

Мне понадобился простой мониторинг состояния транков в asterisk. Иногда после проблем с интернетом на некоторых серверах автоматически не восстанавливаются транки к sip провайдерам. Приходится дергать их вручную через sip reload после того, как пользователи начинают жаловаться. А так как провайдеров может быть несколько, не всегда сразу становится понятно, что какой-то из каналов отвалился.

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, рекомендую познакомиться с **онлайн-курсом «Администратор Linux»** в OTUS. Курс не для новичков, для поступления нужно пройти .

Полный мониторинг всего сервера asterisk я рассмотрел отдельно — настройка мониторинга asterisk. Там в том числе есть мониторинг транков, но более продвинутый и удобный, нежели здесь. Так что рекомендую ознакомиться с новой статьей.

Содержание:

- 1 Введение
- 2 Настройка агента для мониторинга транков (trunks)
- 3 Создание шаблона мониторинга asterisk на сервере zabbix
- 4 Возможные ошибки
- 5 Заключение
- 6 Дополнительные материалы по Zabbix

Введение

Я буду использовать очень простую проверку. В интернете находится много рецептов по **мониторингу asterisk** с помощью zabbix. Есть готовые наборы скриптов на питоне, есть шаблоны. Можно настроить мониторинг практически всего, что только пожелаешь.

Мне не хотелось во всем этом разбираться и нагромождать в систему, так как нужно **только состояние транков — зарегистрирован или нет**. Усложнять чем-то еще свои системы мониторинга не хотелось. Больше никакие данные мне не нужны. Я стараюсь настраивать мониторинг только тех параметров, которые реально необходимы. Это позволяет экономить время и ресурсы сервера.

С помощью простого sh скрипта я буду проверять суммарное количество транков в системе и сравнивать это число с числом зарегистрированных транков. Если разница этих чисел будет отлична от нуля, значит как минимум одна регистрация отвалилась. Нужно на всякий случай проверить сервер и выяснить причину.

После этой проверки можно тем же скриптом и перезапустить регистрации, но я в статье ограничусь только мониторингом состояния. Лично мне перезапуск в данном случае не нужен, так как ситуации бывают редко и я хочу сам проверить, почему регистрации отвалились.

Если у вас еще нет своего сервера для мониторинга, то рекомендую материалы на эту тему. Для тех, кто предпочитает систему CentOS:

1. Установка CentOS 7.
2. Настройка CentOS 7.
3. Установка и настройка zabbix сервера.

То же самое на Debian 9, если предпочитаете его:

1. Установка Debian 9.
2. Базовая настройка Debian 9.
3. Установка и настройка zabbix на debian.

Если у вас есть вопросы по настройке Asterisk, рекомендую мою очень подробную статью на эту тему. Там разобран на примерах основанной функционал современной ip атс.

Настройка агента для мониторинга транков (trunks)

Создадим папку для пользовательских скриптов в каталоге zabbix:

```
# mkdir /etc/zabbix/scripts
```

Создаем в ней скрипт asterisk.trunk.sh следующего содержания:

```
#!/bin/sh
# Получаем количество всех транков в системе
number_trunks=`/usr/sbin/asterisk -rx "sip show registry" | grep "SIP registrations" | awk '{print $1}'`
# Считаем количество зарегистрированных транков
reg_trunks=`/usr/sbin/asterisk -rx "sip show registry" | grep Registered | wc -l`
# Вычисляем разницу между полученными значениями
let result=$number_trunks-$reg_trunks
# Выводим результат вычисления
echo $result
```

Назначаем владельцем файла пользователя zabbix и выставляем права на выполнение скрипта:

```
# chown zabbix. /etc/zabbix/scripts/asterisk.trunk.sh
# chmod 0750 /etc/zabbix/scripts/asterisk.trunk.sh
```

Теперь просто запустите скрипт и посмотрите результат. Если все регистрации подключены, то скрипт должен вернуть значение 0:

```
# /etc/zabbix/scripts/asterisk.trunk.sh
0
```

Открываем на редактирование конфиг агента и добавляем параметр **UserParameter**:

```
UserParameter=asterisk.trunk,/etc/zabbix/scripts/asterisk.trunk.sh
```

Перезапускаем zabbix-agent той командой, что соответствует вашей системе. В CentOS 7 делаем так:

```
# systemctl restart zabbix-agent
```

Теперь проверим с помощью `zabbix_agentd` какое значение будет отправлять агент на сервер:

```
# zabbix_agentd -t asterisk.trunk  
asterisk.trunk [t|0]
```

Все в порядке, то что надо. На этом настройка агента на сервере asterisk закончена. Переходим на сервер мониторинга.

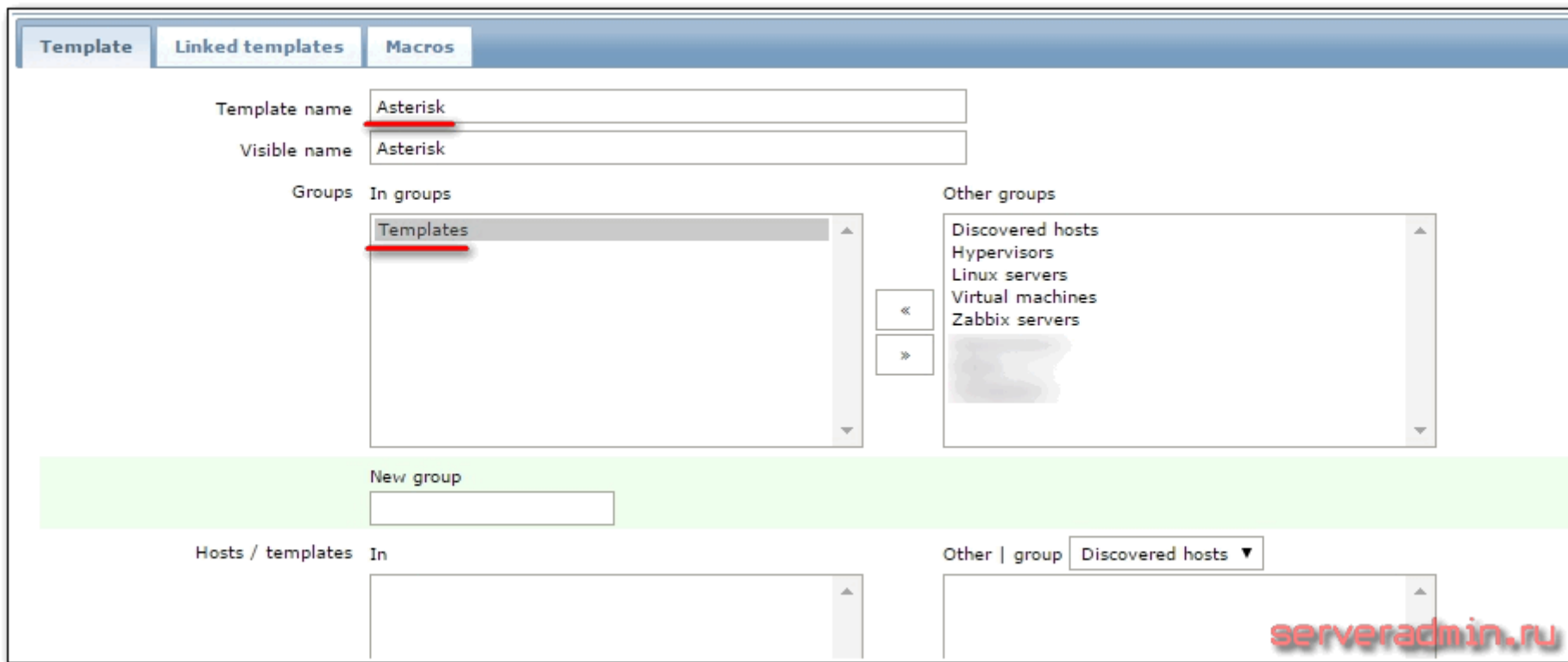
Создание шаблона мониторинга asterisk на сервере zabbix

Нам нужно создать шаблон на сервере, для назначения его серверам с астериском. Здесь ничего сложного, делаем все то же самое, что мы делали ранее, настраивая мониторинг рейда mdadm. Идем в раздел **Configuration -> Templates**, нажимаем **Create template**.

The screenshot shows the Zabbix web interface for the 'CONFIGURATION OF TEMPLATES' section. The navigation menu includes Monitoring, Inventory, Reports, Configuration, and Administration. The 'Configuration' menu is expanded to show Host groups, Templates, Hosts, Maintenance, Actions, Screens, Slide shows, Maps, and Discovery. A search bar is visible on the right. The main content area shows a list of templates with columns for Applications, Items, Triggers, Graphs, Screens, Discovery, and Web. A red arrow points to the 'Create template' button in the top right corner of the main content area. A watermark 'serveradmin.ru' is visible in the bottom right corner of the screenshot.

<input type="checkbox"/>	Templates	Applications	Items	Triggers	Graphs	Screens	Discovery	Web	Linked templates	Linked to
<input type="checkbox"/>	Asterisk	Applications (1)	Items (1)	Triggers (0)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	
<input type="checkbox"/>	Mail server	Applications (1)	Items (3)	Triggers (3)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	
<input type="checkbox"/>	Mdadm_status	Applications (1)	Items (1)	Triggers (1)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	

Пишем название и добавляем его в группу с шаблонами.



Открываем шаблон, переходим в раздел **Items** и ждем на **Create item**. Заполняем параметры нового итема как на картинке.

The screenshot shows the Zabbix configuration interface for a new agent named 'Trunk status'. The form includes the following fields and options:

- Name:** Trunk status
- Type:** Zabbix agent
- Key:** asterisk.trunk
- Type of information:** Numeric (unsigned)
- Data type:** Decimal
- Units:** (empty)
- Use custom multiplier:** (value: 1)
- Update interval (in sec):** 120
- Flexible intervals:** A table with columns 'Interval', 'Period', and 'Action'. It currently shows 'No flexible intervals defined.'
- New flexible interval:** Interval (in sec) 50, Period 1-7,00:00-24:00, Add button.
- History storage period (in days):** 30
- Trend storage period (in days):** 90
- Store value:** As is
- Show value:** As is, with a link 'show value mappings'
- New application:** (empty)
- Applications:** A dropdown menu with '-None-' and 'Asterisk'.
- Populates host inventory field:** -None-
- Description:** (empty text area)
- Enabled:**

At the bottom, there are buttons for 'Update', 'Clone', 'Delete', and 'Cancel', and a watermark 'serveradmin.ru'.

Я поставил интервал проверок 120 секунд = 2 минуты. Вы можете поменять под свои нужды этот параметр. Сохраняем новый итем.

Для сбора данных все готово. Теперь назначаем шаблон серверам с астериском, где мы подготовили агенты мониторинга и скрипты и ждем поступления данных. Проверять как обычно идем в **Latest data**. Через пару минут должны поступить значения.

The screenshot shows the Zabbix 2.4.7 web interface. The top navigation bar includes 'Monitoring', 'Inventory', 'Reports', 'Configuration', and 'Administration'. The breadcrumb trail is 'History: Configuration of items » Configuration of templates » Configuration of triggers » Configuration of items » Latest data'. The main section is titled 'LATEST DATA' and contains a filter section with fields for 'Host groups', 'Hosts', and 'Application' (set to 'Asterisk'). Below the filters are 'Filter' and 'Reset' buttons. The main table has columns: 'Host', 'Name', 'Last check', 'Last value', and 'Change'. One item is listed under the 'voip' host group: 'Trunk status' with a last check of '2016-03-19 22:46:23' and a last value of '0'. A 'Graph' link is visible for this item. The footer shows 'Zabbix 2.4.7 Copyright 2001-2015 by Zabbix SIA' and a watermark 'serveradmin.ru'.

Host	Name	Last check	Last value	Change
voip	Asterisk (1 Item)			
	Trunk status	2016-03-19 22:46:23	0	-

У меня пришло значение 0 как и должно быть. Все регистрации на месте. Теперь нам нужно добавить триггер, который будет отправлять уведомление, если какая-нибудь из регистраций отвалится. Для этого идем в созданный шаблон и переходим в раздел **Triggers**. Жмем **Create trigger** и указываем значения, как у меня.

Name

Expression

[Expression constructor](#)

Multiple PROBLEM events generation

Description

URL

Severity

Enabled

serveradmin.ru

Если 3 последние проверки покажут значение, отличное от 0, сработает триггер и прилетит уведомление на почту. Сохраняем триггер. На этом все. Можете проверить работу любым способом, который придумаете. Например, временно измените пароль на одном из транков, или добавив новый с несуществующими параметрами. Можно и не проверять, все должно и так работать :)

Возможные ошибки

Если ваш новый item не работает, получает статус **not supported**, а в описании причины ошибка:

```
Unable to connect to remote asterisk (does /var/run/asterisk/asterisk.ctl exist?)
```

Необходимо разрешить пользователю zabbix, от которого работает скрипт, запускать asterisk. Для этого добавляем в файл `/etc/sudoers` в самый конец следующую строку:

```
zabbix ALL = NOPASSWD: /usr/sbin/asterisk
```

После этого проверяем выполнение скрипта от пользователя zabbix:

```
# sudo -u zabbix /etc/zabbix/scripts/asterisk.trunk.sh
```

Если получаете вывод значения, значит все в порядке. Если же все равно видите ошибку:

```
Unable to connect to remote asterisk (does /var/run/asterisk/asterisk.ctl exist?)
```

Добавьте в скрипт перед `/usr/sbin/asterisk` выполнение `/usr/bin/sudo`. Должно получиться так:

```
reg_trunks=`/usr/bin/sudo /usr/sbin/asterisk -rx "sip show registry" | grep Registered | wc -l`
```

После этого отцепите шаблон от хоста и добавьте заново. Подождите несколько минут обновления данных. Вы можете снова получить ошибку, но уже другого рода:

```
sudo: sorry, you must have a tty to run sudo
```

Эта ошибка появится, если у вас пользователь zabbix создан без какой-либо оболочки. Это зависит от системы и версии заббикс агента. Проверить можно в файле `/etc/passwd`:


```
# cat /etc/passwd | grep zabbix  
zabbix:x:496:496:Zabbix Monitoring System:/var/lib/zabbix:/sbin/nologin
```

Пользователям без оболочки по-умолчанию запрещено использовать sudo, которое мы добавили в скрипт. Исправить это можно в файле `/etc/sudoers`, закомментировав параметр:

```
#Defaults    requiretty
```

После этого снова отцепляем шаблон и прикрепляем заново. Теперь должно быть все в порядке.

Постарался предусмотреть все варианты развития событий. Наличие тех или иных ошибок будет зависеть от операционной системы и версии zabbix и asterisk.

Заключение

В очередной раз zabbix подтвердил свою простоту и гибкость настроек. Буквально за несколько минут, после беглого безрезультатного поиска в гугле, я смастерил простенький скрипт и добавил его к системе мониторинга. Поставленную задачу он успешно выполняет. Так как ставил на разные системы, пришлось отловить несколько ошибок.

Помогла статья? Есть возможность отблагодарить автора

Дополнительные материалы по Zabbix

[Заказать настройку asterisk от 1500 р.](#)

[Заказать настройку Zabbix от 500 р.](#)

Онлайн курс "Администратор Linux"

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, рекомендую познакомиться с **онлайн-курсом «Администратор Linux»** в OTUS. Курс не для новичков, для поступления нужны базовые знания по сетям и установке Linux на виртуалку. Обучение длится 5 месяцев, после чего успешные выпускники курса смогут пройти собеседования у партнеров. Проверьте себя на вступительном тесте и смотрите программу подробнее по .

Рекомендую полезные материалы по Zabbix:

Настройки системы

- Установка 4.0
- Обновление 3.0 -> 3.2
- Обновление 3.4 -> 4.0
- Установка Zabbix Proxy
- Работа на NGINX

Видео и подробное описание установки и настройки Zabbix 4.0, а также установка агентов на linux и windows и подключение их к мониторингу.

Подробное описание обновления системы мониторинга zabbix версии 3.4 до новой версии 4.0.

Пошаговая процедура обновления сервера мониторинга zabbix 2.4 до 3.0. Подробное описание каждого шага с пояснениями и рекомендациями.

Подробное описание установки и настройки zabbix proxy для организации распределенной системы мониторинга. Все показано на примерах.

Подробное описание установки системы мониторинга Zabbix на веб сервер на базе nginx + php-fpm.

Мониторинг служб и сервисов

- Температура процессора
- Nginx и php-fpm
- Mysql репликация
- Службы Linux
- Рейд mdadm
- Транки Asterisk
- Synology

Мониторинг температуры процессора с помощью zabbix на Windows сервере с использованием пользовательских скриптов.

Настройка полноценного мониторинга web сервера nginx и php-fpm в zabbix с помощью скриптов и пользовательских параметров.

Мониторинг репликации mysql с помощью Zabbix. Подробный разбор методики и тестирование работы.

Описание настройки мониторинга tcp служб с помощью zabbix и его инструмента простых проверок (simple checks)

Настройка мониторинга рейда mdadm с помощью zabbix. Подробное пояснение принципа работы и пошаговая инструкция.

Подробное описание мониторинга регистраций транков (trunk) в asterisk с помощью сервера мониторинга zabbix.

Подробная инструкция со скриншотами по настройке мониторинга по snmp дискового хранилища synology с помощью сервера мониторинга zabbix.

Мониторинг различных значений

- Мониторинг сайта
- Мониторинг бэкапов
- Размер бэкапа
- Делегирование домена
- Значения из текстового файла
- Мониторинг логов

Настройка мониторинга web сайта в zabbix. Параметры для наблюдения - доступность сайта, время отклика, скорость доступа к сайту.

Один из способов мониторинга бэкапов с помощью zabbix через проверку даты последнего изменения файла из архивной копии с помощью vfs.file.time.

Подробное описание настройки мониторинга размера бэкапов в Zabbix с помощью внешних скриптов.

Пример настройки мониторинга за временем делегирования домена с помощью Zabbix и внешнего скрипта. Все скрипты и готовый шаблон представлены.

Пример распознавания и мониторинга за изменением значений в обычных текстовых файлах с помощью zabbix.

Описание мониторинга лог файлов в zabbix на примере анализа лога программы arcpupd. Отправка оповещений по событиям из лога.