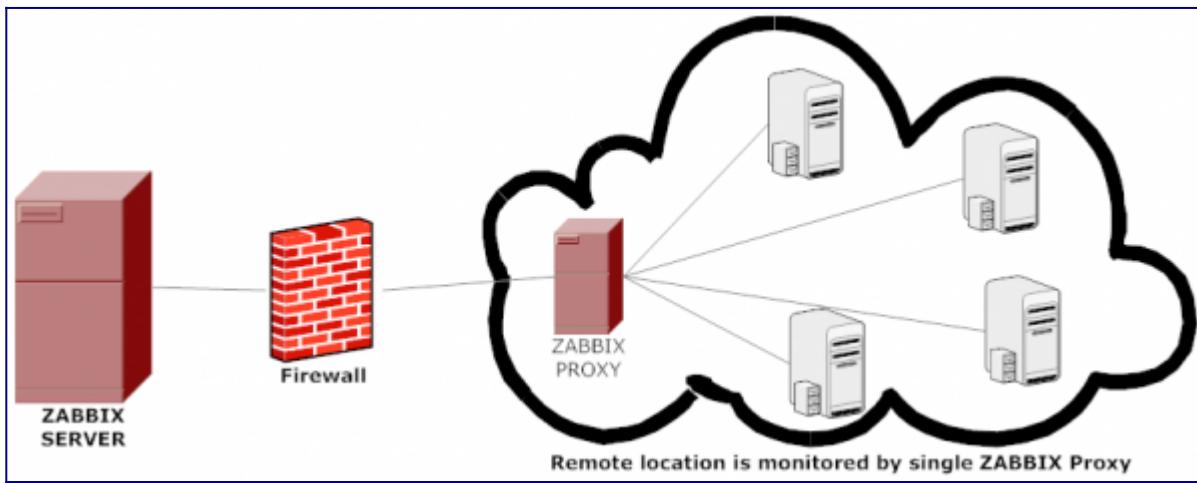
Обзор

Zabbix прокси может собирать данные о производительности и доступности от имени Zabbix сервера. Таким образом, прокси может взять на себя некоторую часть нагрузки по сбору данных и разгрузить Zabbix сервер.

Кроме того, использование прокси это самый простой способ осуществления централизованного и распределенного мониторинга, когда все агенты и прокси отчитываются перед одним Zabbix сервером и все данные собираются в централизованном порядке.

Zabbix прокси можно использовать для:

- Мониторинга удаленных мест
- Мониторинга в местах с ненадежной связью
- Снижения нагрузки на Zabbix сервер при мониторинге тысяч устройств
- Упрощения обслуживания распределенного мониторинга



Прокси требует только одно TCP соединение к Zabbix серверу. Таким образом, будет проще настроить обход брандмауэра, вам потребуется настроить только одно правило в брандмауэре.



Zabbix прокси должен использовать отдельную базу данных. Если указать базу данных Zabbix сервера, то конфигурация будет испорчена.

Все данные собранные прокси, перед отправкой их серверу, хранятся локально. Таким образом, данные не теряются из-за временных проблем со связью с сервером. Параметры *ProxyLocalBuffer* и *ProxyOfflineBuffer* в [файле конфигурации прокси](#) управляют как долго данные будут храниться локально.



Может случиться, что прокси получающий изменения конфигурации из базы Zabbix сервера, будет иметь более свежую конфигурацию, чем сам Zabbix сервер, чья конфигурация может обновляться реже согласно значению параметра [CacheUpdateFrequency](#). В результате, прокси начнет сбор данных и будет отправлять эти данные Zabbix серверу, который будет их игнорировать.

Zabbix прокси - коллектор данных. Он не вычисляет триггеры, не обрабатывает события и не отправляет оповещения. Для обзора возможностей прокси, просмотрите следующую таблицу:

Функция	Поддержка прокси
Элементы данных	
Проверки через Zabbix агент	Да
Проверки через Zabbix агент (активный)	Да ¹
Простые проверки	Да
Элементы данных трассера	Да
SNMP проверки	Да
SNMP трассы	Да
IPMI проверки	Да
JMX проверки	Да
Мониторинг файлов журналов	Да
Внутренние проверки	Да
Проверки через SSH	Да
Проверки через Telnet	Да
Внешние проверки	Да
Зависимые элементы данных	Да ²
Встроенный веб-мониторинг	Да
Обнаружение сети	Да

Функция	Поддержка прокси
Низкоуровневое обнаружение	Да
Удаленные команды	Да
Вычисление триггеров	Нет
Обработка событий	Нет
Корреляция событий	Нет
Отправка оповещений	Нет
Предобработка значений элементов данных	Нет



[1] Чтобы убедиться, что агент запрашивает активные проверки у прокси (а не у сервера), IP адрес прокси должен быть указан в параметре **ServerActive** файла конфигурации агента.



[2] Предобработка значений элементов данных на стороне Zabbix сервера необходима для извлечения требуемого значения с данных мастер элемента данных.

Настройка

Если у вас есть [установленный](#) и [настроенный](#) прокси, то самое время настроить его в веб-интерфейсе Zabbix.

Добавление прокси

Для настройки прокси в веб-интерфейсе Zabbix:

- Перейдите в: Администрирование → Прокси
- Нажмите на Создать прокси

Параметр

Имя прокси

Введите имя прокси. Оно должно совпадать с именем параметра *Hostname* в файле конфигурации прокси.

Выберите режим прокси.

Активный - прокси будет подключаться к Zabbix серверу и запрашивать данные конфигурации

Пассивный - к прокси будет подключаться Zabbix сервер

Режим прокси

Обратите внимание, что (деликатные) данные конфигурации могут стать доступны лицам, имеющим доступ к порту траппера Zabbix сервера, при использовании активного прокси. Такое возможно потому что, кто угодно может представиться активным прокси и запросить данные конфигурации, если аутентификация не выполняется или адреса прокси не ограничены в поле Адрес прокси.

Адрес прокси

Если задано значение, тогда запросы активного прокси принимаются только из этого

Описание

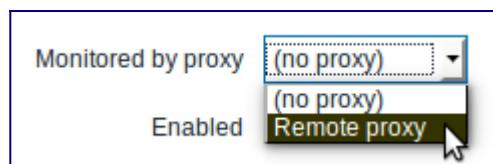
Параметр	Описание
<i>Интерфейс</i>	списка разделенных запятыми IP адресов, дополнительно в CIDR нотации, или DNS имён активного Zabbix прокси.
<i>IP адрес</i>	Это поле доступно только, если выбран активный прокси в поле <i>Режим прокси</i> . Макросы не поддерживаются.
<i>DNS имя</i>	Эта опция поддерживается начиная с Zabbix 4.0.0.
<i>Подключение через</i>	Укажите детали интерфейса пассивному прокси. Это поле доступно только, если в поле <i>Режим прокси</i> выбран пассивный прокси.
<i>Порт</i>	IP адрес пассивного прокси (опционально).
<i>Описание</i>	DNS имя пассивного прокси (опционально).
	Нажав соответствующую кнопку, вы сообщите Zabbix серверу что использовать для получения данных с прокси: IP - Подключение к прокси по IP адресу (рекомендуется) DNS - Подключение к прокси по DNS имени
	Номер TCP/UDP порта пассивного прокси (по умолчанию 10051).
	Введите описание прокси.

Вкладка **Шифрование** позволяет вам требовать шифрованных соединений с прокси.

Параметр	Описание
<i>Подключения к прокси</i>	Каким образом сервер подключается к пассивному прокси: без шифрования (по умолчанию), используя PSK (pre-shared key) или сертификат.
<i>Соединения с прокси</i>	Выберите какой тип подключений разрешен с активного прокси. Можно выбрать несколько типов соединений одновременно (полезно при тестировании и переключении на другой тип соединения). По умолчанию - “Без шифрования”.
<i>Издатель</i>	Разрешенный эмитент сертификата. Сертификат сначала подтверждается CA (центром сертификации). Если он действительный, подписан с помощью CA, тогда можно использовать поле <i>Издатель</i> для более строго ограничения разрешенных CA. Это поле опционально, предназначено для использования, если ваша Zabbix инсталляция использует сертификаты от нескольких CA.
<i>Тема</i>	Разрешенная тема сертификата. Сертификат сначала подтверждается CA. Если он действительный, подписан с помощью CA, тогда можно использовать поле <i>Тема</i> . Если поле пустое, тогда принимается любой сертификат, подписанный настроенным CA.
<i>Идентификатор PSK</i>	Строка идентификации pre-shared key.
<i>PSK</i>	Pre-shared key (строка в шестнадцатеричном формате). Максимальная длина: 512 шестнадцатеричных цифр (256-байт PSK), если Zabbix использует библиотеки GnuTLS или OpenSSL, 64 шестнадцатеричных цифр (32-байт PSK), если Zabbix использует библиотеку mbedtls TLS (PolarSSL). Пример: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Настройка узлов сети

Вы можете указать какие конкретно узлы сети должны наблюдаться через прокси в диалоге [настройки узла сети](#), используя поле *Наблюдение через прокси*.



[Массовое обновление](#) узлов сети является еще одним способом, указать, что узлы сети должны наблюдаться через прокси.

17. Шифрование

Обзор

Zabbix поддерживает шифрование соединений между Zabbix сервером, Zabbix прокси, Zabbix агентом, zabbix_sender и zabbix_get утилитами с использованием Transport Layer Security (TLS) протокола v.1.2. Шифрование поддерживается начиная с Zabbix 3.0. Поддерживаются шифрования на основе сертификата и на основе pre-shared ключа.

Шифрование опционально и настраивается для отдельных компонентов (например, некоторые прокси и агенты можно настроить на использование шифрования с сервером на основе сертификатов, в то время как другие могут использовать шифрование на основе pre-shared ключа, а остальные могут продолжать использовать незашифрованные соединения как и ранее).

Сервер (прокси) может использовать различные настройки с разными узлами сети.

Программы Zabbix демонов слушают один порт для шифрованных и незашифрованных входящих подключений. Добавление шифрования не потребует открывать новые порты на брандмауэрах.

Ограничения

- Приватные ключи хранятся в формате обычного текста в файлах, которые Zabbix компонентычитывают в процессе запуска.
- Введенные pre-shared ключи в веб-интерфейсе Zabbix хранятся в базе данных Zabbix в виде обычного текста.
- Встроенное шифрование не защищает коммуникации:
 - между веб-сервером с веб-интерфейсом Zabbix и веб-браузером на строне пользователя,
 - между Zabbix веб-интерфейсом и Zabbix сервером,
 - между Zabbix сервером (прокси) и базой данных Zabbix.
- В настоящее время каждое незашифрованное соединение открывается с полными TLS переговорами, кэширование сессий и билеты не реализованы.
- Добавление шифрования увеличивает время проверок и действий, в зависимости от сетевых задержек.

Например, если пакет опаздывает на 100мс, тогда открытие TCP соединение и отправка незашифрованного запроса займет около 200мс.

При наличии шифрования на установку TLS соединения добавится около 1000 мс.

Возможно потребуется увеличить время ожидания, в противном случае некоторые элементы данных и действия, выполняющие удаленные скрипты на агентах смогут работать с незашифрованными соединениями,

но не смогут при шифрованном соединении (будет превышено время ожидания).

Компиляция Zabbix с поддержкой шифрования

Для поддержки шифрования Zabbix должен быть скомпилирован и связан с по крайней мере одной крипто библиотекой:

- *mbed TLS* (ранее *PolarSSL*) (версия 1.3.9 или более новые 1.3.x). *mbed TLS* 2.x в настоящее время не поддерживается, это не простая замена ветки 1.3, Zabbix не скомпилируется с *mbed TLS* 2.x.
- *GnuTLS* (с версии 3.1.18)
- *OpenSSL* (с версии 1.0.1)

Библиотека выбирается при помощи опции в скрипте “configure”:

- `--with-mbedtls[=DIR]`

- `--with-gnutls[=DIR]`
- `--with-openssl[=DIR]`

Например, чтобы сконфигурировать исходные коды 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 --with-libxml2 --with-openssl
```

Можно скомпилировать разные компоненты Zabbix с различными крипто библиотеками (например, сервер с *OpenSSL*, агент с *GnuTLS*).



Если вы планируете использовать pre-shared ключи (PSK) рассмотрите возможность использования библиотек *GnuTLS* или *mbed TLS* с компонентами Zabbix, использующих PSK. Библиотеки *GnuTLS* и *mbed TLS* поддерживают наборы шифров PSK с [Совершенной прямой секретностью](#) (Perfect forward secrecy). *OpenSSL* библиотека (версии 1.0.1, 1.0.2c) поддерживает PSK, но доступные наборы шифров PSK не обеспечивают Совершенную прямую секретность.

Управление зашифрованными соединениями

Соединения в Zabbix могут использовать:

- без шифрования (по умолчанию)
- [RSA шифрование на основе сертификатов](#)
- [шифрование на основе PSK](#)

Имеется два важных параметра, которые используются, чтобы указать шифрование между компонентами Zabbix:

- `TLSConnect`
- `TLSAccept`

`TLSConnect` задает какое использовать шифрование и может принимать *одно из 3* значений (*unencrypted*, *PSK*, *certificate*). `TLSConnect` используется в файлах конфигурации Zabbix прокси (в активном режиме задает только подключения к серверу) и Zabbix agentd (при активных проверках). В веб-интерфейсе Zabbix параметр `TLSConnect` является эквивалентом поля *Подключения к узлу сети* с вкладки *Настройка* → *Узлы сети* → <какой-то узел сети> → *Шифрование* и поля *Подключения к прокси* с вкладки *Администрирование* → *Прокси* → <какой-то прокси> → *Шифрование*. Если настроенный тип шифрования для соединения завершится неудачей, другие типы шифрования не будут опробованы.

`TLSAccept` задает какой тип соединений разрешен при входящих подключениях. Тип подключений: *unencrypted*, *PSK*, *certificate*. Можно указать *одно или более* значений. `TLSAccept` используется в файлах конфигурации Zabbix прокси (в пассивном режиме задает только соединения с сервера) и Zabbix agentd (при пассивных проверках). В веб-интерфейсе Zabbix параметр `TLSAccept` является эквивалентом поля *Соединения с узла сети* с вкладки *Настройка* → *Узлы сети* → <какой-то узел сети> → *Шифрование* и поля “*Соединения с прокси*” с вкладки *Администрирование* → *Прокси* → <какой-то прокси> → *Шифрование*.

Как правило, вы настраиваете только один тип шифрования для входящих подключений. Но вы можете захотите переключить режим шифрования, например с незащищенного на основанный на сертификатах с минимальным временем простоя и с возможностью отката. Для этого вы можете задать `TLSAccept=unencrypted, cert` в файле конфигурации agentd и перезапустить агента Zabbix.

Затем вы можете протестировать подключение от `zabbix_get` к агенту, используя сертификат. Если подключение работает, вы можете перенастроить шифрование у этого агента в Zabbix веб-интерфейсе на вкладке *Настройка → Узлы сети → <какой-то узел сети> → Шифрование*, переключив настройку *Подключения к узлу сети* на “Сертификат”.

Когда кэш конфигурации сервера обновится (и конфигурация прокси обновится, если узел сети наблюдается через прокси), тогда подключения к этому агенту будут зашифрованы.

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 узлом сети* задано равным “Сертификат”, 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.

Обзорные настройки шифрования отображаются в веб-интерфейсе Zabbix *Настройка → Узлы сети* по каждому узлу сети по правой стороне, в колонке *ШИФРОВАНИЕ АГЕНТА*. Примеры отображения настроек:

Пример	Подключения К узлу сети	Разрешенные подключения ОТ узла сети	Отклоненные подключения С узла сети
	Незашифровано	Незашифровано	Зашифровано на основе сертификата и PSK
	Зашифровано, на основе сертификата	Зашифровано, на основе сертификата	Незашифровано и на основе PSK
	Зашифровано на основе PSK	Зашифровано на основе PSK	Незашифровано и на основе сертификата
	Зашифровано на основе PSK	Незашифровано и зашифровано на основе PSK	На основе сертификата
	Зашифровано на основе PSK	Незашифровано на основе PSK или зашифровано на основе сертификата	-



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 конфигурируются внутри в процессе запуска Zabbix и зависят от крипто библиотеки, в настоящее время алгоритмы нельзя настраивать пользователями.

Настроенные алгоритмы шифрования по типу библиотеки с более высокого уровня к низкому уровню:

Библиотека	Алгоритмы шифрования сертификатов	Алгоритмы шифрования PSK
<i>mbed TLS</i> (PolarSSL) 1.3.9	TLS-ECDHE-RSA-WITH-AES-128- GCM-SHA256 TLS-ECDHE-RSA-WITH-AES-128- CBC-SHA256 TLS-ECDHE-RSA-WITH-AES-128- CBC-SHA TLS-RSA-WITH-AES-128-GCM- SHA256 TLS-RSA-WITH-AES-128-CBC-SHA256 TLS-RSA-WITH-AES-128-CBC-SHA TLS_ECDHE_RSA_AES_128_GCM_SH A256 TLS_ECDHE_RSA_AES_128_CBC_SH A256 TLS_ECDHE_RSA_AES_128_CBC_SH A1 TLS_RSA_AES_128_GCM_SHA256 TLS_RSA_AES_128_CBC_SHA256 TLS_RSA_AES_128_CBC_SHA1 ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-CCM8 AES128-CCM AES128-SHA256 AES128-SHA	TLS-ECDHE-PSK-WITH-AES-128- CBC-SHA256 TLS-ECDHE-PSK-WITH-AES-128- CBC-SHA TLS-PSK-WITH-AES-128-GCM- SHA256 TLS-PSK-WITH-AES-128-CBC-SHA256 TLS-PSK-WITH-AES-128-CBC-SHA TLS_ECDHE_PSK_AES_128_CBC_SH A256 TLS_ECDHE_PSK_AES_128_CBC_SH A1 TLS_PSK_AES_128_GCM_SHA256 TLS_PSK_AES_128_CBC_SHA256 TLS_PSK_AES_128_CBC_SHA1
<i>GnuTLS</i> 3.1.18		PSK-AES128-CBC-SHA
<i>OpenSSL</i> 1.0.2c		
<i>OpenSSL</i> 1.1.0		ECDHE-PSK-AES128-CBC-SHA256 ECDHE-PSK-AES128-CBC-SHA PSK-AES128-GCM-SHA256 PSK-AES128-CCM8 PSK-AES128-CCM PSK-AES128-CBC-SHA256 PSK-AES128-CBC-SHA

Алгоритмы шифрования при использовании сертификатов:

TLS клиент	<i>mbed TLS</i> (PolarSSL)	<i>GnuTLS</i>	<i>OpenSSL</i> 1.0.2
<i>mbed TLS</i> (PolarSSL))	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256
<i>GnuTLS</i>	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256
<i>OpenSSL</i> 1.0.2	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256	TLS-ECDHE-RSA-WITH-AES- 128-GCM-SHA256

Алгоритмы шифрования при использовании PSK:

TLS клиент	<i>mbed TLS (PolarSSL)</i>	<i>GnuTLS</i>	<i>OpenSSL 1.0.2</i>
<i>mbed TLS</i>	TLS-ECDHE-PSK-WITH-AES-128-CBC-SHA256	TLS-ECDHE-PSK-WITH-AES-128-CBC-SHA256	TLS-PSK-WITH-AES-128-CBC-SHA
<i>GnuTLS</i>	TLS-ECDHE-PSK-WITH-AES-128-CBC-SHA256	TLS-ECDHE-PSK-WITH-AES-128-CBC-SHA256	TLS-PSK-WITH-AES-128-CBC-SHA
<i>OpenSSL</i>	TLS-PSK-WITH-AES-128-CBC-SHA	TLS-PSK-WITH-AES-128-CBC-SHA	TLS-PSK-WITH-AES-128-CBC-SHA

17.1 Использование сертификатов

Обзор

Zabbix может использовать RSA сертификаты в формате PEM, подписанные публичным или внутренним центром сертификации (CA). Проверка сертификата выполняется в отношении с заранее подготовленным CA сертификатом. Самоподписанные сертификаты не поддерживаются. Опционально можно использовать списки отзывов сертификатов (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 запросы на сертификаты и подписывать их, как отзывать сертификаты, всё это вы можете найти в большом количестве различных руководств в сети, например, [OpenSSL PKI Tutorial v1.1](#).

Тщательно продумывайте и тестируйте ваши расширения сертификатов - смотри [Ограничения при использовании расширений X.509 v3 сертификатов](#).

Параметры настройки сертификатов

Параметр	Обязателен	Описание
<i>TLSCAFile</i>	*	Абсолютный путь к файлу, который содержит сертификаты верхнего уровня CA(и) для верификации сертификата узла. При наличии цепочки сертификатов с несколькими членами, они должны быть отсортированы: сначала следуют сертификаты CA низкого уровня за сертификатами более высокого уровня CA(и). Сертификаты из нескольких CA(и) можно включать в один файл.
<i>TLSCRLFile</i>		Абсолютный путь к файлу, который содержит списки отзываемых сертификатов. Смотрите заметки в Списки отзываемых сертификатов (CRL) .
<i>TLCertFile</i>	*	Абсолютный путь к файлу, который содержит сертификат (цепочку сертификатов).
<i>TLSKeyFile</i>	*	В случае цепочки сертификатов с несколькими членами они должны быть отсортированы: сначала сервер, прокси или агент, с последующими CA сертификатами низкого уровня и затем CA сертификаты более высокого уровня.
<i>TLSServerCertIssuer</i>		Абсолютный путь к файлу, который содержит приватный ключ.
<i>TLSServerCertSubject</i>		Задайте права доступа к этому файлу - он должен быть доступен для чтения только пользователю Zabbix.
		Разрешенный эмитент сертификата сервера.
		Разрешенный субъект сертификата сервера.

Настройка сертификата на Zabbix сервере

1. Для того, чтобы проверять сертификаты хостов, Zabbix сервер должен иметь доступ к файлу с их корневыми верхнего уровня самоподписными CA сертификатами. Например, если мы ожидаем

сертификаты от двух независимых корневых CA, мы можем поместить их сертификаты в файл /home/zabbix/zabbix_ca_file, примерно следующим образом:

```
Certificate:
Data:
    Version: 3 (0x2)
    Serial Number: 1 (0x1)
Signature Algorithm: sha1WithRSAEncryption
    Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
    ...
    Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root1 CA
    Subject Public Key Info:
        Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
    ...
X509v3 extensions:
    X509v3 Key Usage: critical
        Certificate Sign, CRL Sign
    X509v3 Basic Constraints: critical
        CA:TRUE
    ...
-----BEGIN CERTIFICATE-----
MIID2jCCAsKgAwIBAgIBATANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
...
9wEzdN8uTrqoyU78gi12npLj08LegRKjb5hFTVm0
-----END CERTIFICATE-----
Certificate:
Data:
    Version: 3 (0x2)
    Serial Number: 1 (0x1)
Signature Algorithm: sha1WithRSAEncryption
    Issuer: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
    ...
    Subject: DC=com, DC=zabbix, O=Zabbix SIA, OU=Development group, CN=Root2 CA
    Subject Public Key Info:
        Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
    ...
X509v3 extensions:
    X509v3 Key Usage: critical
        Certificate Sign, CRL Sign
    X509v3 Basic Constraints: critical
        CA:TRUE
    ...
-----BEGIN CERTIFICATE-----
MIID3DCCAsSgAwIBAgIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQB
...
vdGNYoSfvu41GQAR5Vj5FnRJRzv5XQ0Z3B6894GY1zY=
-----END CERTIFICATE-----
```

2. Поместите цепочку сертификатов Zabbix сервера в файл, например, /home/zabbix/zabbix_server.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=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/6aaQ==
-----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-----
MIID4TCAsmgAwIBAgIBAjANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLGQB
...
dyCeWnvL7u5sd6ffo8iRny0QzbHKmQt/wUtcVIvWXdMIFJM0Hw==
-----END CERTIFICATE-----

```

Здесь первым является сертификат Zabbix сервера, за ним промежуточные CA сертификат.

3. Поместите приватный ключ Zabbix сервера в файл, например,
`/home/zabbix/zabbix_server.key`:

```

-----BEGIN PRIVATE KEY-----
MIIEwAIBADANBgkqhkiG9w0BAQEFAASCBKowggSmAgEAAoIBAQC9tIXIJoVnNXDl
...
IJLkhbybBYEF47MLhffWa7XvZTY=
-----END PRIVATE KEY-----

```

4. Измените параметры TLS в файле конфигурации Zabbix сервера, примерно так:

```

TLSCAFile=/home/zabbix/zabbix_ca_file
TLSCertFile=/home/zabbix/zabbix_server.crt
TLSKeyFile=/home/zabbix/zabbix_server.key

```

Настройка шифрования для Zabbix прокси на основе сертификата

1. Подготовьте файлы с CA сертификатами верхнего уровня, сертификатом (цепочкой) прокси и приватным ключем, как описано в [Настройке сертификата на Zabbix сервере](#). Измените параметры `TLSCAFile`, `TLSCertFile`, `TLSKeyFile` в файле конфигурации прокси соответственно.

2. При активном прокси измените `TLSConnect` параметр:

```
TLSConnect=cert
```

При пассивном прокси измените `TLSAccept` параметр:

```
TLSAccept=cert
```

3. Теперь у вас есть минимальная настройка прокси на основе сертификата. Вы возможно захотите улучшить безопасность прокси, указав параметры `TLSServerCertIssuer` и `TLSServerCertSubject` (смотри [Ограничение разрешенных Эмитента и Субъекта сертификата](#)).

4. В конечном итоге параметры TLS в файле конфигурации прокси могут выглядеть следующим образом:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file
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
TLCertFile=/home/zabbix/zabbix_proxy.crt
TLSKeyFile=/home/zabbix/zabbix_proxy.key
```

5. Настройте шифрование этому прокси в веб-интерфейсе Zabbix:

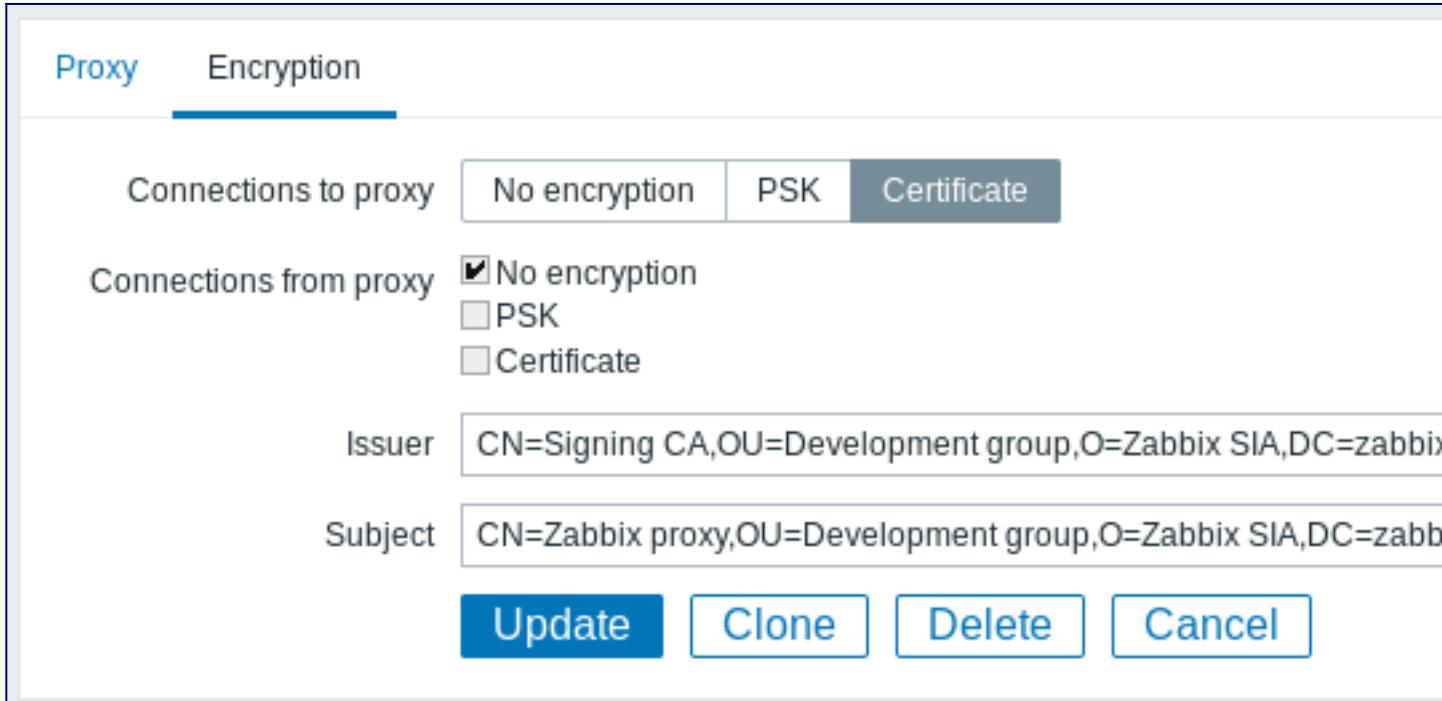
- Перейдите в: *Администрирование* → *Прокси*
- Выберите прокси и нажмите на вкладку **Шифрование**

В примере ниже поля **Эмитент** и **Субъект** заполнены - смотрите [Ограничение разрешенных Эмитента и Субъекта сертификата](#) о том, как использовать эти поля.

При активном прокси

The screenshot shows the 'Encryption' tab of a Zabbix proxy configuration. At the top, there are tabs for 'Proxy' and 'Encryption'. Below the tabs, there are two sections: 'Connections to proxy' and 'Connections from proxy'. Under 'Connections to proxy', there are three buttons: 'No encryption', 'PSK', and 'Certificate', with 'No encryption' being the selected option. Under 'Connections from proxy', there are three checkboxes: 'No encryption', 'PSK', and 'Certificate', with 'Certificate' being checked. Below these sections are two input fields: 'Issuer' containing the value 'CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com' and 'Subject' containing the value 'CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com'. At the bottom of the form are four buttons: 'Update' (blue), 'Clone', 'Delete', and 'Cancel'.

При пассивном прокси



Настройка шифрования для Zabbix агента на основе сертификата

1. Подготовьте файлы с CA сертификатами верхнего уровня, сертификатом (цепочкой) агента и приватным ключем, как описано в [Настройке сертификата на Zabbix сервере](#). Измените параметры `TLSCAFile`, `TLSCertFile`, `TLSKeyFile` в файле конфигурации агента соответственно.

2. При активных проверках измените `TLSConnect` параметр:

`TLSConnect=cert`

При пассивных проверках измените `TLSAccept` параметр:

`TLSAccept=cert`

3. Теперь у вас есть минимальная настройка агента на основе сертификата. Вы возможно захотите улучшить безопасность агента, указав параметры `TLSServerCertIssuer` и `TLSServerCertSubject`.(смотри [Ограничение разрешенных Эмитента и Субъекта сертификата](#)).

4. В конечном итоге параметры TLS в файле конфигурации агента могут выглядеть следующим образом:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file
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
```

(Пример предполагает, что хост наблюдается через прокси, отсюда Субъект сертификата прокси.)

5. Настройте шифрование этому агенту в веб-интерфейсе Zabbix:

- Перейдите в: *Настройка* → *Узлы сети*
- Выберите узел сети и нажмите на вкладку **Шифрование**

В примере ниже поля Эмитент и Субъект заполнены - смотрите [Ограничение разрешенных Эмитента и Субъекта сертификата](#) о том, как использовать эти поля.

Connections to host	<input type="radio"/> No encryption	<input type="radio"/> PSK	<input checked="" type="radio"/> Certificate
Connections from host	<input type="checkbox"/> No encryption <input type="checkbox"/> PSK <input checked="" type="checkbox"/> Certificate		
Issuer	<code>CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com</code>		
Subject	<code>CN=www01,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com</code>		
Update Clone Full clone Delete ...			

Ограничение разрешенных 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, тогда коммуникация может продолжаться. Эмитент и субъект сертификата в этом простом случае не проверяется.

Здесь имеется риск - кто-угодно при наличии действительного сертификата может выдавать себя за другого (например, сертификат хоста можно использовать, чтобы выдавать себя за сервер). Такое поведение может быть приемлемо в небольших средах, где сертификаты подписываются специализированного внутреннего CA и риск действий от чужого имени является минимальным.

Если ваш CA верхнего уровня используется для выдачи других сертификатов, которые не должны приниматься Zabbix или вы хотите снизить риск действий от чужого имени, вы можете ограничить разрешенные сертификаты, указав их строки Эмитента и Субъекта.

Например, вы можете записать в файл конфигурации Zabbix прокси:

```
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
```

При наличии этих настроек активный прокси не будет разговаривать с Zabbix сервером с другими строками Эмитента и Субъекта в сертификате, пассивный прокси не пример запросят от такого сервера.

Несколько заметок о соответствии строк Эмитента и Субъекта:

1. Строки Эмитента и Субъекта проверяются независимо. Обе строки опциональны.
2. Допустимы символы UTF-8.
3. Не указанная строка означает, что принимается любая строка.
4. Строки сравниваются “как-есть”, они должны в точности быть такими же.
5. Шаблоны и регулярные выражения при проверке соответствия не поддерживаются.
6. Реализованы только некоторые требования из [RFC 4514 Lightweight Directory Access Protocol \(LDAP\): String Representation of Distinguished Names](#):

- управляющие символы ““ (U+0022), ‘+’ U+002B, ‘,’ U+002C, ‘;’ U+003B, ‘<’ U+003C, ‘>’ U+003E, ‘\’ U+005C в любом месте в строке.
 - управляющие символы пробела (‘ ’ U+0020) или символ решетки (#’ U+0023) в начале строки.
 - управляющий символ пробела (‘ ’ U+0020) в конце строки.
7. Совпадения не будет, если встречается нулевой символ (U+0000) ([RFC 4514](#) позволяет это).
8. Требования [RFC 4517 Lightweight Directory Access Protocol \(LDAP\): Syntaxes and Matching Rules](#) и [RFC 4518 Lightweight Directory Access Protocol \(LDAP\): Internationalized String Preparation](#) не поддерживаются по причине необходимого объема работы.

Очередность полей в Эмитента и Субъекта строках и форматирование очень важны! Zabbix следует [RFC 4514](#) рекомендации и использует “обратный” порядок этих полей.

Обратный порядок можно продемонстрировать в примере:

```
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
```

Обратите внимание, что он начинается с верхнего уровня (CN), переходит к среднему уровню (OU, O) и заканчивается полями верхнего уровня (DC).

По умолчанию *OpenSSL* отображает поля Эмитента и Субъекта в “нормальном” порядке, в зависимости от использованных дополнительных опций:

```
$ 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
```

Здесь строки Эмитента и Субъекта начинаются с верхнего уровня (DC) и заканчиваются полем нижнего уровня (CN), пробелы и разделители полей зависят от используемых опций. Ни одно из этих значений не будет совпадать в Zabbix полях Эмитента и Субъекта!



Для получения надлежащих строк Эмитента и Субъекта, допустимых в Zabbix, вызовите OpenSSL со специальными опциями `-nameopt`

`esc_2253, esc_ctrl, utf8, dump_nostr, dump_unknown, dump_der, sep_comma_plus, dn_rev, sname:`

```
$ 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
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
```

Теперь строковые поля находятся в обратном” порядке, поля разделены запятой, строки можно использовать в файлах конфигурации Zabbix и в веб-интерфейсе.

Ограничения при использовании расширений X.509 v3 сертификатов

- Расширение **Альтернативное имя субъекта (subjectAltName)**. Альтернативные имена субъектов из *subjectAltName* расширения (такие как IP адрес, e-mail адрес) не поддерживаются Zabbix. В Zabbix проверяется только значение поля “Субъект”

(смотри [Ограничение разрешенных Эмитента и Субъекта сертификата](#)).

Если сертификат использует *subjectAltName* расширение, тогда результат зависит от конкретной комбинации наборов инструментов криптографии с которыми скомпилированы компоненты Zabbix (это расширение может работать, а может и не работать, Zabbix может отказаться принимать такие сертификаты от узлов).

- **Расширение Использование Расширенного Ключа.**

Если используется, то, как правило, необходимо указывать как *clientAuth* (TLS WWW аутентификация клиента), так и *serverAuth* (TLS WWW аутентификация сервера).

Например, при пассивных проверках Zabbix агент выступает в роли TLS сервера, таким образом необходимо указать *serverAuth* в сертификате агента. При активных проверках в сертификате агента необходимо задать *clientAuth*.

GnuTLS выводит предупреждение в случае нарушения использования ключа, но разрешает продолжение соединения.

- **Расширение Ограничения Имени.**

Не все наборы инструментов криптографии поддерживают его. Это расширение может помешать Zabbix в загрузке CA сертификатов, где этот раздел промаркирован как *критический* (зависит от конкретного набора инструментов криптографии).

Списки отзываемых сертификатов (CRL)

Если сертификат скомпрометирован, CA может отзвать его, включив в CRL. Списки CRL можно настраивать в файлах конфигурации сервера, прокси и агента, используя параметр **TLSCRLFile**.

Например:

```
TLSCRLFile=/home/zabbix/zabbix_crl_file
```

где **zabbix_crl_file** может содержать списки CRL от нескольких CA и может выглядеть следующим образом:

```
-----BEGIN X509 CRL-----
MIIB/DCB5QIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixkARkWA2Nv
...
treZeUPjb7LSmZ3K2hpbZN7So0ZcAoHQ3Gwd9npuctg=
-----END X509 CRL-----
-----BEGIN X509 CRL-----
MIIB+TCB4gIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQBGRDY29t
...
CAEebS2CND3ShBedZ8YSil5906JvaDP61lR5lns=
-----END X509 CRL-----
```

CRL файл загружается только при запуске Zabbix. При обновлении CRL требуется перезапуск.



Если компонент Zabbix скомпилирован с *OpenSSL* и используются списки CRL, тогда каждый сертификат верхнего и промежуточного уровней CA в цепочках сертификатов должен иметь соответствующий список CRL (может быть пустым) в **TLSCRLFile**.

Ограничения в использовании CRL расширений

- **Расширение Идентификатор Ключа Полномочий.**

CRL для CA с идентичными именами могут не работать в случае *mbedTLS* (*PolarSSL*), даже с расширением “Идентификатор Ключа Полномочий” (“Authority Key Identifier”).

17.2 Использование pre-shared ключей

Обзор

Каждый pre-shared key (PSK) в Zabbix является парой:

- несекретной строкой идентификации PSK,
- секретным строковым значением PSK.

Строка идентификации PSK является не пустая UTF-8 строка. Например, “PSK ID 001 Zabbix agentd”. Это уникальное имя, по которому этот конкретный PSK упоминается Zabbix компонентами. Не помещайте чувствительную информацию в строку идентификации PSK - она передается незашифрованной по сети.

Значением PSK является сложно угадываемой строкой из шестнадцатеричных цифр, например, “e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9”.

Ограничения размера

В Zabbix имеется ограничение на размеры идентификатора и значения PSK, в некоторых случаях крипто библиотеки 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, введенные как 32 шестнадцатеричных цифры)	2048-бит (256-байт PSK, введенные как 512 шестнадцатеричных цифры)
GnuTLS	128 байт (может включать UTF-8 символы)	-	2048-бит (256-байт PSK, введенные как 512 шестнадцатеричных цифры)
mbed TLS (PolarSSL)	128 UTF-8 символы	-	256-бит (ограничение по умолчанию) (32-байт PSK, введенные как 64 шестнадцатеричных цифры)
OpenSSL	127 байт (может включать UTF-8 символы)	-	2048-бит (256-байт PSK, введенные как 512 шестнадцатеричных цифры)



Веб-интерфейс Zabbix позволяет указать до 128-символов строки идентификатора PSK и до 2048-бит PSK вне зависимости от использованной крипто библиотеки. Если некоторые компоненты Zabbix поддерживают меньшую длину, только пользователь несет ответственность за настройку идентификатора и значения PSK в соответствии с разрешенной длиной в этих компонентах. Превышение ограничений длины приведет к ошибкам в подключениях между Zabbix компонентами.

Перед тем как Zabbix сервер подключится к агенту с использованием PSK, сервер поищет идентификатор PSK и значение PSK этого агента в базе данных (в действительности в кэше конфигурации). После приема подключения 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.



Пользователь несет ответственность за обеспечение того, что не будет двух PSK с одинаковыми строками идентификации, но разными значениями. Несоблюдение этого требования может привести к непредсказуемым сбоям связи между компонентами Zabbix, которые используют PSK с этими строками идентификатора PSK.

Генерирование PSK

Например, можно сгенерировать 256-битный (32 байт) PSK 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 PSK и со связанным с ним PSK. Zabbix ожидает просто PSK в файле с PSK, то есть необходимо удалить строку идентификации и двоеточие (':').

Настройка PSK для соединения сервер-агент (простой пример)

На хосте агента запишите значение PSK в файл, например, /home/zabbix/zabbix_agentd.psk. Этот файл должен содержать PSK в первой строке текста, например:

```
1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952
```

Задайте права доступа к PSK файлу - он должен быть доступен для чтения только пользователю Zabbix.

Измените параметры TLS в файле конфигурации 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 те соединения, которые используют PSK. Идентификатором PSK будет “PSK 001”.

Перезапустите агента. Теперь вы можете протестировать подключение, используя zabbix_get, например:

```
$ zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk \  
--tls-psk-identity="PSK 001" --tls-psk-file=/home/zabbix/zabbix_agentd.psk
```

(Чтобы свести к минимуму времяостояния, смотрите как изменить тип подключения в [Управлении зашифрованными соединениями](#)).

Настройте PSK шифрование этому агенту в веб-интерфейсе Zabbix.

- Перейдите в: *Настройка* → *Узлы сети*
- Выберите узел сети и нажмите на вкладку **Шифрование**

Пример:

Все обязательные поля ввода отмечены красной звёздочкой.

Когда кэш конфигурации синхронизируется с базой данных, новые подключения будут использовать PSK. Проверьте файлы журналов сервера и агента на предмет сообщений об ошибках.

Настройка PSK для соединения сервер - активный прокси (простой пример)

На хосте прокси запишите значение PSK в файл, например, `/home/zabbix/zabbix_proxy.psk`.

Этот файл должен содержать PSK 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:

```
TLSSocket=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

Агент будет подключаться к серверу, используя PSK. Идентификатором PSK будет “PSK 002”.

(Чтобы свести к минимуму время простоя, смотрите как изменить тип подключения в [Управлении зашифрованными соединениями](#)).

Настройте PSK этому прокси в веб-интерфейсе Zabbix. Перейдите на *Администрирование → Прокси*, выберите прокси, перейдите на вкладку “Шифрование”. Переключите “Соединения с прокси” на PSK. Вставьте в поле “Идентификатор PSK” значение “PSK 002” и “e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9” в поле “PSK”. Нажмите “Обновить”.

Перезапустите прокси. Он начнет использовать зашифрованные соединения к серверу на основе PSK. Проверьте файлы журналов сервера и прокси на предмет сообщений об ошибках.

В случае пассивных прокси процедура очень похожа. Разница лишь в том, что необходимо задать `TLSAccept=psk` в файле конфигурации прокси и переключить “Подключения к прокси” в Zabbix веб-интерфейсе на PSK.

17.3 Решение проблем

Общие рекомендации

- В случае проблем начните с понимания какой компонент действует как TLS клиент и какой действует как TLS сервер.
Zabbix сервер, прокси и агенты, в зависимости от взаимодействия между собой, все они могут действовать как TLS сервера, так и TLS клиенты.
Например, Zabbix сервер подключается к агенту для пассивной проверки, действует как TLS клиент. Агент действует в роли TLS сервера.
Zabbix агент запрашивает список активных проверок у прокси, он действует как TLS клиент. Прокси действует как TLS сервер.
Утилиты `zabbix_get` и `zabbix_sender` всегда действуют как TLS клиенты.
- Zabbix использует взаимную аутентификацию.
Каждая сторона проверяет своего партнера и может отказать в соединении.
Например, Zabbix сервер подключается к агенту и может немедленно закрыть соединение, если сертификат агента ошибочен. И наоборот - Zabbix агент принимает подключение с сервера и может закрыть соединение, если агент не доверяет серверу.
- Исследуйте файлы журналов обеих сторон - TLS клиента и TLS сервера.
Сторона, которая отвергает подключение 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.
В подразделах ниже мы попытаемся предоставить (далеко не исчерпывающую) коллекцию сообщений и возможные причины, которые могут помочь в устранении проблем.
Пожалуйста, обратите внимание, что разные наборы инструментов криптографии (OpenSSL, GnuTLS, mbed TLS (PolarSSL)) зачастую дают различные сообщения об ошибках в одинаковых проблемных ситуациях.
Иногда сообщения об ошибках зависят даже от конкретных наборов инструментов криптографии с обеих сторон.

17.3.1 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 *mbed TLS (PolarSSL)* 1.3.11)

```
Get value from agent failed: ssl_handshake(): SSL - The connection indicated an EOF
```

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 *mbed TLS (PolarSSL)*):

```
failed to accept an incoming connection: from 127.0.0.1: ssl_handshake():\  
    SSL - The server has no ciphersuites in common with the client
```

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"
```

17.3.2 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 *mbed TLS (PolarSSL)* и *OpenSSL* узлов:

```
failed to accept an incoming connection: from 127.0.0.1: TLS handshake with 127.0.0.1  
returned error code 1: \file s3_srvr.c line 3251: error:14089086: SSL  
routines:ssl3_get_client_certificate:certificate verify failed: \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 1: \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:paddin
```

CRL устарел или срок действия истечет в процессе операции 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.
```

mbed TLS (PolarSSL), 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]:\
    invalid peer 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]:\
    invalid peer certificate: revoked, CRL expired
```

17.3.3 Проблемы с PSK

PSK содержит нечетное 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 идентификатор 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.
```

В mbed TLS (PolarSSL) передан PSK длиннее 32 байт

В любом журнале Zabbix:

```
ssl_set_psk(): SSL - Bad input parameters to function
```

Используется одинаковая строка идентификации PSK, но разные значения PSK для связи между компонентами (например с OpenSSL)

В журнале на стороне инициатора подключения:

```
...[connect] TCP successful, cannot establish TLS to [[xx.xx.xx.xx]:xxx]: SSL_connect()
returned SSL_ERROR_SSL: file s3_pkt.c line 1472: error:140943FC:SSL
routines:ssl3_read_bytes:sslv3 alert bad record mac: SSL alert number 20: TLS read
fatal alert "bad record mac"
```

В журнале на принимающей стороне:

```
...failed to accept an incoming connection: from xx.xx.xx.xx: TLS handshake returned
error code 1: file s3_pkt.c line 532: error:1408F119:SSL
routines:SSL3_GET_RECORD:decryption failed or bad record mac: TLS write fatal alert
"bad record mac"
```

18. Веб-интерфейс

Обзор

Для более простого доступа Zabbix из любого места и с любой платформы, предусмотрен веб-интерфейс.

Попытка одновременного использования двух инсталляций веб-интерфейсов Zabbix, на одном хосте, но на разных портах, будет неуспешна. Попытка входа в систему во втором веб-интерфейсе завершит сессию в первом и наоборот, если только не изменено имя сессий веб-интерфейса по умолчанию для второго веб-интерфейса в [определениях](#) веб-интерфейса (смотрите ZBX_SESSION_NAME).

18.1 Разделы веб-интерфейса

18.1.1 Мониторинг

Обзор

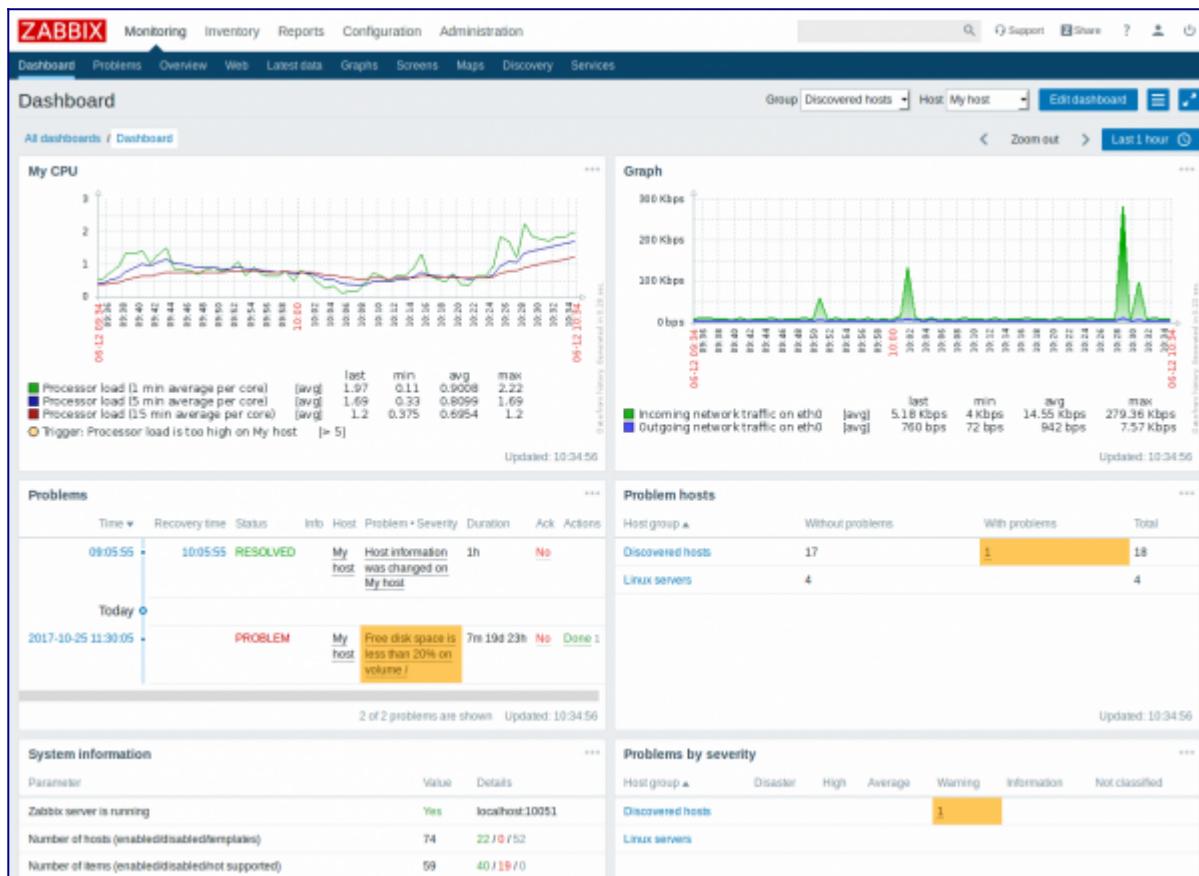
Меню Мониторинг - всё об отображаемых данных. Любая информация, которая настроена в Zabbix на сбор, визуализацию и действия, будет отображаться в различных разделах Мониторинга.

18.1.1.1 ПАНЕЛЬ

Обзор

Раздел Мониторинг → ПАНЕЛЬ предназначена для отображения сводки по всей важной информации.

Панель состоит из виджетов и каждый виджет предназначен для отображения информации определенного типа и источника, таких как обзор, карта, график, часы и так далее.



Виджеты добавляются и редактируются в режиме редактирования панели. Виджеты доступны для просмотра в режиме чтения панели.

Хотя в одной панели вы можете группировать виджеты с разных источников для быстрого обзора, также возможно создать нескольких панелей, которые будут содержать разные наборы обзоров, и переключаться между ними.

Период времени, который отображается в виджетах графиков управляется при помощи [выбора периода времени](#), который расположен выше виджетов. Подпись к выбору периода времени, расположенная справа, отображает текущий выбранный период времени. Нажатие на подписи к вкладке позволяет раскрыть или свернуть выбор периода времени.

Обратите внимание, когда панель отображается в режиме киоска (доступно из полноэкранного режима) и отображаются только виджеты, период графика можно увеличить, применив двойное нажатие на самом графике.

Просмотр панелей

Для доступа ко всем добавленным панелям нажмите на ссылку *Все панели* ниже заголовка раздела.

The screenshot shows a list of dashboards with a header 'Dashboards'. A 'Create dashboard' button is at the top right. Below it is a table with columns for Name, Dashboard, and Dashboard2. The row for 'Dashboard3' has a checked checkbox and is highlighted with a yellow background. At the bottom left are buttons for 'I selected' and 'Delete', and at the bottom right is a message 'Displaying 3 of 3 found'.

Для просмотра отдельной панели нажмите на ее имя в списке со всеми панелями.

Для удаления одного или нескольких панелей отметьте соответствующие панели и нажмите на *Удалить* ниже списка.

Создание панели

При просмотре всех панелей вы можете нажать на кнопку *Создать панель*, чтобы создать новую панель:

The screenshot shows a 'New dashboard' window with a title bar containing 'New dashboard', a gear icon, and buttons for '+ Add widget', 'Save changes', and 'Cancel'. Below the title bar is a link 'All dashboards'. The main area is a large white space with a dashed border, containing the text 'Add a new widget' in blue.

Изначально панель пустая. Вы можете добавлять виджеты на панель:

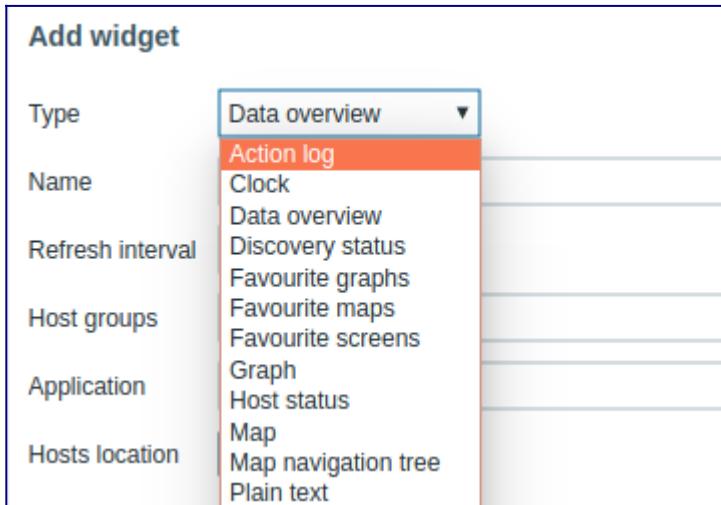
- нажав на вновку *Добавить виджет*
- нажав на ссылку *Добавить новый виджет* при замене места виджета

Нажмите на кнопку *Сохранить изменения*, чтобы сохранить панель. Если вы нажмете на *Отмена*, панель не будет создана.

Добавление виджетов

Для добавления виджета на панель:

- Нажмите на кнопку/ссылку *Добавить виджет* в режиме редактирования панели
- Выберите *Type* виджета
- Задайте параметры виджету
- Нажмите на *Добавить*



На панель можно добавить следующие виджеты:

- [Журнал действий](#)
- [Часы](#)
- [Обзор данных](#)
- [Состояние обнаружения](#)
- [Избранные графики](#)
- [Избранные карты](#)
- [Избранные комплексные экраны](#)
- [График](#)
- [Узлы сети с проблемами](#)
- [Карта сети](#)
- [Дерево навигации карт сетей](#)
- [Простой текст](#)
- [Проблемы](#)
- [Информация о системе](#)
- [Проблемы по важности](#)
- [Обзор триггеров](#)
- [URL](#)
- [Веб-мониторинг](#)

В режиме редактирования панели виджеты можно перемещать в пределах самой панели, нажав на полосу заголовка виджета и переместив ее в новое место. Также, вы можете нажать на следующие кнопки в виджете, чтобы:

- - изменить виджет;

-  - удалить виджет

Нажмите на *Сохранить изменения* у панели, чтобы закрепить любые изменения виджетов.

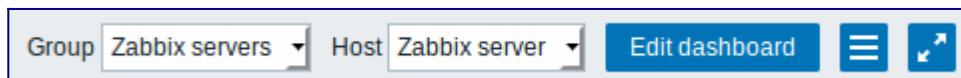
Динамические элементы

Когда настраиваете некоторые виджеты:

- Графики (простой и пользовательский)
- Простой текст
- URL

имеется дополнительная опция, называемая *Динамический элемент*. Вы можете отметить эту опцию и сделать виджет динамическим - то есть с возможностью отображения разного содержимого, основываясь на выбранном узле сети.

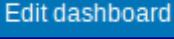
Теперь, после сохранения панели, вы заметите появление двух новых выпадающих меню сверху панели, которые позволяют выбрать группу узлов сети/узел сети:



Таким образом у вас имеется виджет, который может отображать содержимое на основе данных с узла сети, который выбран из выпадающего списка. Преимуществом этого метода является то, что вам не потребуется создавать дополнительные виджеты только потому, что вы хотите увидеть те же самые графики содержащие данные с разных узлов сети.

Просмотр и редактирование панели

При просмотре одной панели доступны следующие опции:

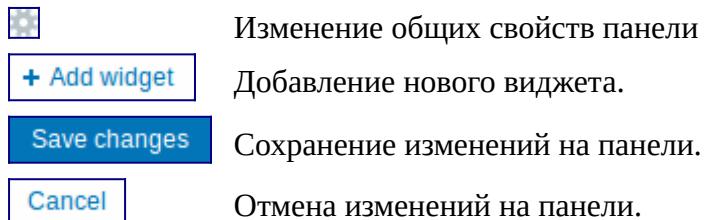
 Edit dashboard	Переключение панели в режим редактирования.
	Открыть меню действий.
	Изменение настроек общего доступа к панели. Панели можно сделать публичными или приватными.
	Публичные панели доступны всем пользователям.
<i>Общий доступ</i>	Приватные панели видны только их владельцам. Владельцы могут предоставлять доступ к приватным панелям другим пользователям и группам пользователей. Для получения более подробным сведений о настройке общего доступасмотрите раздел настройки карты сети.
	Создание новой панели.
<i>Создать новую</i>	Сначала вам будет предложено ввести общие свойства новой панели - владелец и имя. Затем, новая панель откроется в режиме редактирования и вы сможете добавить новые виджеты.
	Создание новой панели копированием свойств с уже существующей панели.
<i>Клонировать</i>	Сначала вам будет предложено ввести общие свойства новой панели - владелец и имя. Затем, откроется новая панель в режиме редактирования со всеми виджетами оригинальной панели.
 Удалить	Удаление панели.
	Отображение панели в режиме полного экрана.
	Отображение панели в режиме киоска. В этом режиме отображаются только виджеты.
	Кнопка режима киоска появляется после активации полноэкранного режима.
	Чтобы выйти из режима киоска, переместите курсор до появления  кнопки

выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

Режим редактирования открывается:

- при создании новой панели
- при нажатии на кнопку *Редактировать панель* у уже существующей панели

В режиме редактирования панели доступны следующие опции:



Права доступа к панелям

Права доступа к панелям обычным и Zabbix Администратором пользователям, начиная с Zabbix 3.4.2, ограничены следующим образом:

- Они могут видеть и клонировать панели, если у них имеются по крайней мере права на ЧТЕНИЕ к ним;
- Они могут редактировать и удалять панели только, если у них имеются права на ЧТЕНИЕ/ЗАПИСЬ к ним;
- Они не могут изменить владельца панели.

До Zabbix 3.4.2, права доступа пользователей уровня Администратора не ограничены подобным образом.

Меню узлов сети

При нажатии на узел сети в виджете *Проблемы* отобразится меню узла сети. Оно включает в себя ссылки на пользовательские скрипты, последние данные, триггеры, инвентарные данные, графики и комплексные экраны этого узла сети.

Problems

Time ▾	Info	Host	Problem • Severity
2018-06-12 13:10:07		Zabbix server	Free disk space is less than 20% on volume /
2018			
2017-10-25 11:30:05		New host	Free disk space is less than 20% on

Data overview

Items

GO TO

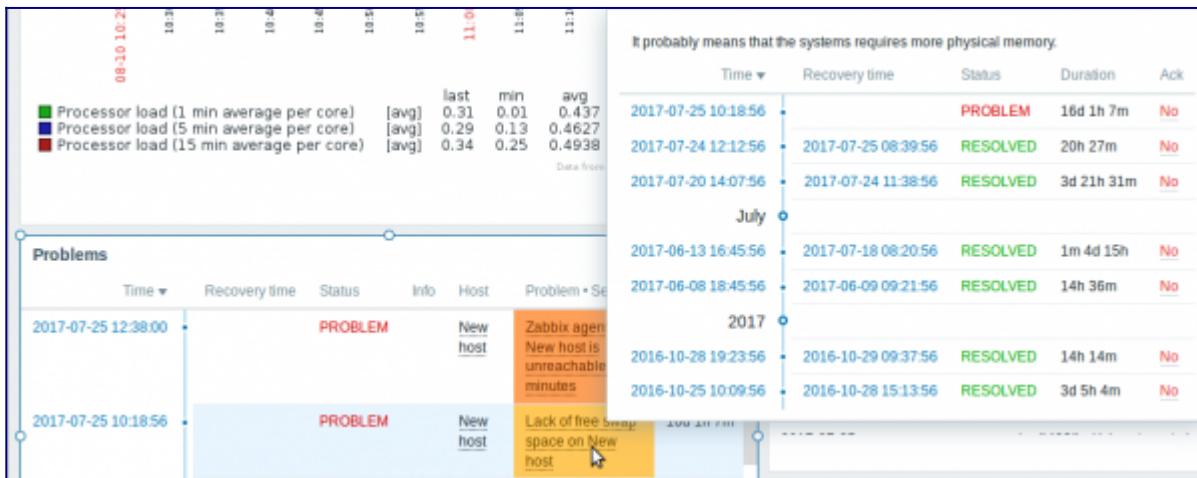
- Host inventory
- Latest data
- Problems
- Graphs
- Host screens

Меню у узлах сети доступно при нажатии на узел сети в некоторых других разделах веб-интерфейса:

- Мониторинг → [Проблемы](#)
- Мониторинг → [Проблемы](#) → Детали события
- Мониторинг → [Обзор \(с Узлы сети: слева\)](#)
- Мониторинг → [Последние данные](#)
- Мониторинг → [Комплексные экраны](#) (в виджетах *События у узла сети* и *События в группах узлов сети*)
- Мониторинг → [Карты сети](#)
- Отчеты → [100 наиболее активных триггеров](#)

Всплывающее окно событий о проблемах

Всплывающее окно событий о проблемах включает в себя список событий о проблемах по триггеру и, если задано, описание триггера и URL с возможностью перехода по ссылке.



Чтобы вызвать всплывающее окно событий о проблемах:

- наведите курсор мыши на имя проблемы в колонке *Проблема-важность* в виджете *Проблемы*. Всплывающее окно исчезнет сразу как вы уберете курсор мыши с имени проблемы.
- нажмите на имя проблемы в колонке *Проблема-Важность* в виджете *Проблемы*. Всплывающее окно исчезнет только, если вы нажмете на имя проблемы еще раз.

18.1.1.1 Виджеты ПАНЕЛИ

Обзор

Этот раздел перечисляет доступные элементы [панелей](#) и содержит подробную информацию по настройке этих элементов.

Следующие параметры общие для всех виджетов:

Имя Укажите имя виджета.

Интервал обновления Задайте интервал обновления по умолчанию. Интервалы обновления по умолчанию для виджетов варьируются от *Без обновления* до *15 минут* в зависимости от типа виджета. Например: *Без обновления* для URL виджета, *1 минута* для виджета журнала действий, *15 минут* для виджета часов.

Интервалы обновлений для виджетов можно задать значением по умолчанию для всех соответствующих пользователей, а также каждый пользователь может выбрать свой собственный интервал обновления:

- Чтобы установить значение по умолчанию для всех соответствующих пользователей, переключитесь в режим редактирования (Нажмите кнопку “Редактировать панель”, найдите необходимый виджет, нажмите кнопку “Изменить”, откроется диалог *Изменение виджета*) и выберите требуемый интервал обновления из списка выпадающего меню.
- Настройка уникального интервала обновления каждому пользователю отдельно возможна в режиме просмотра, необходимо нажат на кнопку для соответствующего виджета.

Уникальный интервал обновления, заданный пользователем, имеет более высокий приоритет над настройкой виджета и однажды заданный, он всегда сохраняется при настройке виджета.

Журнал действий

В виджете журнала действий вы можете отобразить детали операций действий (оповещения, удаленные команды). Этот виджет является копией информации с *Администрирование → Аудит*.

Для настройки, выберите тип *Журнал действий*:

Add widget

Type	Action log
Name	Action log
Refresh interval	Default (1 minute)
Sort entries by	Time (descending)
* Show lines	25

Вы можете указать следующие опции:

Сортировка записей по:

Время (по убыванию или по возрастанию)

Сортировка записей по

Тип (по убыванию или по возрастанию)

Состояние (по убыванию или по возрастанию)

Получатель (по убыванию или по возрастанию).

Отображать строки

Задает как много строк из журнала действий будет отображаться в виджете.

Часы

В виджете часов вы можете отобразить локальное, серверное или время конкретного узла сети.

Для настройки, выберите тип Часы:

Add widget	
Type	Clock
Name	Local time
Refresh interval	Default (15 minutes)
Time type	Local time

Вы можете указать следующие опции:

Тип времени Выберите локальное, серверное или время конкретного узла сети.

Элемент данных Выберите элемент данных для отображения времени. Для отображения времени узла сети, используйте элемент данных [system.localtime\[local\]](#). Элемент данных должен существовать у узла сети.

Это поле доступно только при выбранной опции *Время узла сети*.

Обзор данных

В виджете обзора данных вы можете отобразить последние данные группы узлов сети. Этот виджет является копией информации с Мониторинг → Обзор (при выбранном там Типе равном *Данные*).

Для настройки, выберите тип *Обзор данных*:

Add widget

Type: Data overview

Name: Data overview

Refresh interval: Default (1 minute)

Host groups: Discovered hosts type here to search

Application: CPU

Show suppressed problems:

Hosts location: Left Top

Вы можете указать следующие опции:

- Группы узлов сети** Выберите группы узлов сети. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню. Прокрутите список, чтобы выбрать необходимую. Нажмите на 'x', чтобы удалить выбранную.
- Группа элементов данных** Введите имя группы элементов данных.
- Отображение подавленных проблем** Отметьте, чтобы отображались проблемы, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.
- Расположение узлов сети** Выберите расположение узлов сети - слева или сверху.

Состояние обнаружения

Этот виджет отображает сводку по активным правилам сетевого обнаружения.

Избранные графики

Этот виджет состоит из ссылок на наиболее необходимые графики. Чтобы добавить ссылку в этот список, при просмотре графика необходимо нажать на кнопку *Добавить в избранное*.

Избранные карты сети

Этот виджет состоит из ссылок на наиболее необходимые карты сети. Чтобы добавить ссылку в этот список, при просмотре карты сети необходимо нажать на кнопку *Добавить в избранное*.

Избранные комплексные экраны

Этот виджет состоит из ссылок на наиболее необходимые комплексные экраны. Чтобы добавить ссылку в этот список, при просмотре комплексного экрана необходимо нажать на кнопку *Добавить в избранное*.

График

В виджете графика вы можете отобразить один пользовательский график или один простой график.

Для настройки, выберите тип *График*:

Edit widget

Type	Graph
Name	CPU
Refresh interval	Default (1 minute)
Source	Graph Simple graph
* Graph	My host: CPU load <input type="button" value="Select"/>
Dynamic item	<input checked="" type="checkbox"/>
<input type="button" value="Apply"/> <input type="button" value="Cancel"/>	

Вы можете указать следующие опции:

Источник

Выберите тип графика:

График - пользовательский график

Простой график - простой график

График

Выберите отображаемый пользовательский график.

Эта опция доступна, если в *Источник* выбрана опция 'График'.

Элемент данных

Выберите элемент данных, который необходимо отображать в виде простого графика.

Эта опция доступна, если в *Источник* выбрана опция 'Простой график'.

Динамический элемент

Укажите графику отображать разные данные в зависимости от выбранного узла сети.

Карта сети

В виджете карты сети вы можете отображать:

- одну добавленную карту сети
- одну из добавленных карт сети в дереве навигации карт сетей (при нажатии на имени карты в дереве).

Для настройки, выберите тип *Карта сети*:

Edit widget

Type	Map
Name	Local network
Refresh interval	Default (15 minutes)
Source type	Map Map navigation tree
* Map	Local network <input type="button" value="Select"/>
<input type="button" value="Apply"/> <input type="button" value="Cancel"/>	

Вы можете указать следующие опции:

Тип источника Выберите отображение:

Карта - карта сети

Дерево навигации карт сетей - одна из карт сети в выбранном дереве навигации

карт сетей

Карта сети

Выберите отображаемую карту сети.

Эта опция доступна, если в *Тип источника* выбрана опция 'Карта сети'.

Фильтр

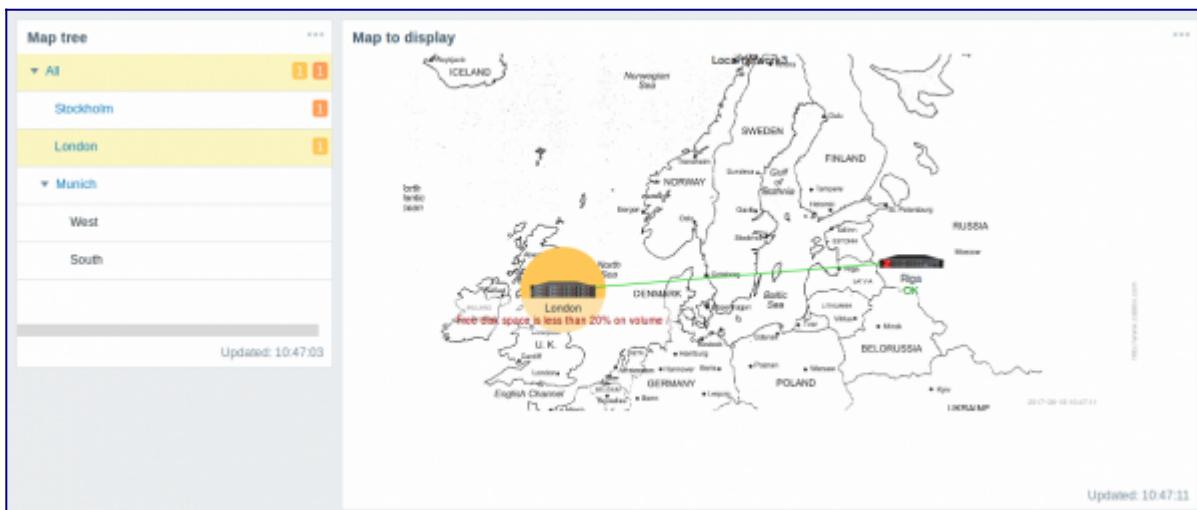
Выберите дерево навигации карт сетей из которого будут отображаться карты.

Эта опция доступна, если в *Дерево навигации карт сетей* выбрана опция 'Карта сети'.

Дерево навигации карт сетей

Этот виджет позволяет создавать иерархию существующих карт сети и одновременно отображать статистику проблем по каждой включенной карте или группе карт сети.

Этот виджет станет более мощным, если вы к дереву навигации присоедините виджет *Карта сети*. В этом случае при нажатии на имени карты в дереве навигации, эта карта сети отобразится в полном виде в виджете *Карта сети*.



Статистика с карты верхнего уровня в иерархии отображает сумму проблем со всех подкарт, а также ее собственные проблемы.

Для настройки виджета дерева навигации, выберите тип *Дерево навигации карт сетей*:

Add widget

Type	Map navigation tree ▾
Name	Map tree
Refresh interval	Default (15 minutes) ▾
Show unavailable maps	<input type="checkbox"/>

Вы можете указать следующие опции:

Отображать недоступные карты

Отметьте эту опцию, чтобы отображались карты сети, к которым у пользователя нет прав чтения.

Недоступные карты сети будут отображаться в дереве навигации с серой иконкой.

Обратите внимание на то что, если эта опция отмечена, доступные подкарты отображаются даже, если карта верхнего уровня недоступна. Если опция не выбрана, доступные подкарты от недоступной карты верхнего уровня не будут отображаться вообще.

Количество проблем вычисляется на основе доступных карт и доступных элементов карт сети.

Простой текст

В виджете простого текста вы можете отобразить последние данные по элементу данных как простой текст.

Для настройки, выберите тип *Простой текст*:

The screenshot shows the 'Add widget' dialog with the following configuration:

- Type: Plain text
- Name: Text item
- Refresh interval: Default (1 minute)
- * Items: Zabbix server: Available memory, Zabbix server: CPU idle time (selected items)
- Items location: Top (selected)
- * Show lines: 25
- Show text as HTML:
- Dynamic items:

At the bottom right are 'Add' and 'Cancel' buttons.

Вы можете указать следующие опции:

Элементы данных

Выберите элементы данных.

Размещение элементов данных

Выберите размещение выбранных элементов данных в виджете.

Отображать строки

Укажите как много последних данных будут отображаться в виджете.

Показать текст в виде HTML

Задайте отображение текста HTML кодом.

Динамический элемент

Задайте отображение разных данных в зависимости от выбранного узла сети.

Узлы сети с проблемами

В виджете информации об узлах сети вы можете отобразить высокоуровневую информацию по доступности узлов сети.

Для настройки, выберите тип *Узлы сети с проблемами*:

Add widget

Type	Problem hosts
Name	Problem hosts
Refresh interval	Default (1 minute)
Host groups	<input type="text" value="type here to search"/> <input type="button" value="Select"/>
Exclude host groups	<input type="text" value="type here to search"/> <input type="button" value="Select"/>
Hosts	<input type="text" value="type here to search"/> <input type="button" value="Select"/>
Problem	<input type="text"/>
Severity	<input type="checkbox"/> Not classified <input type="checkbox"/> Information <input type="checkbox"/> Warning <input type="checkbox"/> Average <input type="checkbox"/> High <input type="checkbox"/> Disaster
Show suppressed problems	<input type="checkbox"/>
Hide groups without problems	<input type="checkbox"/>
Problem display	<input type="radio"/> All <input type="radio"/> Separated <input type="radio"/> Unacknowledged only
<input type="button" value="Add"/> <input type="button" value="Cancel"/>	

Вы можете указать следующие опции:

Параметр

Описание

Группы узлов сети

Укажите отображаемые группы узлов сети в виджете. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню.

Исключить группы узлов сети

Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.

Данные узлов сети из этих групп узлов сети будут отображаться в виджете. Если группы узлов сети не заданы, будут отображаться все группы узлов сети.

Укажите группы узлов сети, которые необходимо исключить из виджета. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню.

Узлы сети

Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.

Данные узлов сети из этих групп узлов сети не будут отображаться в виджете. Например, узлы сети 001, 002, 003 могут входить в Группа А и также узлы сети 002, 003 в Группу В. Если мы выберем *отображение* Группа А и в то же время *исключение* Группа В, на Панели будут отображены данные только с узла сети 001.

Проблема

Укажите отображаемые узлы сети в виджете. Это поле с функцией автодополнения, таким образом после начала ввода имени узла сети, будет автоматически предложен список совпадающих узлов сети в выпадающем меню. Если узлы сети не заданы, будут отображаться все узлы сети.

Вы можете ограничить количество отображаемых проблем узлов сети по имени проблемы. Если вы зададите здесь строку, будут отображаться только те узлы

Параметр	Описание
<i>Важность</i>	сети с проблемами в именах которых СОДЕРЖИТСЯ соответствующая заданная строка. Макросы не раскрываются.
<i>Отображение подавленных проблем</i>	Отметьте отображаемые важности проблем в виджете.
<i>Скрыть группы без проблем</i>	Отметьте опцию <i>Скрыть группы без проблем</i> , чтобы в виджете скрывались данные с групп узлов сети в которых проблемы отсутствуют. Отображать количество проблем как: Все - будет отображаться общее количество проблем Отдельно - количество неподтвержденных проблем будет отображаться отдельным числом от общего количества проблем Только неподтвержденные - будет отображаться только количество неподтвержденных проблем.
<i>Отображение проблем</i>	

Проблемы

В этом виджете вы можете отображать текущие проблемы. Этот виджет является копией информации с *Мониторинг → Проблемы*.

Для настройки, выберите тип *Проблемы*:

Edit widget

The dialog box contains the following configuration options:

- Type: Problems
- Name: Problems
- Refresh interval: Default (1 minute)
- Show: Recent problems (selected), Problems, History
- Host groups: Discovered hosts (with a search input field and a Select button)
- Exclude host groups: type here to search (with a Select button)
- Hosts: type here to search (with a Select button)
- Problem: (empty input field)
- Severity: Not classified, Information, Warning, Average, High, Disaster (checkboxes)
- Tags: And/Or (selected), Or (radio button), tag (input field), Like (radio button), Equal (radio button), value (input field), Remove (button), Add (link)
- Show tags: None, 1, 2, 3 (radio buttons)
- Tag name: Full (selected), Shortened, None (radio buttons)
- Tag display priority: Comma-separated list (input field)
- Show suppressed problems: checked
- Show unacknowledged only: unchecked
- Sort entries by: Time (descending) (dropdown menu)
- Show timeline: checked
- Show lines: 25 (input field)
- Buttons: Apply, Cancel

Вы можете ограничить количество отображаемых проблем в этом виджете несколькими способами - при помощи состояния проблем, имени проблем, важности, группы узлов сети, узлу сети, тегу событий, состоянию подтверждений и т.п.

Параметр	Описание
Показать	Фильтрация по состоянию проблем: Недавние проблемы - отображение нерешенных и недавно решенных проблем (по умолчанию) Проблемы - Отображение нерешенных проблем История - отображение истории всех событий
Группы узлов сети	Укажите отображаемые группы узлов сети в виджете. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов

Параметр	Описание
<i>Исключить группы узлов сети</i>	сети в выпадающем меню. Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Данные узлов сети из этих групп узлов сети будут отображаться в виджете. Если группы узлов сети не заданы, будут отображаться все группы узлов сети.
<i>Узлы сети</i>	Укажите группы узлов сети, которые необходимо исключить из виджета. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню. Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети. Данные узлов сети из этих групп узлов сети не будут отображаться в виджете. Например, узлы сети 001, 002, 003 могут входить в Группа А и также узлы сети 002, 003 в Группу В. Если мы выберем <i>отображение Группа А</i> и в то же время <i>исключение Группа В</i> , на Панели будут отображены данные только с узла сети 001.
<i>Проблема</i>	Укажите отображаемые узлы сети в виджете. Это поле с функцией автодополнения, таким образом после начала ввода имени узла сети, будет автоматически предложен список совпадающих узлов сети в выпадающем меню.
<i>Важность</i>	Если узлы сети не заданы, будут отображаться все узлы сети. Вы можете ограничить количество отображаемых проблем узлов сети по имени проблемы. Если вы зададите здесь строку, будут отображаться только те узлы сети с проблемами в именах которых СОДЕРЖИТСЯ соответствующая заданная строка. Макросы не раскрываются.
<i>Теги</i>	Отметьте отображаемые важности проблем в виджете. Укажите имя и значение тега событий, чтобы ограничить количество отображаемых проблем. Для добавления нескольких значений и имен тегов, нажмите на <i>Добавить</i> . Имеется два типа вычислений для нескольких условий: И/Или - все условия совпадают, условия у которых имеется одинаковое имя тега будут сгруппированы по условию Или Или - достаточно совпадения одного условия Имеется два способа соответствия значения тегов: Содержит - соответствие похожей строке Равно - точное регистрозависимое соответствие строки При фильтрации теги указанные здесь будут сначала отображены с проблемой, если только не предопределены списком <i>Приоритет отображения тегов</i> (см. ниже). Выбор количества отображаемых тегов: Нет - колонка Теги отсутствует в Мониторинг → Проблемы 1 - колонка Теги содержит один тег 2 - колонка Теги содержит два тега 3 - /колонка Теги содержит три тега Чтобы увидеть все теги по проблеме наведите курсор мыши на иконку с тремя точками. Выберите режим отображения имени тега: Полный - имена тегов и их значения отображаются в полном формате Сокращённое - имена тегов сокращены до 3 символов; значения тегов отображаются в полном формате Нет - отображаются только значения тегов; без имён Ведите приоритет отображения тегов для проблемы, в виде разделённого
<i>Отображать теги</i>	
<i>Имя тега</i>	
<i>Приоритет</i>	

Параметр	Описание
отображения тегов	запятыми списка тегов (например: Сервисы, Приложения, Приложение). Необходимо использовать только имена тегов, не значения. Теги из этого списка всегда отображаются первыми, переопределяя естественную сортировку по алфавиту.
Отображение подавленных проблем	Отметьте, чтобы отображались проблемы, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.
Отображать только неподтвержденные	Отметьте эту опцию, чтобы отображались только неподтвержденные проблемы.
Сортировка записей по	Сортировка записей по: Время (по убыванию или по возрастанию) Важность (по убыванию или по возрастанию) Имя проблемы (по убыванию или по возрастанию) Узел сети (по убыванию или по возрастанию).
Отображать выбор времени	Отметьте, чтобы отображался визуальный выбор времени.
Отображать строки	Укажите количество отображаемых строк проблем.

Проблемы по важности

В этом виджете вы можете отобразить проблемы по важности. Вы можете ограничить какие узлы сети и триггеры будут отображаться в виджете и определить как будут подсчитываться проблемы.

Для настройки, выберите тип *Проблемы по важности*:

Add widget

Type	Problems by severity
Name	Problems by severity
Refresh interval	Default (1 minute)
Host groups	type here to search <input type="button" value="Select"/>
Exclude host groups	type here to search <input type="button" value="Select"/>
Hosts	type here to search <input type="button" value="Select"/>
Problem	
Severity	<input type="checkbox"/> Not classified <input type="checkbox"/> Information <input type="checkbox"/> Warning <input type="checkbox"/> Average <input type="checkbox"/> High <input type="checkbox"/> Disaster
Show suppressed problems	<input type="checkbox"/>
Hide groups without problems	<input type="checkbox"/>
Problem display	All <input checked="" type="radio"/> Separated <input type="radio"/> Unacknowledged only
Show timeline	<input checked="" type="checkbox"/>
<input type="button" value="Add"/> <input type="button" value="Cancel"/>	

Вы можете указать следующие опции:

Параметр	Описание
<i>Группы узлов сети</i>	Укажите отображаемые группы узлов сети в виджете. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню. Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.
<i>Исключить группы узлов сети</i>	Данные узлов сети из этих групп узлов сети будут отображаться в виджете. Если группы узлов сети не заданы, будут отображаться все группы узлов сети. Укажите группы узлов сети, которые необходимо исключить из виджета. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню. Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.
<i>Узлы сети</i>	Данные узлов сети из этих групп узлов сети не будут отображаться в виджете. Например, узлы сети 001, 002, 003 могут входить в Группа А и также узлы сети 002, 003 в Группу В. Если мы выберем <i>отображение</i> Группа А и в то же время <i>исключение</i> Группа В, на Панели будут отображены данные только с узла сети 001. Укажите отображаемые узлы сети в виджете. Это поле с функцией автодополнения, таким образом после начала ввода имени узла сети, будет автоматически предложен список совпадающих узлов сети в выпадающем меню. Если узлы сети не заданы, будут отображаться все узлы сети.
<i>Проблема</i>	Вы можете ограничить количество отображаемых проблем узлов сети по имени проблемы. Если вы зададите здесь строку, будут отображаться только те узлы сети с проблемами в именах которых СОДЕРЖИТСЯ соответствующая заданная строка. Макросы не раскрываются.
<i>Важность</i>	Отметьте отображаемые важности проблем в виджете.
<i>Отображение подавленных проблем</i>	Отметьте, чтобы отображались проблемы, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.
<i>Скрыть группы без проблем</i>	Отметьте опцию <i>Скрыть группы без проблем</i> , чтобы в виджете скрывались данные с групп узлов сети в которых проблемы отсутствуют.
<i>Отображение проблем</i>	Отображать количество проблем как: Все - будет отображаться общее количество проблем Отдельно - количество неподтвержденных проблем будет отображаться отдельным числом от общего количества проблем Только неподтвержденные - будет отображаться только количество неподтвержденных проблем.
<i>Отображать выбор времени</i>	Отметьте, чтобы отображался визуальный выбор времени.

Информация о системе

В виджете Информация о системе вы можете отобразить высокоуровневую информацию о Zabbix и Zabbix сервере.

Для настройки, выберите тип *Информация о системе*:

Add widget

Type System information ▾

Name System information

Refresh interval Default (15 minutes) ▾

Add

Cancel

Обзор триггеров

В виджете обзора триггеров вы можете отобразить состояния триггеров из группы узлов сети. Этот элемент является копией информации с Мониторинг → Обзор (при выбранном там Типе равном Триггеры).

Для настройки, выберите тип *Обзор триггеров*:

Add widget

Type Trigger overview ▾

Name Trigger overview

Refresh interval Default (1 minute) ▾

Show Recent problems Problems Any

Host groups type here to search Select

Application

Show suppressed problems

Hosts location Left Top

Add

Cancel

Вы можете указать следующие опции:

Фильтрация по состоянию проблем:

Недавние проблемы - отображение нерешенных и недавно решенных проблем (по умолчанию)

Проблемы - Отображение нерешенных проблем

Любые - отображение истории всех событий

Показать

Группы узлов сети

Группа элементов данных

Отображение подавленных проблем

Расположение узлов сети

Выберите группу(ы) узлов сети. Это поле с функцией автодополнения, таким образом после начала ввода имени группы узлов сети, будет автоматически предложен список совпадающих групп узлов сети в выпадающем меню.

Введите имя группы элементов данных.

Отметьте, чтобы отображались проблемы, которые в противном случае были подавлены (не показаны) по причине обслуживания узлов сети.

Выберите расположение узлов сети - слева или сверху.

URL

В виджете URL вы можете отобразить содержимое URL из внешнего источника.

Для настройки, выберите тип *URL*:

Add widget

Type	URL
Name	URL
Refresh interval	Default (No refresh)
* URL	<input type="text" value="http://"/>
Dynamic item	<input type="checkbox"/>

Вы можете указать следующие опции:

- URL** Введите отображаемый URL.
 URL должен начинаться с `http://`.
 Поддерживаются макросы `{HOST.*}`.
- Динамический элемент** Укажите отображение разного содержимого URL в зависимости от выбранного узла сети.
 Эта опция работает, если в URL используются макросы `{HOST.*}`.



Браузеры возможно не станут загружать HTTP страницу, которая включена в этот виджет, если веб-интерфейс Zabbix доступен через протокол HTTPS.

Веб-мониторинг

Этот виджет отображает сводку по активным сценариям веб-мониторинга.



В случае, когда у пользователя не хватает прав для доступа к отдельным элементам виджетов, тогда в процессе настройки виджета имя элемента будет отображаться как *Недоступно*. 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*.

18.1.1.2 Проблемы

Обзор

В разделе *Мониторинг → Проблемы* вы можете увидеть какие проблемы сейчас у вас имеются. Проблемами называются те триггеры, которые в состоянии “Проблема”.

Problems										Export to CSV	Filter
Time	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags	
2017-10-25 11:30:05	Average		PROBLEM		New host	Free disk space is less than 20% on volume /	8m 13d 3h	Yes	↑ ↻		
14:55:56	Information		PROBLEM		New host	New host has just been restarted	4m 7s	No			
Displaying 2 of 2 found											
0 selected	Mass update										

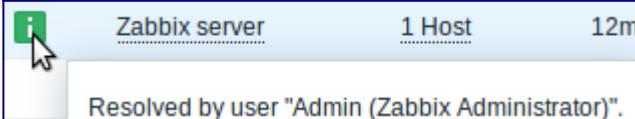
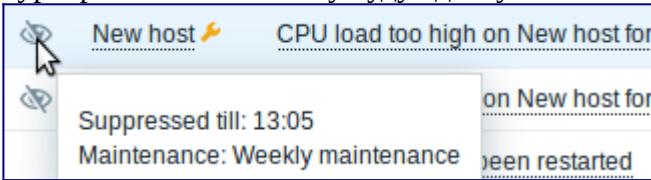
Колонка

Время Время начала проблемы.

Важность Важность проблемы.

Важность Важность проблемы изначально основывается на важности основного триггера проблемы, однако, после того как событие произошло важность проблемы можно обновить, используя [экран Обновление проблемы](#). Цвет важности проблемы

Описание

Колонка	Описание
Время восстановления	используется фоном ячейки в течении времени проблемы.
Состояние	<p>Время решения проблемы.</p> <p>Состояние проблемы:</p> <p>Проблема - нерешенная проблема</p> <p>Решено - недавно решенная проблема. Вы можете скрыть недавно решенные проблемы при помощи фильтра.</p> <p>Новые и недавно решенные проблемы мигают в течении 2 минуты. Решенные проблемы в целом отображаются в течении 5 минут. Оба этих значения настраиваются в Администрировании → Общие → Опции отображения триггеров.</p> <p>Зеленая информационная иконка, если проблема закрыта глобальной корреляцией или вручную при обновлении проблемы. При наведении курсора мыши на иконку будут доступны более детальные сведения:</p>  <p>Resolved by user "Admin (Zabbix Administrator)".</p>
Инфо	Следующая иконка появляется, если отображаются подавленные проблемы (смотрите в фильтре Отображение подавленных проблем опцию). При наведении курсора мыши на иконку будут доступны более детальные сведения:
	 <p>New host ⚡ CPU load too high on New host for</p> <p>Suppressed till: 13:05 on New host for</p> <p>Maintenance: Weekly maintenance been restarted</p>
Узел сети	Узел сети с проблемой.
Проблема	Имя проблемы.
Длительность	Имя проблемы основывается от имени основного триггера проблемы.
Подтверждение	<p>Длительность проблемы.</p> <p>Смотрите также: отрицательная длительность проблем</p> <p>Состояние подтверждения проблемы:</p> <p>Да - зеленый текст сигнализирует о том, что проблема подтверждена. Проблема считается подтверждённой, если все её события подтверждены.</p> <p>Нет - красная ссылка сигнализирует о неподтверждённых событиях.</p> <p>Если вы нажмёте на эту ссылку, вы попадёте на экран обновления проблемы, где с проблемой можно выполнить различные действия, включая комментирование и подтверждение проблемы.</p> <p>История действий над проблемой в виде символов иконок:</p> <ul style="list-style-type: none"> - добавлены комментарии. Также отображается количество комментариев. - важность проблемы увеличена (например, Информационный → Предупреждение) - важность проблемы уменьшена (например, Предупреждение → Информационный) - важность проблемы изменена, но затем вернулась к изначальному уровню (например, Предупреждение → Информационный → Предупреждение) - выполнены действия. Также отображается количество действий. - выполнены действия, по крайней мере одно в процессе выполнения. Также отображается количество действий. - выполнены действия, по крайней мере одно завершилось с ошибкой. Также отображается количество действий.
Действия	

Колонка

Описание

При наведении курсора мыши на иконки, отобразится всплывающее окно с детальной информацией о действиях. Смотрите [просмотр деталей](#) для объяснений используемых иконках во всплывающем меню по предпринятым действиям.

Теги

[Теги события](#) (если имеются).

Отрицательная длительность проблемы

Фактически имеется вероятность в некоторых распространенных ситуациях наличие отрицательной длительности проблемы то есть, когда время решения наступает раньше времени создания проблемы, например:

- Если какой-либо узел сети наблюдается прокси и происходит сетевая ошибка, которая приводит к тому, что данные от прокси не поступают некоторое время, тогда триггер item.nodata() сработает на стороне сервера. Когда соединение восстановится, сервер получит данные по элементу данных от прокси, у этих данных будет время из прошлого. Тогда проблема item.nodata() будет решена и у нее будет отрицательная длительность проблемы;
- Когда данные элемента данных, которые решают событие проблемы поступили от Zabbix sender и содержат штамп времени более старый чем время создания проблемы, в этом случае также отобразится отрицательная длительность проблемы.

Опции массового редактирования

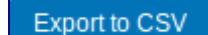
Кнопки ниже списка предлагают некоторые опции массового редактирования:

- *Массовое обновление* - обновление выбранных проблем с переходом к экрану [обновление проблемы](#)

Для использования этой опции, отметьте соответствующие проблемы, затем нажмите на кнопку *Массовое обновление*.

Кнопки

Кнопки справа предоставляют следующие опции:

 Export to CSV

Экспорт содержимого страницы в CSV.



Отображение страницы в полноэкранном режиме.

Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.

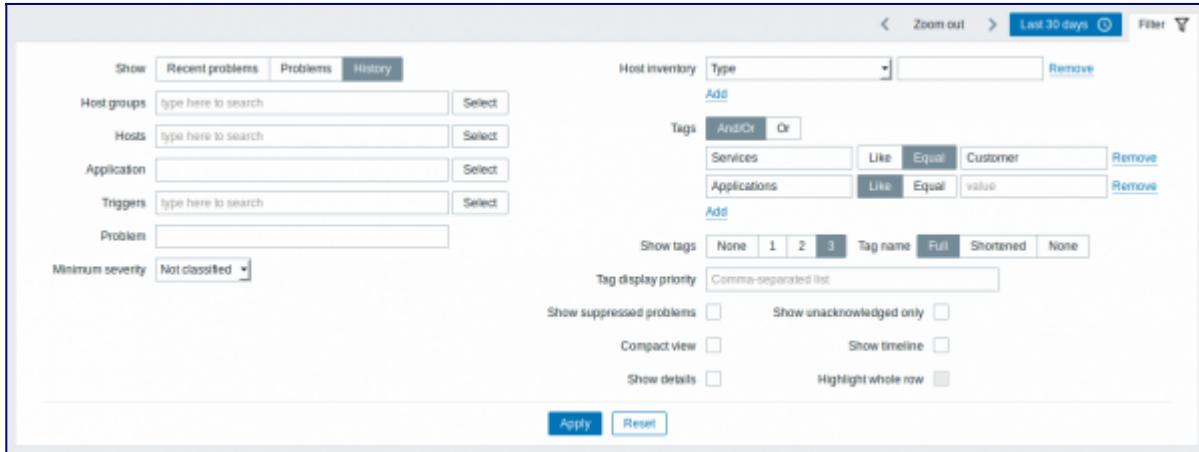


Кнопка режима киоска появляется после активации полноэкранного режима.

Чтобы выйти из режима киоска, переместите курсор до появления  кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

Использование фильтра

Вы можете воспользоваться фильтром, чтобы отображать только те проблемы в которых вы заинтересованы. Фильтр располагается над таблицей.



Параметр

Показать

Группы узлов сети

Узлы сети

Группа элементов данных

Триггеры

Проблема

Минимальная важность

Инвентарные данные узла сети

Теги

Отображать теги

Имя тега

Описание

Фильтр по состоянию проблем:

Недавние проблемы - нерешенные и недавно решенные проблемы (по умолчанию)

Проблемы - нерешенные проблемы

История - история всех событий

Фильтр по одному или нескольким группам узлов сети.

Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.

Фильтр по одному или нескольким узлам сети.

Фильтр по имени группы элементов данных.

Фильтр по одному или нескольким триггерам.

Фильтр по имени проблемы.

Фильтр по минимальной важности триггеров (проблем).

Фильтр по типу инвентарных данных и значению.

Фильтр по имени тега событий и значению тега событий.

Можно задать несколько условий. Имеется два типа вычислений для нескольких условий:

И/Или - все условия совпадают, условия у которых имеется одинаковое имя тега будут сгруппированы по условию Или

Или - достаточно совпадения одного условия

Имеется два способа соответствия значения тегов:

Содержит - регистрозависимое соответствие подстроке

Равно - регистрозависимое соответствие строке

При фильтрации теги указанные здесь будут сначала отображены с проблемой, если только не предопределены списком *Приоритет отображения тегов* (см. ниже).

Выбор количества отображаемых тегов:

Нет - колонка *Теги* отсутствует в *Мониторинг → Проблемы*

1- колонка *Теги* содержит один тег

2- колонка *Теги* содержит два тега

3- колонка *Теги* содержит три тега

Чтобы увидеть все теги по проблеме наведите курсор мыши на иконку с тремя точками.

Выберите режим отображения имени тега:

Полный - имена тегов и их значения отображаются в полном формате

Сокращённое - имена тегов сокращены до 3 символов; значения тегов отображаются в полном формате

Параметр

Описание

Приоритет отображения тегов

Нет - отображаются только значения тегов; без имён

Введите приоритет отображения тегов для проблемы, в виде разделённого запятыми списка тегов (например: Сервисы, Приложения, Приложение). Необходимо использовать только имена тегов, не значения. Теги из этого списка всегда отображаются первыми, переопределяя естественную сортировку по алфавиту.

Отображение подавленных проблем
Компактный вид

Отметьте, чтобы отображались проблемы, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.
Активация компактного вида.

Показывать детали

Отметьте, чтобы по проблемам отображались выражения основных триггеров. Деактивировано, если выбран параметр *Компактный вид*.

Отображать только неподтверждённые

Отметьте, чтобы отображались только неподтверждённые проблемы.

Отображать выбор времени

Отметьте, чтобы отображались визуальный выбор времени и группировка. Деактивировано, если выбран параметр *Компактный вид*.

Подсвечивать всю строку

Отметьте, чтобы по нерешенным проблемам подсвечивалась вся строка. Для подсветки используется цвет важности проблемы.

Активно только, если выбран параметр *Компактный вид* в стандартных синей и темной темах. *Подсвечивать всю строку* недоступен в высококонтрастных темах.

Просмотр деталей

В Мониторинг → Проблемы поля времени начала проблемы и восстановления являются ссылками. При нажатии на них, откроется более детальная информация по событию.

The screenshot shows the 'Event details' page in Zabbix. It displays two main sections: 'Trigger details' and 'Actions'.

Trigger details:

Host	New host
Trigger	Free disk space is less than 20% on volume /
Severity	Warning
Problem expression	<code>{New host}vfs[fs.size[pfree],last()<20}</code>
Recovery expression	
Event generation	Normal
Allow manual close	No
Enabled	Yes

Actions:

Step	Time	User/Recipient	Action	Message/Command	Status	Info
1	2018-07-05 14:52:52	Admin (Zabbix Administrator)	↑			
	2018-07-05 11:46:31	Admin (Zabbix Administrator)	↑			
	2018-07-05 11:45:15	Admin (Zabbix Administrator)	↓			
	2018-07-04 08:59:05	Admin (Zabbix Administrator)	↑			
	2018-06-26 08:01:14	Admin (Zabbix Administrator)	✓			
1	2017-10-25 11:30:09	Admin (Zabbix Administrator) Martins.Valkovskis@zabbix.com	✉	PROBLEM: Free disk space is less than 20% on volume / Trigger: Free disk space is less than 20% on volume / Trigger status: PROBLEM Trigger severity: Warning Trigger URL: Item values: 1. Free disk space on / (percentage) (My hostvfs[fs.size[pfree]]): 2.67 % 2. "UNKNOWN" ("UNKNOWN":"UNKNOWN"); "UNKNOWN" 3. "UNKNOWN" ("UNKNOWN":"UNKNOWN"); "UNKNOWN"	Sent	

Event details:

Event	Free disk space is less than 20% on volume /
Severity	Average
Time	2017-10-25 11:30:05
Acknowledged	Yes
Tags	

Event list [previous 20]:

Time	Recovery time	Status	Age	Duration	Ack	Actions
2017-10-25 11:30:05		PROBLEM	8m 13d 3h	8m 13d 3h	Yes	↑ ↻

Обратите внимание на то, как важность проблемы отличается у триггера и события о проблеме - у события о проблеме цвет был обновлен с использованием [экрана Обновление проблемы](#).

В списке действий для обозначения типов действий используются следующие иконки:

- - сгенерировано событие о проблеме
- - сообщение отправлено
- - событие о проблеме подтверждено
- - добавлен комментарий
- - увеличена важность проблемы (например, Информационный → Предупреждение)
- - уменьшена важность проблемы (например, Предупреждение → Информационный)
- - важность проблемы изменилась, но затем вернулась к изначальному уровню (например, Предупреждение → Информационный → Предупреждение)
- - выполнена удаленная команда
- - восстановлено событие о проблеме
- - проблема закрыта вручную

18.1.1.3 Обзор

Обзор

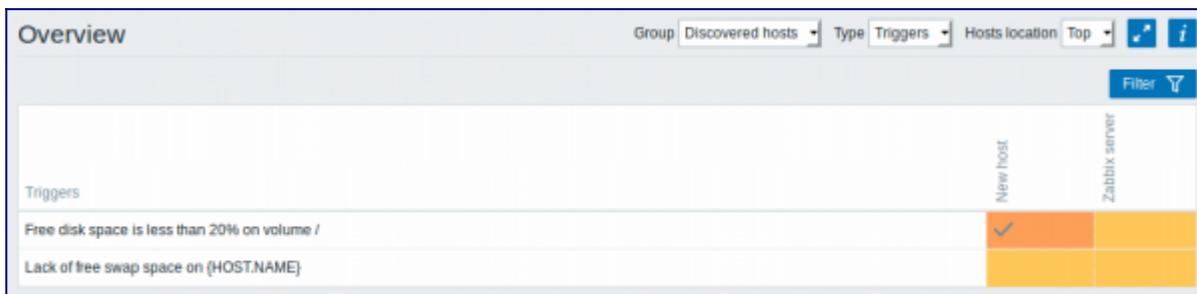
Раздел Мониторинг → Обзор предлагает обзор состояний триггеров, а также возможность сравнения данных с различных узлов сети за раз.

Доступны следующие опции просмотра:

- выбор всех узлов сети или конкретных групп узлов сети в выпадающем меню *Группа*
- выбор отображаемой информации (триггеры или данные) в выпадающем меню *Тип*
- выбор горизонтального или вертикального отображения информации в выпадающем меню *Расположение узлов сети*

Обзор триггеров

На следующем снимке экрана выбрано Триггеры в выпадающем меню *Тип*. Как результат, цветными блоками (цвет триггеров с проблемами зависит от цвета важности проблемы, которую можно задать в экране [обновления проблемы](#)) отобразятся состояния триггеров двух локальных узлов сети:

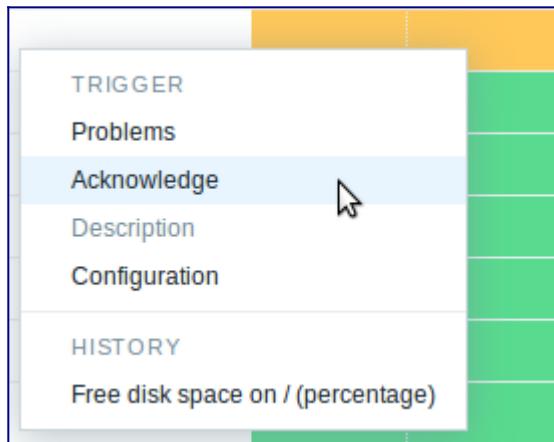


Обратите внимание, что недавно изменившие свои состояния триггера (за последние 2 минуты) будут отображаться мигающими блоками.

Синие стрелки вверх и вниз указывают на то, что триггеры имеют зависимости. При наведении указателя мыши отобразятся детали зависимости.

Иконка флагка указывает на подтверждённые проблемы.

При нажатии на блок триггеров появятся ссылки, которые зависят от контекста, к событиям о проблемах триггера, экрану подтверждения проблемы, настройке триггера, URL триггера или простому графику/списку последних значений.



Кнопки

Кнопки справа предоставляют следующие опции:



Отображение страницы в полноэкранном режиме.

Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.



Кнопка режима киоска появляется после активации полноэкранного режима.



Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).



Отображение дополнительной информации о содержимом на странице, если вы переместите курсор мыши на эту кнопку.

Использование фильтра

Вы можете воспользоваться фильтром, чтобы отображать только те проблемы в которых вы заинтересованы. Фильтр располагается над таблицей.

A screenshot of a filter interface. It includes dropdown menus for "Show" (Recent problems, Problems, Any), "Acknowledge status" (Any, Not classified), "Minimum severity" (Not classified), "Age less than" (14 days), "Application" (Select), "Host inventory Type" (Add, Remove), "Show suppressed problems" (checkbox), and a "Name" input field. At the bottom are "Apply" and "Reset" buttons.

Параметр

Показать

Состояние подтверждения

Описание

Фильтр по состоянию проблем:

Недавние проблемы - нерешенные и недавно решенные проблемы (по умолчанию)

Проблемы - нерешенные проблемы

История - история всех событий

Фильтр по состоянию подтверждения:

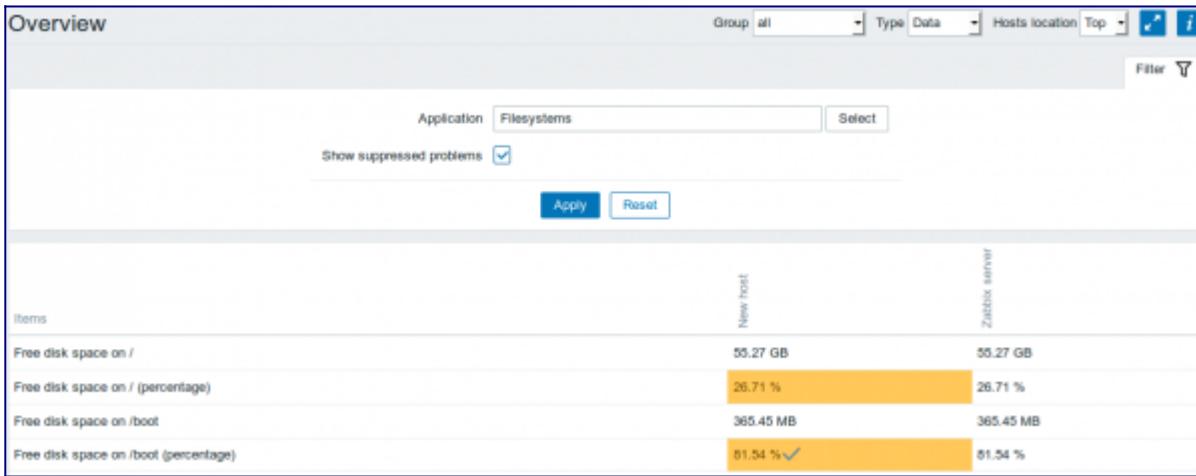
Любое - подтвержденные и неподтвержденные (по умолчанию)

С неподтвержденными событиями - проблемы с неподтвержденными событиями

С последним неподтвержденным событием - проблемы с последним событием, который не подтвержден

Параметр	Описание
Минимальная важность	Фильтр по минимальной важности триггеров (проблем).
Возраст (менее чем)	Фильтр по длительности проблем.
Имя	Фильтр по имени проблемы.
Группа элементов данных	Фильтр по группе элементов данных.
Инвентарные данные узла сети	Фильтр по типу инвентарных данных и значению.
Отображение подавленных проблем	Отметьте, чтобы отображались проблемы, которые в противном случае были бы подавлены (не показаны) по причине обслуживания узлов сети.
Обзор данных	

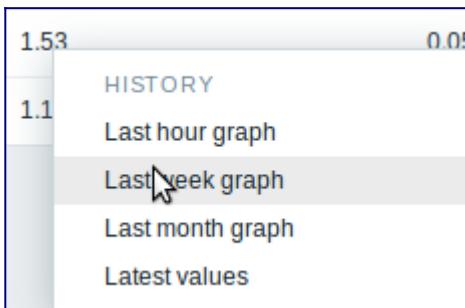
На следующем снимке экрана выбрано Данные в выпадающем меню *Tip*. Как результат, отобразятся данные производительности элементов данных двух локальных узлов сети.



Цвет элементов данных с проблемами основывается на цвете важности проблемы, который можно настроить на экране [обновления проблемы](#).

По умолчанию отображаются только те значения, которые поступили в течении последних 24 часов. Это ограничение введено с целью улучшения времени изначальной загрузки данных на больших страницах. Также имеется возможность изменения этого ограничения, изменив значение [константы ZBX_HISTORY_PERIOD](#) в *include/defines.inc.php*.

При нажатии на блоке данных отобразятся ссылки на некоторые предопределенные графики или последние значения.



18.1.1.4 Веб

Обзор

В разделе Мониторинг → Веб отображается текущая информация о [веб-сценариях](#).

Web monitoring

Group Discovered hosts Host Zabbix server

Host	Name	Number of steps	Last check	Status
Zabbix server	Zabbix frontend	5	2018-07-25 11:11:20	OK

Displaying 1 of 1 found

Возьмите на заметку: Имя деактивированного узла сети отображается красным цветом (в выпадающем меню с узлами сети и в списке). Данные по деактивированным узлам сети доступны начиная с Zabbix 2.2.0.

По умолчанию отображаются только те значения, которые были собраны за последние 24 часа. Такое ограничение введено с целью улучшения скорости изначальной загрузки больших страниц веб-мониторинга. Также имеется возможность изменения этого ограничения, изменив значение ZBX_HISTORY_PERIOD [константы](#) в *include/defines.inc.php*.

Имя сценария является ссылкой к более подробной статистике по этому сценарию:



Кнопки

Кнопки справа предоставляют следующие опции:



Отображение страницы в полноэкранном режиме.

Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.

 Кнопка режима киоска появляется после активации полноэкранного режима.

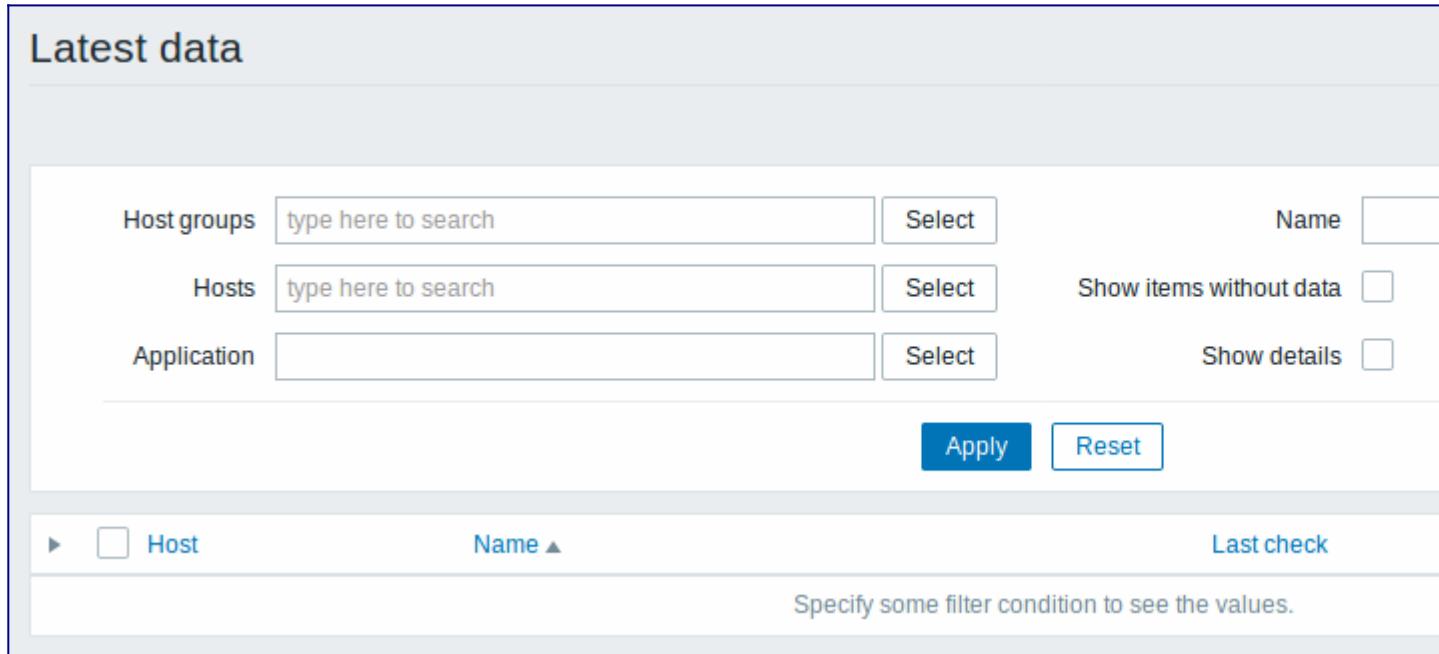
 Чтобы выйти из режима киоска, переместите курсор до появления  кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

18.1.1.5 Последние данные

Обзор

Раздел Мониторинг → Последние данные можно использовать для просмотреть последних значений, собранных элементами данных, а также для доступа к различным графикам по этим элементам данных.

Когда вы открываете эту страницу в первый раз, ничего не отобразится.



The screenshot shows the 'Latest data' page interface. At the top, there are three search fields: 'Host groups' (with placeholder 'type here to search'), 'Hosts' (with placeholder 'type here to search'), and 'Application' (empty). To the right of each field is a 'Select' button. Further to the right are two checkboxes: 'Name' (unchecked) and 'Show items without data' (unchecked). Below these are two more checkboxes: 'Show details' (unchecked) and 'Last check' (unchecked). At the bottom of this section are 'Apply' and 'Reset' buttons. Below this is a table header row with columns: '▶' (checkbox), 'Host' (checkbox), 'Name ▲' (sorted by Name), and 'Last check'. A note below the table says 'Specify some filter condition to see the values.'

Для доступа к данным вам необходимо сделать выбор в фильтре. Чтобы это сделать, нажмите на **Фильтр** и выберите группу узлов сети, узел сети, группу элементов данных или имя элемента данных.

The screenshot shows the Zabbix web interface with the following search parameters:

- Host groups:** type here to search [Select]
- Hosts:** My host × Zabbix server × [Select] type here to search
- Application:** Zabbix frontend [Select]
- Name:** [Search input]
- Show items without data:**
- Show details:**

Buttons at the bottom: **Apply** and **Reset**.

Host	Name	Last check	Last value	Change	Action
My host (18 items)					
	Download speed for scenario "Zabbix frontend".	2018-06-12 12:22:30	559.77 Kbps	+268.14 Kbps	Graph
	Download speed for step "First page" of scenario "Zabbix fronte...".	2018-06-12 12:22:30	28.64 Kbps	-8.21 Kbps	Graph
	Download speed for step "Log in" of scenario "Zabbix frontend".	2018-06-12 12:22:30	602.82 Kbps	+214.78 Kbps	Graph
	Download speed for step "Login check" of scenario "Zabbix fron...".	2018-06-12 12:22:30	1.02 MBps	+535.32 Kbps	Graph
	Download speed for step "Log out" of scenario "Zabbix frontend".	2018-06-12 12:22:30	265.56 Kbps	+159.42 Kbps	Graph
	Download speed for step "Logout check" of scenario "Zabbix fro...".	2018-06-12 12:22:30	854.24 Kbps	+439.38 Kbps	Graph
	Failed step of scenario "Zabbix frontend".	2018-06-12 12:22:30	0		Graph
	Last error message of scenario "Zabbix frontend".	2018-06-12 12:02:16	required pattern "Us..."		History
	Response code for step "First page" of scenario "Zabbix frontend".	2018-06-12 12:22:30	200		Graph
	Response code for step "Log in" of scenario "Zabbix frontend".	2018-06-12 12:22:30	200		Graph
	Response code for step "Login check" of scenario "Zabbix front...".	2018-06-12 12:22:30	200		Graph
	Response code for step "Log out" of scenario "Zabbix frontend".	2018-06-12 12:22:30	200		Graph
	Response code for step "Logout check" of scenario "Zabbix fron...".	2018-06-12 12:22:30	200		Graph
	Response time for step "First page" of scenario "Zabbix frontend".	2018-06-12 12:22:30	117.5ms	+20ms	Graph
	Response time for step "Log in" of scenario "Zabbix frontend".	2018-06-12 12:22:30	100ms	-50ms	Graph
	Response time for step "Login check" of scenario "Zabbix front...".	2018-06-12 12:22:30	57.5ms	-60ms	Graph
	Response time for step "Log out" of scenario "Zabbix frontend".	2018-06-12 12:22:30	57.9ms	-90ms	Graph
	Response time for step "Logout check" of scenario "Zabbix front...".	2018-06-12 12:22:30	70.6ms	-70ms	Graph
Zabbix server (18 items)					
Zabbix frontend (18 items)					

В отображаемом списке нажмите на '+' рядом с именем узла сети и соответствующей группой элементов данных, чтобы раскрыть последние значения этого узла сети и группы элементов данных. Вы можете раскрыть все узлы сети и все группы элементов данных, нажав на '+' в строке заголовка, таким образом, раскроются все элементы данных.

Обратите внимание: Имя деактивированного узла сети отображается красным цветом. Данные по деактивированным узлам сети, включая графики и списки значений элементов данных, доступны в *Последних данных* начиная с Zabbix 2.2.0.

Элементы данных отображаются вместе с их именем, временем последней проверки, последним значением, величиной изменения и ссылкой на простой график/историю значений элемента данных.

По умолчанию отображаются только значения, которые поступили в течении последних 24 часов. Это ограничение введено с целью улучшения времени изначальной загрузки данных на больших страницах. Также имеется возможность изменения этого ограничения, изменив значение [константы ZBX_HISTORY_PERIOD](#) в *include/defines.inc.php*.

Кнопки

Кнопки справа предоставляют следующие опции:



Отображение страницы в полноэкранном режиме.



Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.

Кнопка режима киоска появляется после активации полноэкранного режима.



Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в

полноэкранный режим).

Использование фильтра

Вы можете использовать фильтр, чтобы отображались только те элементы данных в которых вы заинтересованы. Ссылка на *Показать фильтр* расположена справа, выше таблицы. Вы можете использовать его, чтобы отфильтровать элементы данных по строке в имени; вы также можете выбрать отображение элементов данных, с которых еще не собирались данные.

Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.

Показывать детали позволяет расширить отображаемую информацию по выбранным элементам данных. Будет отображаться такая информация как интервал обновления, настройки истории и динамики изменений, тип элементов данных и ошибки элементов данных (в порядке/неподдерживаемые). Также доступны ссылки на настройки элементов данных.

The screenshot shows the Zabbix 'Latest data' screen. At the top, there are three search fields: 'Host groups' (empty), 'Hosts' containing 'My host' and 'Zabbix server', and 'Application' containing 'Zabbix frontend'. Below these are two checkboxes: 'Show items without data' (checked) and 'Show details' (unchecked). There are 'Apply' and 'Reset' buttons. The main area is a table with columns: Host, Name, Last check, Last value, and Change. It lists two hosts: 'My host' (with one item 'Zabbix frontend') and 'Zabbix server' (with one item 'Download speed for scenario "Zabbix frontend"'). For 'My host', the last check was 2018-06-12 12:38:38, last value was 516.67 KBps, and change was -3.12 KBps. For 'Zabbix server', the last check was 2018-06-12 12:38:56, last value was 638.86 KBps, and change was +265.55 KBps. At the bottom, there are buttons for 'Display stacked graph' and 'Display graph'.

Host	Name	Last check	Last value	Change
My host	Zabbix frontend (1 item)			
My host	Download speed for scenario "Zabbix frontend".	2018-06-12 12:38:38	516.67 KBps	-3.12 KBps
Zabbix server	Zabbix frontend (1 item)			
Zabbix server	Download speed for scenario "Zabbix frontend".	2018-06-12 12:38:56	638.86 KBps	+265.55 KBps

По умолчанию, элементы данных без истории и детали не отображаются.

Ситуационные графики для сравнения элементов данных

Вы можете использовать маркер из второй колонки, чтобы выбрать несколько элементов данных и затем сравнить эти данные на простом или стэкируемом [ситуационных графиках](#). Чтобы это сделать, выберите интересующие элементы данных, затем нажмите на требуемую кнопку графика.

Ссылки на историю значений/простой график

Последняя колонка в списке последних значений предлагает:

- ссылку на **История** (для всех текстовых элементов данных) - ведет к списку (Значения/500 последних значений), который отображает историю предыдущих значений элемента данных.
- ссылку на **График** (для всех числовых элементов данных) - ведет к [простому графику](#). Однако, как только график будет отображен, выпадающее меню справа сверху предложит возможность переключиться к Значениям/500 последних значений.

My host: Processor load (1 min average per core)	
	Processor load (1 min average per core)
Timestamp	
2018-06-12 12:03:39	1.575
2018-06-12 12:02:39	1.82
2018-06-12 12:01:39	1.635
2018-06-12 12:00:39	1.785
2018-06-12 11:59:39	1.69
2018-06-12 11:58:39	1.25
2018-06-12 11:57:39	0.73
2018-06-12 11:56:39	0.6
2018-06-12 11:55:39	0.525

Значения в списке отображаются как “сырые”, то есть, постобработка не применяется.

Общее количество отображаемых значений определяется значением параметра *Лимит элементов в результатах поиска и фильтрах*, указанному в [Администрирование -> Общие](#).

18.1.1.6 Графики

Обзор

В разделе *Мониторинг → Графики* отображаются все настроенные [пользовательские графики](#).



Для просмотра графика, выберите группу узлов сети, узел сети и затем график из выпадающего меню справа.

Обратите внимание: В выпадающем меню узлов сети имя деактивированного узла сети отображается красным цветом. Графики по деактивированным узлам сети доступны начиная с Zabbix 2.2.0.

Выбор периода времени

Обратите внимание на селектор периода времени выше графика. Он позволяет выбрать часто требуемые периоды за одно нажатие.

Смотрите также: [Селектор периода времени](#)

Кнопки

Кнопки справа предоставляют следующие опции:

-  Добавление графика в виджет избранного на [Панели](#).
-  Отображение страницы в полноэкранном режиме.
- Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.
-  Кнопка режима киоска появляется после активации полноэкранного режима.
- Чтобы выйти из режима киоска, переместите курсор до появления  кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

18.1.1.7 Комплексные экраны

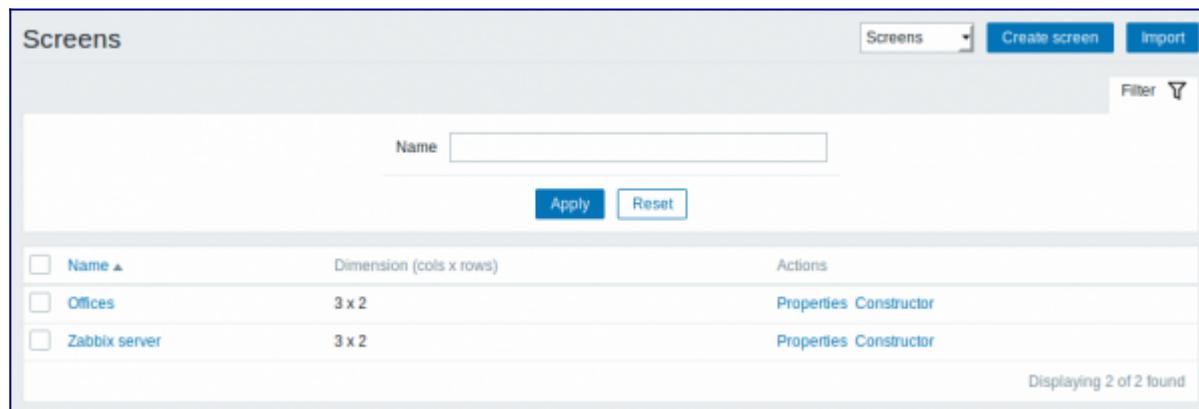
Обзор

В разделе *Мониторинг → Комплексные экраны* можно просмотреть все настраивать, управлять и просматривать глобальные [комплексные экраны](#) и [слайд-шоу](#).

Когда вы откроете этот раздел, вы увидите либо последний комплексный экран/слайд-шоу, который открывали ранее, или список всех объектов к которым у вас имеется доступ. Начиная с Zabbix 3.0, комплексные экраны/слайд-шоу могут быть как публичными, так и приватными. Публичные доступны всем пользователям, тогда как, приватные доступны только их владельцам и пользователям, с которыми поделились этим объектом.

Используйте выпадающее меню в строке заголовка для переключения между комплексными экранами и слайд-шоу.

Список комплексных экранов



The screenshot shows the 'Screens' list interface. At the top, there is a search bar labeled 'Name' with a placeholder 'Name' and two buttons: 'Apply' and 'Reset'. Below the search bar is a table with three columns: 'Name', 'Dimension (cols x rows)', and 'Actions'. There are two entries in the table:

Name	Dimension (cols x rows)	Actions
Offices	3 x 2	Properties Constructor
Zabbix server	3 x 2	Properties Constructor

At the bottom right of the table, it says 'Displaying 2 of 2 found'.

Отображаемые данные:

Колонка

Описание

Имя Имя комплексного экрана. Нажмите на имя, чтобы [увидеть](#) комплексный экран.

Размеры Количество колонок и строк комплексного экрана.

Доступны два действия:

Действия Свойства - изменение общих [свойств](#) комплексного экрана (имя и размеры)

Конструктор - доступ к сетке для добавления [элементов](#) комплексного экрана

Чтобы [настроить](#) новый комплексный экран, нажмите на кнопку *Создать комплексный экран* в верхнем правом углу. Чтобы импортировать комплексный экран из XML файла, нажмите на кнопку *Импорт* в верхнем правом углу. Пользователь, который импортирует комплексный экран, будет установлен владельцем этого комплексного экрана.

Опции массового редактирования

Две кнопки ниже списка предлагают некоторые опции массового редактирования:

- Экспорт - экспорт комплексных экранов в XML файл
- Удалить - удаление комплексных экранов

Чтобы использовать эти опции, отметьте соответствующие комплексные экраны, затем нажмите на требуемую кнопку.

Просмотр комплексных экранов

Для просмотра комплексного экрана нажмите на его имя в списке комплексных экранов.



Выбор периода времени

Обратите внимание на селектор периода времени выше комплексного экрана. Он позволяет выбрать часто требуемые периоды за одно нажатие, влияя на отображаемые данные на графиках и т.п.

Смотрите также: [Селектор периода времени](#)

Кнопки

Кнопки справа предоставляют следующие опции:

Edit screen

Переход в конструктор комплексного экрана для изменения комплексного экрана.



Добавление комплексного экрана в виджет избранного на [Панели](#).



Отображение страницы в полноэкранном режиме.



Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.

Кнопка режима киоска появляется после активации полноэкранного режима.

Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

Список слайд-шоу

Используйте выпадающее меню в строке заголовка для переключения с комплексных экранов на слайд-шоу.

Name	Delay	Number of slides	Actions
Zabbix	30s	2	Properties

Displaying 1 of 1 found

Отображаемые данные:

Колонка

Описание

Имя Имя слайд-шоу. Нажмите на имя, чтобы [увидеть](#) слайд-шоу.

Задержка Длительность отображения одного слайда по умолчанию.

Количество слайдов Количество отображаемых слайдов в слайд-шоу.

Действия Доступно одно действие:

Свойства - изменение [свойств](#) слайд-шоу

Чтобы [настроить](#) новое слайд-шоу, нажмите на кнопку *Создать слайд-шоу* в верхнем правом углу.

Опции массового редактирования

Кнопка ниже списка предлагает одну опцию массового редактирования:

- Удалить - удаление слайд-шоу

Чтобы использовать эту опцию, отметьте соответствующие слайд-шоу, затем нажмите на Удалить.

Просмотр слайд-шоу

Для просмотра слайд-шоу нажмите на его имя в списке всех слайд-шоу.

Кнопки

Кнопки справа предоставляют следующие опции:

Edit slide show

Переход в свойства слайд-шоу.



Добавление слайд-шоу в виджет избранного на [Панели](#).



Отображение страницы в полноэкранном режиме.



Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.

Кнопка режима киоска появляется после активации полноэкранного режима.

Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

Ссылки на комплексный экран

На комплексные экраны можно ссылаться параметрами GET как по `elementid`, так и по `screenname`. Например,

<http://zabbix/zabbix/screens.php?screenname=Zabbix%20server>

откроется комплексный экран с именем (Zabbix server).

Если указаны и `elementid` (ID комплексного экрана), и `screenname` (имя комплексного экрана), `screenname` имеет более высокий приоритет.

18.1.1.8 Карты сети

Обзор

В разделе *Мониторинг → Карты сети* можно просмотреть все настроенные [карты сети](#).

Когда вы откроете этот раздел, вы увидите либо последнюю карту сети, которую открывали ранее, или список всех карт сети к которым у вас имеется доступ. Начиная с Zabbix 3.0, карты могут быть как публичными, так и приватными. Публичные карты доступны всем пользователям, тогда как, приватные карты доступны только их владельцам и пользователям, с которыми поделились этой картой сети.

Список карт сети

Name	Width	Height	Actions
Local network	680	600	Properties Constructor
Local network_2	600	450	Properties Constructor
Offices	750	600	Properties Constructor

Отображаемые данные:

Колонка	Описание
Имя	Имя карты сети. Нажмите на имя карты, чтобы увидеть карту сети.
Ширина	Ширина карты сети.
Высота	Высота карты сети.
Действия	Доступны два действия:

Колонка

Описание

Свойства - изменение общих [свойств](#) карты

Конструктор - доступ к сетке для добавления [элементов карты сети](#)

Чтобы [настроить](#) новую карту сети, нажмите на кнопку *Создать карту сети* в верхнем правом углу. Чтобы импортировать карту сети из XML файла, нажмите на кнопку *Импорт* в верхнем правом углу. Пользователь, который импортирует карту сети, будет установлен владельцем этой карты сети.

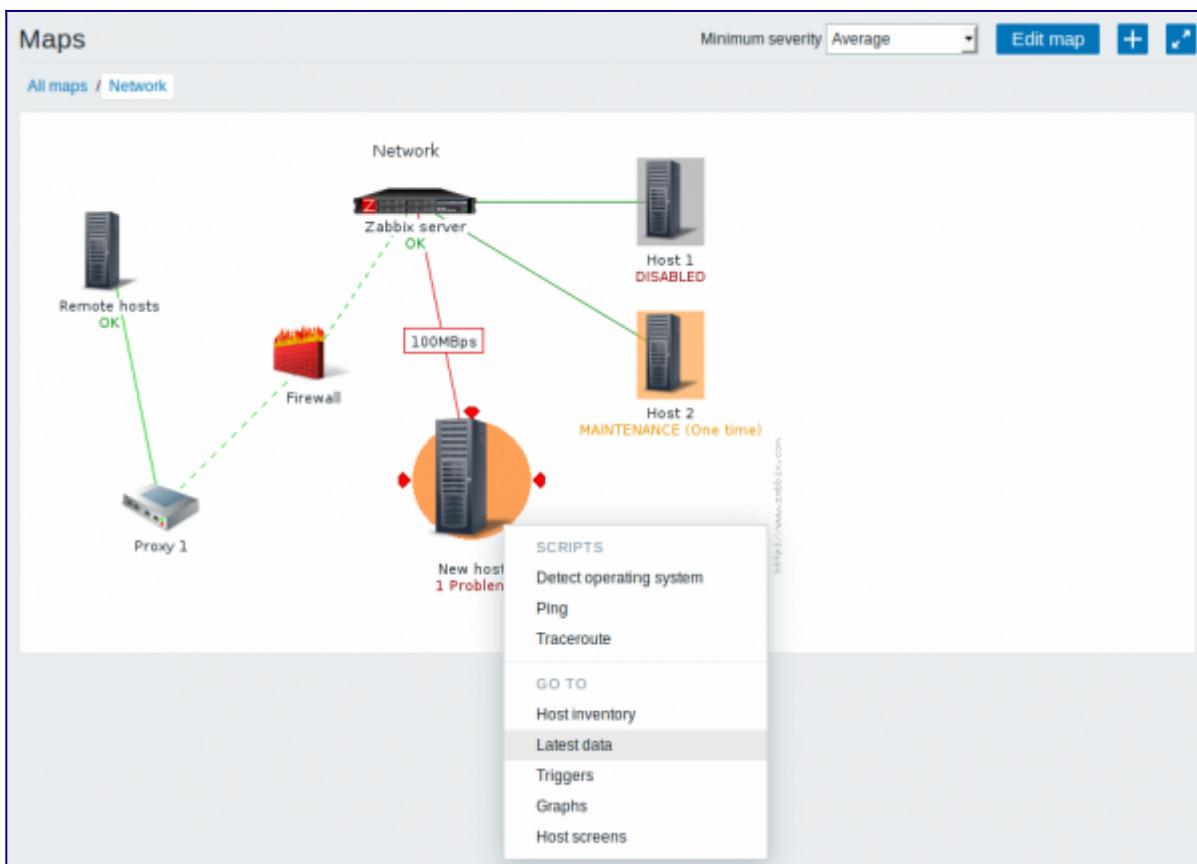
Две кнопки ниже списка предлагают некоторые опции массового редактирования:

- **Экспорт** - экспорт карты сети в XML файл
- **Удалить** - удаление карты сети

Чтобы использовать эти опции, отметьте соответствующие карты сети, затем нажмите на требуемую кнопку.

Просмотр карты сети

Для просмотра карты сети нажмите на его имя в списке всех карт.



Вы можете использовать выпадающий список в панели заголовка карт сетей для выбора наименьшего уровня важности отображаемых триггеров. Важность отмеченная как *по умолчанию* является уровнем указанным при настройке карты сети. Если карта содержит подкарту, перемещение к подкарте сохранит важность указанную в карте сети более высокого уровня.

Подсветка иконок

Если элемент карты находится в состоянии проблемы, он будет подсвечиваться окружностью. Цвет заливки окружности соответствует цвету важности проблемы. Только проблемы равные или выше согласно выбранной важности будут отображаться на элементах. Если все проблемы подтверждены, то вокруг окружности отображается зеленая рамка.

Кроме того:

- узел сети в состоянии [обслуживания](#) будет подсвечиваться заполненным оранжевым квадратом. Обратите внимание, что подсветка обслуживания имеет больший приоритет над подсветкой важности проблем.
- деактивированный (не под наблюдением) узел сети подсвечивается заполненным серым квадратом.

Подсветка отображается, если выбрана опция *Подсветка иконок* в [настройках](#) карты сети.

Маркеры недавнего изменения

Направленные внутрь красные треугольники вокруг элемента указывают на недавнее изменение состояния триггера - что произошло в последние 30 минут. Эти треугольники отображаются, если выбрана опция *Помечать элементы при изменении состояния триггера* в [настройках](#) карты сети.

Связи

Нажав на элементе карты сети, будет открыто меню с доступными связями.

Кнопки

Кнопки справа предоставляют следующие опции:

Edit map

Переход в конструктор карты для редактирования содержимого карты сети.



Добавление карты в виджет избранного на [Панели](#).



Отображение страницы в полноэкранном режиме.

Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.



Кнопка режима киоска появляется после активации полноэкранного режима.



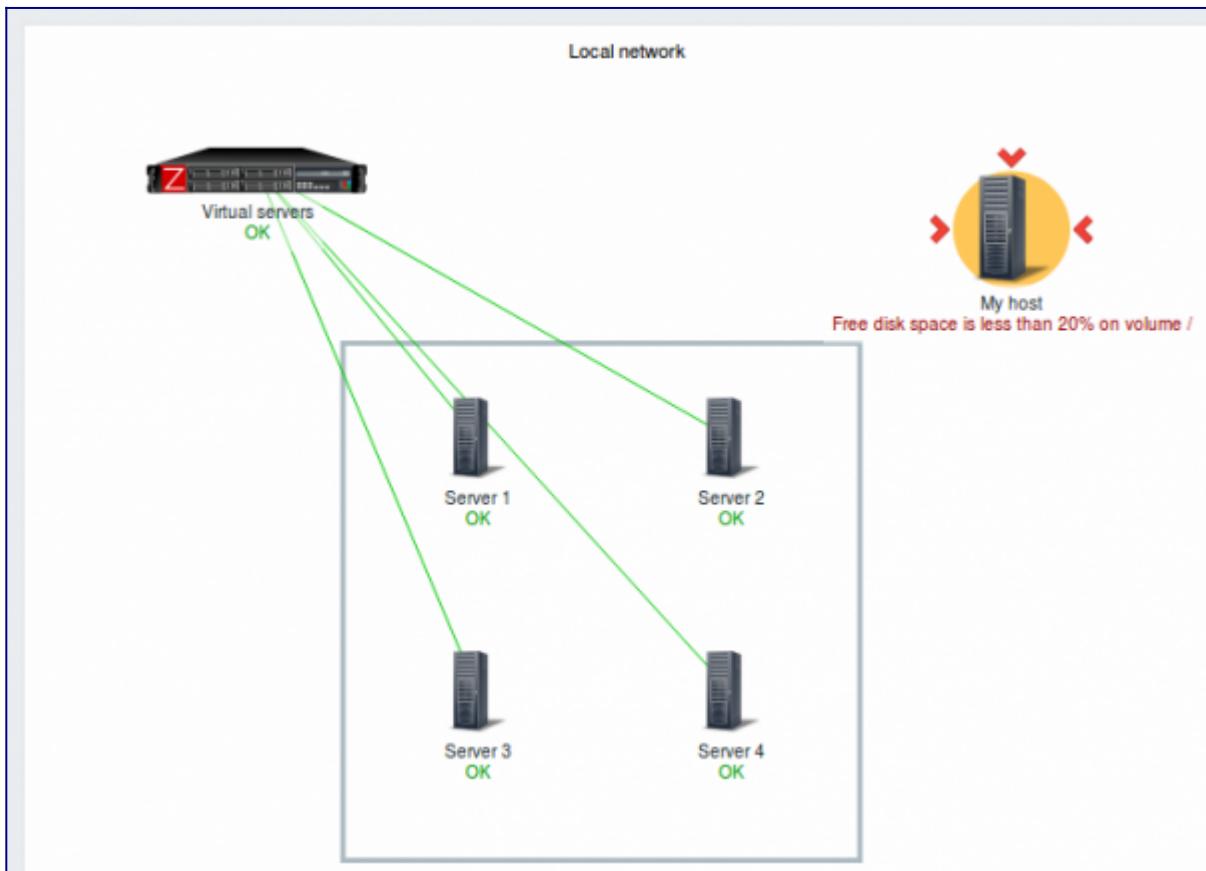
Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

Удобочитаемая сводка на картах

К картам сети добавлено скрытое свойство “aria-label”, которое позволяет считывать информацию о карте при помощи устройств чтения с экрана. Доступно как общее описание карты сети, так и описания по отдельным элементам, в следующем формате:

- для описания карты сети: <Имя карты>, <* из * элементов в состоянии проблемы>, <* проблем всего>.
- для описания одного элемента с одной проблемой: <Тип элемента>, Состояние <Состояние элемента>, <Имя элемента>, <Описание проблемы>.
- для описания одного элемента с несколькими проблемами: <Тип элемента>, Состояние <Состояние элемента>, <Имя элемента>, <* проблем>.
- для описания одного элемента без проблем: <Тип элемента>, Состояние <Состояние элемента>, <Имя элемента>.

Например, такое описание доступно:



Ссылки на карту сети

На карты сети можно ссылаться параметрами GET как по `sysmapid`, так и по `mapname`. Например,
`http://zabbix/zabbix/maps.php?mapname=Local%20network`

откроется карта сети с этим именем (Local network).

Если указаны и `sysmapid` (ID карты сети) и `mapname` (имя карты), `mapname` имеет более высокий приоритет.

18.1.1.9 Обнаружение

Обзор

В разделе *Мониторинг → Обнаружение* отображаются результаты [сетевого обнаружения](#). Обнаруженные устройства отсортированы по правилу обнаружения.

Status of discovery						
Discovered device	Monitored host	Uptime/Downtime	HTTP	HTTPS	HTTPS	HTTPS
Discovery by server/proxy (12 devices)						
192.168.3.6 (laserjet.zabbix.lan)	laserjet.zabbix.lan	44 days, 23:07:36	1m 14d 23h	8d 12h 40m		
192.168.3.7 (procurve.zabbix.lan)	procurve.zabbix.lan	9 days, 02:40:09	9d 1h 40m			
192.168.3.9	192.168.3.9	44 days, 23:07:36	1m 14d 23h	8d 21h 40m		

Если это устройство уже наблюдалось, его имя узла сети будет значиться в колонке *Наблюдаемый узел сети*, и длительность с момента обнаружения или с момента потери после предыдущего обнаружения будут отображаться в колонке *Доступен/Недоступен*.

После этих колонок следующие отображают отдельные состояния сервисов каждого обнаруженного устройства (зеленые ячейки отображают сервисы, которые доступны, красные ячейки - сервисы, которые недоступны). Время доступности или время недоступности включены в саму ячейку.



Только те сервисы, которые найдены по крайней мере на одном устройстве, будут иметь колонку, которая отобразит состояние этих сервисов.

Кнопки

Кнопки справа предоставляют следующие опции:



Отображение страницы в полноэкранном режиме.

Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.



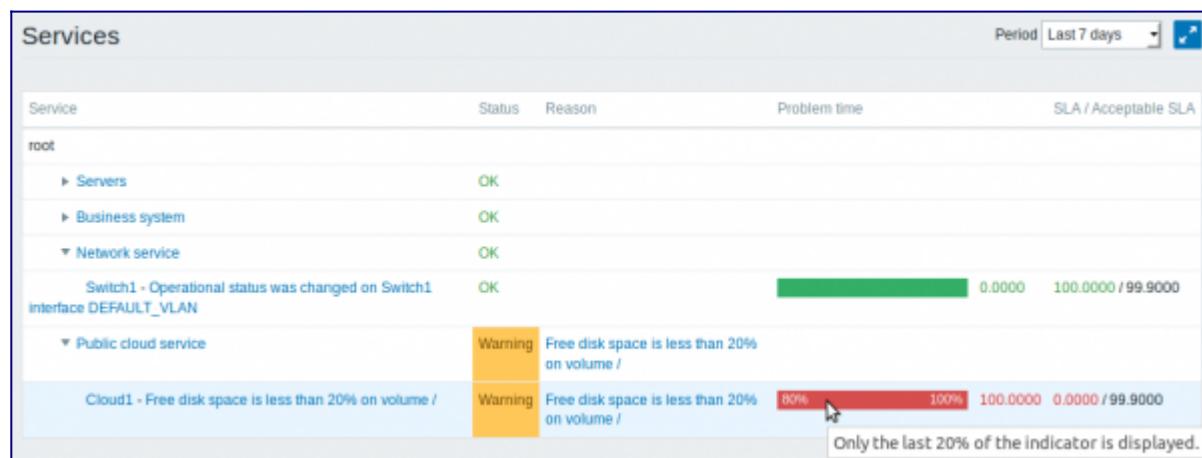
Кнопка режима киоска появляется после активации полноэкранного режима.

Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

18.1.1.10 Услуги

Обзор

В разделе Мониторинг → Услуги отображается состояние [услуг](#) ИТ инфраструктуры или бизнеса.



Список существующих услуг отображается вместе с данными о их состоянии и SLA. Из выпадающего списка в правом верхнем углу вы можете выбрать желаемый период отображения.

Отображаемые данные:

	Параметр	Описание
Услуга	Имя услуги.	
Состояние	Состояние услуги: OK - проблем нет (цвет и важность триггера) - указывает на наличие проблемы и её важность	
Причина	Указывается причина проблемы (если таковая имеется).	
Продолжительность проблем	Отображается полоса SLA. Соотношение зеленого/красного указывает на пропорцию доступности/проблем. Полоса отображает последние 20% SLA (от 80% до 100%).	
SLA/Допустимый уровень SLA	Полоса содержит ссылку на график доступности данных. Отображается текущее значение SLA/ожидаемое значение SLA. Если текущее значение ниже допустимого уровня, то значение отображается	

Параметр

Описание

красным цветом.

Вы также можете нажать на имя услуги, чтобы получить доступ к *Отчет о доступности услуг*.

Service availability report: Switch1						
From	Till	Ok	Problems	Downtime	SLA	Acceptable SLA
2017-02-20 00:00	2017-02-21 12:13	1d 12h 13m			100.0000	99.9000
2017-02-13 00:00	2017-02-20 00:00	7d 0h 0m			100.0000	99.9000
2017-02-06 00:00	2017-02-13 00:00	7d 0h 0m			100.0000	99.9000

Здесь вы можете оценить доступность данных услуги за более длительный период времени на основе дня/недели/месяца/года.

Кнопки

Кнопки справа предоставляют следующие опции:



Отображение страницы в полноэкранном режиме.

Отображение страницы в режиме киоска. В этом режиме отображается только содержимое страницы.



Кнопка режима киоска появляется после активации полноэкранного режима.



Чтобы выйти из режима киоска, переместите курсор до появления кнопки выхода и нажмите на неё. Обратите внимание, что вы вернётесь в нормальный режим (не в полноэкранный режим).

18.1.2 Инвентаризация

Обзор

Меню Инвентаризация включает разделы обеспечивающие обзор инвентарных данных узлов сети по выбранному параметру, а также, возможность просмотра деталей инвентаризации узлов сети.

18.1.2.1 Обзор

Обзор

В разделе *Инвентаризация* → *Обзор* имеется несколько способов просмотра обзорных сведений [инвентарных данных узлов сети](#).

В обзоре, который отобразится, выберите группу узлов сети (или все группы) и поле инвентарных данных по которому отобразятся данные. Будет отображено количество узлов сети соответствующих каждому выбранному полю.

Host inventory overview	
TYPE	HOST COUNT
Zabbix server	1
Workstation	1
Switch	1

Заполненность обзора зависит от того как много информации об инвентаре занесено в узел сети.

Количество в колонке *Количество узлов сети* является ссылкой; оно перенаправляет на эти узлы сети, отфильтрованные в таблице *Инвентарные данные узлов сети*.

Host inventory

Group all Filter ▾

Host	Group	Name	Type	OS	Serial number	Tag	MAC address
Zabbix server	Discovered hosts, Zabbix servers	martins-HP-Pro-3010-Small-Form-Factor-PC	Zabbix server	Linux martin-HP-Pro-3010-Small-Form-Factor-PC 4.4.0-135-generic			

Displaying 1 of 1 found

18.1.2.2 Узлы сети

Обзор

В разделе *Инвентаризация* → *Узлы сети* отображаются [инвентарные данные](#) узлов сети.

Выберите группу из выпадающего меню в верхнем правом углу и вы увидите инвентарные данные узлов сети из этой группы. Вы также можете отфильтровать узлы сети по любому из полей инвентаризации, чтобы увидеть только те узлы сети в которых вы заинтересованы.

Host inventory

Group all Filter ▾

Host	Group	Name	Type	OS	Serial number	Tag	MAC address
Zabbix server	Discovered hosts	martins-HP-Pro-3010-Small-Form-Factor-PC	Zabbix server	Linux martin-HP-Pro-3010-Small-Form-Factor-PC 4.4.0-128-generic			

Displaying 1 of 1 found

Для просмотра всех инвентарных данных узлов сети, выберите “Все” в выпадающем меню группы, очистите в фильтре поле сравнения и нажмите “Фильтр”.

В то время как в таблице отображаются только лишь несколько ключевых полей инвентаризации, вы можете также просмотреть все доступные инвентарные данные какого-либо узла сети. Чтобы это сделать, нажмите на имени узла сети в первой колонке.

Детали инвентаря

Вкладка **Обзор** содержит лишь общую информацию о узле сети вместе с ссылками на предустановленные скрипты, последние данные наблюдения и параметры настройки узлов сети:

Host inventory

Overview Details

Host name Zabbix server 1

Visible name Zabbix server

Agent interfaces	IP address	DNS name	Connect to	Port	Default
	192.168.3.220		IP DNS	10050	<input checked="" type="radio"/>

SNMP interfaces	IP address	DNS name	IP DNS	Port	Default
	127.0.0.1		IP DNS	161	<input checked="" type="radio"/>

OS Linux linux-qvvt 3.11.10-21-default #1 SMP Mon Jul 21 15:28:46 U

Description Added on 2015-07-28.

Monitoring Web Latest data Triggers Problems Graphs Screens

Configuration Host Applications 13 Items 81 Triggers 47 Graphs 12 Discovery 3 Web 1

[Cancel](#)

Вкладка **Детали** содержит все доступные детали инвентаря по выбранному узлу сети:

Overview Details

Type Zabbix server

Name linux-qvvt

OS Linux linux-qvvt 3.11.10-21-default #1 SMP Mon Jul 21 15:28:46 U

OS (Full details) Linux version 3.11.10-21-default (geeko@buildhost) (gcc version 4.8.1 20130909 [gcc-4_8-branch revision 202388] (SUSE Linux)) #1 SMP Mon Jul 21 15:28:46 UTC 2014 (9a9565d)

MAC address A [enp0s3] 08:00:27:62:c4:53, [enp0s3] 08:00:27:62:c4:53

Location Head Office

Site city Riga

[Cancel](#)

Заполненность инвентарных данных зависит от того как много информации об инвентаре занесено в узел сети. Если информация недоступна, вкладка *Детали* отключена.

18.1.3 Отчеты

Обзор

Меню Отчеты включает в себя несколько разделов, которые содержат различные предустановленные и пользовательские отчеты, направленные на обзор таких параметров как информации о системе, триггеров и собранных данных.

18.1.3.1 Информация о системе

Обзор

В разделе *Отчеты → Информация о системе* отображается краткая сводка о ключевых данных системы.

System information

Parameter	Value	Details
Zabbix server is running	Yes	localhost:10051
Number of hosts (enabled/disabled/templates)	92	12 / 0 / 80
Number of items (enabled/disabled/not supported)	349	330 / 0 / 19
Number of triggers (enabled/disabled [problem/ok])	211	208 / 3 [16 / 192]
Number of users (online)	8	1
Required server performance, new values per second	7.32	

Этот отчет также отображается виджетом на [ПАНЕЛИ](#).

Отображаемые данные

Параметр	Значение	Детали
<i>Zabbix сервер запущен</i>	<p>Состояние Zabbix сервера: Да - сервер запущен Нет - сервер не запущен</p> <p><i>Обратите внимание:</i> Чтобы отображать некоторую информацию, веб-интерфейсу необходим работающий сервер и у сервера должен быть запущен по крайней мере один процесс траппера (параметр StartTrappers в zabbix_server.conf файле > 0).</p> <p>Отображается общее количество добавленных узлов сети. Шаблоны тоже считаются подтипаами узлов сети.</p>	Размещение и порт Zabbix сервера.
<i>Количество узлов сети</i>	Отображается общее количество элементов данных	Количество наблюдаемых узлов сети/не наблюдаемых узлов сети/шаблонов.
<i>Количество элементов данных</i>	Отображается общее количество элементов данных.	Количество наблюдаемых/деактивированных/неподдерживаемых элементов данных. Элементы данных на деактивированных узлах сети считаются деактивированными. Количество активированных/деактивированных триггеров. [Триггеры в состоянии проблема/ок.]
<i>Количество триггеров</i>	Отображается общее количество триггеров.	Триггеры назначенные на деактивированные узлы сети или зависящие от деактивированных элементов данных считаются деактивированными.
<i>Количество пользователей</i>	Отображается общее количество добавленных пользователей.	Количество пользователей в сети.
<i>Требуемое быстродействие сервера, новые значения в секунду</i>	Отображается ожидаемое количество новых значений обрабатываемых Zabbix сервером в секунду.	Требуемое быстродействие сервера является оценочным и может быть полезным как ориентир. Для точных чисел обработанных значений, используйте внутренний элемент данных zabbix[wcache,values,all] .
		В вычисление включаются активированные элементы данных с узлов сети под

Параметр

Значение

Детали

наблюдением. Элементы данных журналов считаются как одно значение за период интервала обновления элемента данных. Подсчитываются обычные значений интервалов; значения переменных и интервалов по расписанию не учитываются. Вычисление не корректируется во время периода обслуживания “без данных”. Траппер элементы данных не подсчитываются.

18.1.3.2 Отчет о доступности

Overview

В разделе *Отчеты → Отчет о доступности* вы можете просмотреть какую часть времени каждый триггер был в состояниях проблема/ок/неизвестно. Для каждого состояния отображается процентное отношение к общему времени.

Таким образом, можно легко определить ситуации доступности различных элементов вашей системы.

Availability report

Mode: By host

Host group: all

Host: New host

Last 30 days

Filter

Apply Reset

Host	Name	Problems	Ok	Graph
New host	/etc/passwd has been changed on New host	0.4167%	99.5833%	Show
New host	Configured max number of opened files is too low on New host	100.0000%	100.0000%	Show
New host	Configured max number of processes is too low on New host	100.0000%	100.0000%	Show
New host	CPU load too high on New host for 3 minutes	2.9527%	97.0473%	Show
New host	Disk I/O is overloaded on New host	7.4999%	92.5001%	Show
New host	Free disk space is less than 20% on volume /	100.0000%	100.0000%	Show
New host	Free inodes is less than 20% on volume /	100.0000%	100.0000%	Show
New host	Host information was changed on New host	0.2778%	99.7222%	Show
New host	Host name of zabbix_agentd was changed on New host	100.0000%	100.0000%	Show

Из выпадающего списка в верхнем правом углу вы можете выбрать режим выборки - следует ли отображать триггеры по узлам сети или по триггерам, принадлежащим шаблону.

Availability report

Mode: By trigger template

Template group: all

Template: Template OS Linux

Template trigger: Disk I/O is overloaded on [HOST.NAME]

Host group: all

Last 30 days

Filter

Apply Reset

Host	Name	Problems	Ok	Graph
New host	Disk I/O is overloaded on New host	7.4999%	92.5001%	Show
Zabbix server	Disk I/O is overloaded on Zabbix server	7.6502%	92.3498%	Show

Displaying 2 of 2 found

Имя триггера является ссылкой на последние события этого триггера.

Использование фильтра

Вы можете использовать фильтр чтобы уменьшить количество выбранных записей. Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.

Фильтр расположен ниже надписи раздела *Отчет о доступности*. Его можно открыть и свернуть если нажать слева на вкладке *Фильтр*.

Селектор периода времени

Селектор периода времени позволяет выбирать часто требуемые периоды за одно нажатие мышкой. Селектор периода времени можно открыть если нажать на вкладку периода времени, которая расположена после фильтра.

При нажатии на *Показать* в колонке График, вы увидите столбчатый график, на котором отобразится информация о доступности в формате, когда каждый столбец представляет прошлую неделю текущего года.



Зеленая часть столбцов обозначает время в состоянии ОК, красная в состоянии проблем.

18.1.3.3 100 наиболее активных триггеров

Обзор

В разделе *Отчеты → 100 наиболее активных триггеров* вы можете просмотреть триггеры, которые изменили свои состояния в течении периода оценки, отсортированные по количеству изменений состояния.

100 busiest triggers

Host	Trigger	Severity	Number of status changes
New host	CPU load too high on New host for 3 minutes	Warning	92
Zabbix server	Disk I/O is overloaded on Zabbix server	Warning	88
New host	Disk I/O is overloaded on New host	Warning	82
New host	New host has just been restarted	Information	19
Zabbix server	Zabbix server has just been restarted	Information	19
Zabbix server	Lack of free swap space on Zabbix server	Warning	16
New host	Lack of free swap space on New host	Warning	12
New host	Zabbix agent on New host is unreachable for 5 minutes	Average	8
Zabbix server	Zabbix agent on Zabbix server is unreachable for 5 minutes	Average	8
New host	/etc/passwd has been changed on New host	Warning	4

Объекты в колонках узлов сети и триггеров являются ссылками, которые предлагают некоторые полезные функции:

- для узлов сети - ссылки на пользовательские скрипты, последние данные, инвентарь и комплексные экраны узла сети
- для триггеров - ссылки на последние события, диалог настройки триггера и простой график

Использование фильтра

Вы можете использовать фильтр чтобы уменьшить количество записей, используя фильтр по группе узлов сети, узлу сети или важности триггера. Указывая родительскую группу узлов сети косвенным образом будут выбраны все вложенные группы узлов сети.

Фильтр расположен ниже надписи раздела *100 наиболее активных триггеров*. Его можно открыть и свернуть если нажать слева на вкладке *Фильтр*.

Селектор периода времени

Селектор периода времени позволяет выбирать часто требуемые периоды за одно нажатие мышкой. Селектор периода времени можно открыть если нажать на вкладку периода времени, которая расположена после фильтра.

18.1.3.4 Аудит

Обзор

В разделе *Отчеты → Аудит* пользователи могут просмотреть записи о выполненных изменениях в веб-интерфейсе и детали этих действий.

Журналы

На этом экране можно просмотреть журналы аудита различных изменений сделанных из веб-интерфейса.

Audit log

The screenshot shows the 'Audit log' section of a web interface. At the top, there are filter controls: 'User' (with a 'Select' button), 'Action' (set to 'Update'), and 'Resource' (set to 'All'). Below these are 'Apply' and 'Reset' buttons. The main area is a table with the following columns: Time, User, IP, Resource, Action, ID, Description, and Details. The table contains the following data:

Time	User	IP	Resource	Action	ID	Description	Details
2018-06-12 12:48:16	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Cell changed coordinates "1,1" resource type "0"
2018-06-12 12:47:41	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Cell changed screen itemid "90" resource type "0"
2018-06-12 12:47:19	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Cell changed screen itemid "44" resource type "0"
2018-06-12 12:45:35	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Cell changed screen itemid "44" resource type "0"
2018-06-12 12:45:14	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Cell changed coordinates "0,1" resource type "2"
2018-06-12 12:44:56	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Cell changed coordinates "0,0" resource type "0"
2018-06-12 12:44:36	Admin	192.168.3.31	Screen	Updated	16	Zabbix server	Column added
2018-06-12 12:36:51	Admin	192.168.3.31	Host	Updated	10152	Zabbix server	hosts.status: 1 => 0

Отображаемые данные:

Колонка	Описание
Время	Время записи аудита.
Пользователь	Пользователь, выполнивший действия.
IP	IP адрес, с которого производились действия.
Ресурс	Измененный ресурс.
Действие	Тип активности - Вход, Выход, Добавлено, Обновлено, Удалено, Активировано или Деактивировано.
ID	ID измененного ресурса.
Описание	Описание ресурса.
Детали	Подробная информация о действиях над ресурсом.

Использование фильтра

Вы можете использовать фильтр чтобы уменьшить количество записей, используя фильтр по пользователю, типу активности и ресурсу, над которым производились действия.

Фильтр расположен ниже надписи раздела *Аудит*. Его можно открыть и свернуть если нажать слева на вкладке *Фильтр*.

Селектор периода времени

Селектор периода времени позволяет выбирать часто требуемые периоды за одно нажатие мышкой. Селектор периода времени можно открыть если нажать на вкладку периода времени, которая расположена после фильтра.

18.1.3.5 Журнал действий

Обзор

На этом экране имеется подробная информация о выполненных действиями операций (оповещения, удаленные команды).

Action log

The screenshot shows the Zabbix Action log interface. At the top, there are buttons for 'Recipient' (with a search bar), 'Select', 'Apply', and 'Reset'. Below is a table with columns: Time, Action, Type, Recipient, Message, Status, and Info.

Time	Action	Type	Recipient	Message	Status	Info
2018-06-12 11:23:41	Auto registration	Email	Admin (Zabbix Administrator) Martins.Valkovskis@zabbix.com	Subject: Auto registration: My host Message: Host name: My host Host IP: 192.168.3.31 Agent port: 10050 Remote proxy For testing.	Sent	
2018-06-12 11:23:41	Auto registration		user (New user)	Subject: Auto registration: My host Message: Host name: My host Host IP: 192.168.3.31 Agent port: 10050 Remote proxy For testing.	Failed	!

A 'Debug' button is located at the bottom right of the table area.

Отображаемые данные:

Колонка

Описание

Время	Время действия.
Действие	Имя действия, вызвавшего операции. Имя действия отображается начиная с Zabbix 2.4.0.
Тип	Тип действия - <i>Email</i> или <i>Команда</i> .
Получатель(и)	Псевдоним пользователя, его имя и фамилия (в круглых скобках) и e-mail адрес получателя оповещения. Имя пользователя, его имя и фамилия отображаются начиная с Zabbix 2.4.0.
Сообщение	Содержимое сообщения/удаленной команды. Удаленная команда отделяется от узла сети, который является целью, при помощи символа двоеточия: <узел сети>:<команда>. Если удаленная команда выполнялась на Zabbix сервер, тогда информация имеет следующий формат: Zabbix server:<команда>
Состояние	Состояние действия: <i>в процессе</i> - действие в процессе выполнения Для действий в состоянии в процессе отправки отображается оставшееся количество попыток - сколько раз сервер будет пытаться еще отправить оповещение. <i>отправлено</i> - оповещение было отправлено <i>выполнено</i> - команда была выполнена <i>не отправлено</i> - действие не завершилось
Инфо	Информация об ошибке (если имеется) при выполнении действия.

Использование фильтра

Вы можете использовать фильтр чтобы уменьшить количество записей, используя фильтр по e-mail получателя.

Фильтр расположен ниже надписи раздела *Журнал действий*. Его можно открыть и свернуть если нажать слева на вкладку *Фильтр*.

Селектор периода времени

[Селектор периода времени](#) позволяет выбирать часто требуемые периоды за одно нажатие мышкой. Селектор периода времени можно открыть если нажать на вкладку периода времени, которая расположена после фильтра.

18.1.3.6 Оповещения

Обзор

В разделе *Отчеты → Оповещения* имеется отчет по количеству отправленных оповещений по каждому пользователю.

Из выпадающего меню в верхнем правом углу вы можете выбрать способ оповещения (или все), период (данные за каждый день/неделю/месяц/год) и год, когда оповещения были отправлены.

DAY	ADMIN (ZABBIX ADMINISTRATOR)	GUEST	USER (NEW USER?)
2016-01-01			
2016-01-02	6 (6/0/0/0)		
2016-01-03	2 (2/0/0/0)		
2016-01-04	10 (10/0/0/0)		
2016-01-05	24 (24/0/0/0)		
2016-01-06	10 (10/0/0/0)		
2016-01-07	6 (6/0/0/0)		
2016-01-08	4 (4/0/0/0)		

Каждая колонка отображает суммарную информацию по каждому пользователю системы.

18.1.4 Настройка

Обзор

Меню Настройка содержит разделы посвященные настройке самых важных функций Zabbix, таких как узлы сети и группы узлов сети, сбор данных, пороги данных, отправка оповещений о проблемах, создание визуализации данных и многое другое.