

Настройка журналирования на сетевом оборудовании

Администрирование сетевых подсистем

Кобзев Д. К.

Российский университет дружбы народов, Москва, Россия

24 ноября 2025

Информация

- ▶ Кобзев Дмитрий Константинович
- ▶ Студент
- ▶ Российский университет дружбы народов
- ▶ НПИбд-01-23

Тема: Настройка журналирования на сетевом оборудовании

Актуальность: В современных сетях, где количество устройств и сервисов постоянно растёт, оперативное отслеживание их состояния становится критически важным элементом администрирования сетевых подсистем. Журналирование системных событий позволяет:

- ▶ Выявлять сбои и нарушения безопасности в реальном времени
- ▶ Проводить анализ инцидентов после их возникновения
- ▶ Соответствовать требованиям стандартов информационной безопасности
- ▶ Оптимизировать работу сетевой инфраструктуры

Цель: Показать практические аспекты настройки централизованной системы журналирования в сетевой инфраструктуре на базе rsyslog, включая:

- ▶ Настройку сервера для приёма логов
- ▶ Конфигурацию клиентов для отправки данных
- ▶ Инструменты для просмотра и анализа собранной информации

В системах на базе Unix/Linux важное место при администрировании занимает отслеживание системных событий (и в частности возникновение возможных ошибок в процессе настройки каких-то служб) через ведение log-файлов процессов системы. Журналирование системных событий заключается в фиксировании с помощью сокета syslog в лог-файлах сообщений об ошибках и сообщений о состоянии работы практически всех процессов системы. Обычно лог-файлы располагаются в каталоге `/var/log`.

Для управления логированием событий обычно используется служба syslog или её модификация rsyslog. С их помощью можно настроить уровень подробности логирования для каждого процесса. Все настройки rsyslog находятся в файле `/etc/rsyslog.conf`. В этот же файл подключаются дополнительные файлы настройки из каталога `/etc/rsyslog.d/`.

Зачем нужен сервер сетевого журнала

Сохранение всех событий системы приводит к быстрому заполнению дискового пространства. Кроме того, если требуется администрировать несколько узлов сети, то удобнее это делать с одного узла:

- ▶ проще обеспечить безопасность и целостность лог-сообщений, которые в этом случае не будут доступны злоумышленнику, если не нарушена безопасность самого сервера;
- ▶ проще и удобнее управлять дисковым пространством и политиками по времени хранения информации в журналах, в том числе настроив logrotate для сохранения сообщений в течение более длительного периода, чем период по умолчанию;
- ▶ проверять файлы журналов на одном сервере проще, чем подключиться к нескольким серверам для анализа информации, которая была зарегистрирована.

В зависимости от того, информация какого типа фиксируется системой, формируются разные типы записей. Поэтому логи делятся на: системные, серверные, почтовые, логи аутентификации, авторизации, log file приложений, баз данных и т.д. Подобная группировка помогает быстрее находить нужный лог и оптимизировать работу с ним.

Часто используется несколько уровней важности сообщений, таких как DEBUG, INFO, WARNING, ERROR и CRITICAL. Это помогает фильтровать и приоритизировать информацию в журналах.

Настройка сервера сетевого журнала

Создаем файл конфигурации сетевого хранения журналов

```
cd /etc/rsyslog.d  
touch netlog-server.conf
```

Включаем в нем приём записей журнала по TCP-порту 514:

```
$ModLoad imtcp  
$InputTCPServerRun 514
```

Перезапускаем службу rsyslog:

```
systemctl restart rsyslog
```

Настраиваем межсетевой экран для приёма сообщений по TCP-порту 514:

```
firewall-cmd --add-port=514/tcp  
firewall-cmd --add-port=514/tcp --permanent
```


Настройка клиента сетевого журнала

Создаем файл конфигурации сетевого хранения журналов:

```
cd /etc/rsyslog.d  
touch netlog-client.conf
```

Включаем в нем перенаправление сообщений журнала на 514 TCP-порт сервера:

```
*.* @@server.user.net:514
```

Перезапускаем службу rsyslog:

```
systemctl restart rsyslog
```

После настройки можно просматривать логи на сервере с помощью:

- ▶ Команды `tail -f /var/log/messages`
- ▶ Графических утилит, например `gnome-system-monitor`
- ▶ Специализированных программ, таких как `lnav`

Внесение изменений в настройки внутреннего окружения виртуальных машин

Для автоматизации всех описанных этапов настройки можно использовать скрипты на базе Vagrant. Процесс автоматизации реализуется через создание provisioning-скриптов для сервера и клиента. Эти скрипты размещаются в соответствующих каталогах `/vagrant/provision/` и выполняют полную настройку при запуске виртуальных машин.

Серверный скрипт `netlog.sh` автоматически копирует конфигурационные файлы из подготовленного каталога в целевую систему, настраивает `firewall` для открытия TCP-порта 514 и перезапускает службу `rsyslog`. Клиентский скрипт выполняет аналогичные операции, дополнительно устанавливая необходимые пакеты вроде `lnav` для анализа логов.

Интеграция с Vagrant обеспечивается через добавление соответствующих секций в `Vagrantfile`, где указываются пути к скриптам для сервера и клиента. Это позволяет автоматически развернуть полностью готовую систему сетевого журналирования без ручного вмешательства.

Централизованное журналирование с использованием rsyslog и автоматизация развертывания через Vagrant позволяют создать надежную систему мониторинга сетевой инфраструктуры. Данный подход обеспечивает безопасное хранение логов, упрощает администрирование и ускоряет реакцию на инциденты.