



Сервер телефонии asterisk востребованный и популярный функционал на сегодняшний день. Ранее я подробно о нем рассказывал и показывал, как его использовать в среднестатистическом офисе. Сегодня хочу рассказать, как настроить мониторинг asterisk с помощью сервера мониторинга zabbix. Я подробно расскажу о метриках, которые буду снимать и в конце представлю готовый шаблон для использования.

Если у вас есть желание научиться искать и эксплуатировать уязвимости в информационных сетях, рекомендую познакомиться с **онлайн-курсом «Практикум по Kali Linux»** в OTUS. Курс рассчитан на тех, у кого нет опыта в информационной безопасности, для поступления нужно пройти .

Содержание:

- 1 Введение
- 2 Параметры мониторинга asterisk
- 3 Настройка zabbix-agent
- 4 Мониторинг asterisk в zabbix сервере
- 5 Заключение

## Введение

У меня есть подробный материал по пошаговой настройке asterisk. В нем не хватает для полноты картины мониторинга. Без него обслуживание и поддержка работоспособности сервера затруднительна. Я всегда стараюсь все, что можно, завести на мониторинг. Так спокойнее живется. Ранее я писал о мониторинге транков. В принципе, на простейший вопрос по доступности транков он отвечает, но в процессе эксплуатации я много раз переделывал этот мониторинг, потому что он малоинформативен.

Например, у вас периодически отваливает один из транков. Мониторинг вам об этом говорит, но вы не знаете, что за транк отвалился и по какой причине. Надо идти на сервер и проверять глазами. Когда на одном из серверов у меня начали глючить пару транков, я не выдержал и переделал этот мониторинг.



А заодно и добавил некоторые другие проверки, которые мне показались важными. В итоге получился законченный шаблон для мониторинга всего сервера asterisk. Далее я расскажу обо всем подробнее.

Я все настраивал на сервере с версией 3.2, агенты той же версии. Работоспособность в 3.4 не проверял, но, по идее, все должно работать и там. Если у вас еще нет сервера мониторинга, то можете его установить и настроить — 3.2 или 3.4 на Centos 7 или 3.4 на Debian 9.

## Параметры мониторинга asterisk

В интернете есть примеры мониторинга asterisk. Кое-что я оттуда подсмотрел, но готового решения, которое бы мне подошло полностью, я не увидел, поэтому решил сделать по-своему. Я подумал и решил, что мне полезны для мониторинга следующие метрики:

1. Состояние транков. Если один из них отваливается, я узнаю о его имени уже в оповещении на почте.
2. Состояние самой службы asterisk на сервере. Тут все просто — либо работает, либо нет.
3. Состояние работы программы fail2ban. Тоже все просто — либо работает, либо нет.
4. Наличие цепочек fail2ban в таблице iptables. Эта проверка гарантирует, что fail2ban не только запущен, но и успешно блокирует нарушителей.
5. Факт перезапуска сервиса asterisk. Как бонус к этой метрике — uptime службы астериска.
6. Количество активных разговоров в данный момент.

Небольшие комментарии к метрикам. Если сервер смотрит в интернет, то обязательно наличие iptables и fail2ban. Боты будут регулярно к вам стучаться и перебирать учетки. Так что следить за работой этих служб необходимо. По транкам я уже сказал, это один из основных параметров, так как время от времени они отваливаются. Гораздо чаще, чем сами сервисы. На всякий случай будем следить за работой сервиса астериск на сервере. Он хоть у меня обычно и не падает, даже не припомню такого, но теоретически с ним это может происходить. Мониторинг перезагрузки и аптайма службы сделан в основном для любопытства. Практической пользы в этой метрике я не вижу, если вы единственный админ на сервере и без вас его никто не перезапускает. Мониторинг за количеством активных звонков в asterisk сделан для того, чтобы можно было оценить примерную нагрузку и занятость линий.

Я больше не придумал, на что еще стоит обратить внимание. Некоторые считают количество звонков в день, количество транков, пиров, их статус. Я лично не вижу в этом большого смысла. Если кто-то считает, что есть еще важные параметры в asterisk, которые надо мониторить, прошу поделиться.

## Настройка zabbix-agent

Для мониторинга за asterisk, я буду использовать один скрипт и несколько обработанных выводов из запросов к астериску. Начнем со скрипта. Он будет



проверять статус транков и если какой-либо из них будет в offline, назовет его. Кладем этот скрипт в директорию `/etc/zabbix/scripts`.

```
# cat /etc/zabbix/scripts/asterisk.trunk-with-name.sh
```

```
#!/bin/bash

# Получаем количество всех транков в системе
total=`sudo asterisk -rx 'sip show registry' | sed -n '/registrations/p' | awk '{print $1}'`
# Получаем число активных транков
active=`sudo asterisk -rx 'sip show registry' | sed -n '/Registered/p' | wc -l`
# Получаем имена транков с проблемам
offline=`sudo asterisk -rx 'sip show registry' | sed -n '/Request\|Rejected\|Authentication\|Auth/p' | awk '{print $3}'`
# Сравниваем общее число с числом активных транков и выводим сообщение об их состоянии
if [ $active -lt $total ]
then
echo Trunks offline $offline
else
echo All trunks are online
fi
```

Я все прокомментировал. Добавлю только пояснение к регулярке, которая определяет проблемные транки. Я знаком со следующими состояниями транков, когда они не работают:

- Request Sent
- No Authentication
- Auth. Sent
- Rejected

Соответственно, эти состояния я и ищу. Если один из транков находится в одном из этих состояний, то он считается проблемным.

Далее создаем файл конфигурации zabbix с UserParameters в `/etc/zabbix/zabbix_agentd.d`.



```
# cat /etc/zabbix/zabbix_agentd.d/asterisk.conf
```

```
# Статус службы fail2ban
UserParameter=asterisk.fail2ban_status,ps cax | grep fail2ban | wc -l
# Количество цепочек fail2ban в iptables
UserParameter=asterisk.fail2ban_chain,iptables -nL | grep Chain | grep -E 'f2b|fail2ban' | wc -l
# Время работы службы asterisk
UserParameter=asterisk.uptime,asterisk -rx "core show uptime seconds" | grep --text -i "System uptime:" | gawk '{print $3}'
# Количество активных разговоров
UserParameter=asterisk.active_calls,asterisk -rvvvvx 'core show channels'| grep --text -i 'active call'| cut -c1
# Статус транков
UserParameter=asterisk.trunk,/etc/zabbix/scripts/asterisk.trunk-with-name.sh
# Статус службы asterisk
UserParameter=asterisk.asterisk_status,ps cax | grep asterisk | wc -l
```

Некоторые из приведенных метрик требуют права root для своего исполнения. Скажу честно, мне было лениво разбираться с разрешениями и я просто запустил zabbix с правами root. Для этого в его конфиге `/etc/zabbix/zabbix_agentd.conf` раскомментировал следующую строку:

```
AllowRoot=1
```

После этого можно перезапустить zabbix-agent и проверить работу будущих итемов.

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

Проверяем, насколько корректно агент возвращает указанные значения.

```
# zabbix_agentd -t asterisk.asterisk_status
asterisk.asterisk_status [t|2]
```

```
# zabbix_agentd -t asterisk.fail2ban_chain
```



```
asterisk.fail2ban_chain [t|2]
```

```
# zabbix_agentd -t asterisk.trunk  
asterisk.trunk [t|All trunks are online]
```

Все в порядке. В завершении настройки агента, поясню некоторые моменты для тех, кто не очень разбирается в bash скриптах. Для мониторинга статусов служб я просто вывожу список всех процессов, вычлняю из него нужный мне процесс, в данном случае asterisk или fail2ban и подсчитываю количество этих процессов. Далее в шаблоне я просто буду проверять число этих процессов. Если оно будет равно 0, значит процессы не работают.

Для анализа цепочки iptables я также обрабатываю ее вывод, ищу строки со словами fail2ban и подсчитываю их количество. Если они равны 0, значит fail2ban не работает, либо работает, но не добавляет правила в iptables, что по большому счету и равно тому, что он не работает.

Для мониторинга активных разговоров, я обрабатываю соответствующий вывод asterisk, вырезаю лишние строки и оставляю только необходимое мне число.

## Мониторинг asterisk в zabbix сервере

Всю теорию я изложил выше. Осталось только добавить готовый шаблон на сервер и посмотреть на результат. Скачиваем шаблон — [zabbix-asterisk-template.xml](#) и импортируем его на сервер. В шаблоне присутствуют 6 элементов данных, которые мы определили в агенте, 5 триггеров и 1 график просто для красоты, чтобы был :). В этом шаблоне надобности в графиках нет. Остановлюсь подробнее на триггерах.



Таблица триггеров в шаблоне мониторинга asterisk

Asterisk down on {HOST.NAME}	Срабатывает, если последняя проверка количества запущенных процессов астериск вернула 0.
Asterisk restarted on {HOST.NAME}	Триггер реагирует на изменение uptime службы. Если он стал меньше 300 секунд, то считается, что служба была перезапущена.
Fail2ban down on {HOST.NAME}	Срабатывает, если последняя проверка количества запущенных процессов fail2ban вернула 0.
Fail2ban inactive on {HOST.NAME}	Проверяет, есть ли в правилах iptables цепочки fail2ban. Если нет, срабатывает.
Trunk not registered on {HOST.NAME}	Срабатывает, если хотя бы одна из регистраций отвалилась.



Первые 4 триггера очевидны, по ним комментировать нечего. Остановлюсь на последнем с проверкой транков. Данный триггер проверяет текстовую строку «All trunks are online», которую возвращает итем. Если 2 раза подряд ее не было, значит проблема, и идет оповещение. Проблема закрывается, когда она же появляется 2 раза подряд. Сделал специально 2 проверки, чтобы не было ложных срабатываний на моментах перерегистрации или кратковременных сбоях, которые не требуют вмешательства.

После применения шаблона на хостах, увидите примерно такую картинку в «Последних данных».



Если какой-то транк отвалится, то в оповещении на почте будет его название. У меня вот так выглядят подобные оповещения.



На этом по мониторингу asterisk в zabbix все. Надеюсь, раскрыл тему достаточно подробно и понятно.

## Заключение

Не понравилась статья и хочешь научить меня администрировать? Пожалуйста, я люблю учиться. Комментарии в твоём распоряжении. Расскажи, как сделать правильно!

Уже в процессе написания статьи понял, чего в ней не хватает. Мониторинг пиров все же нужен, так как не всегда подключение к провайдеру идет с регистрацией. Если ее нет, то за провайдерским пиром надо бы наблюдать. Возможно, дополню статью позже. Сервера, на которых я все проверял не имели таких пиров, поэтому упустил этот момент, не обратил внимание.

Конечно, не очень правильно запускать zabbix от root. Тут я схалювил, но у меня есть оправдание — в заббиксе нет нормального средства для дебага не работающих скриптов. Итем просто становится неподдерживаемым и частенько приходится гадать, в чем же проблема. Даже когда выставляешь все необходимые права, все перепроверяешь, на стороне агента запускаешь и все работает как надо. Но на сервере итем не принимает данные.

Данный шаблон должен работать вместе со стандартным шаблоном для linux. Я видел в сети примеры шаблонов мониторинга asterisk, куда были добавлены метрики таких параметров сервера, как память, процессор, сеть и т.п. Не вижу смысла в таком подходе. Шаблоны надо максимально



разделять, чтобы ими было удобнее пользоваться и не нагружать сервера лишними метриками. Я под каждый функционал сервера делаю отдельный шаблон. Примеры таких шаблонов можно посмотреть в моей рубрике zabbix.

На этом все. Не забываем меня научить, как надо администрировать в комментариях :)

## Онлайн курс "DevOps практики и инструменты"

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, научиться непрерывной поставке ПО, мониторингу и логированию web приложений, рекомендую познакомиться с **онлайн-курсом «DevOps практики и инструменты»** 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 детальнее по .

Помогла статья? Есть возможность отблагодарить автора