

Отчёт по лабораторной работе 15

Настройка сетевого журналирования

Элсаиед Адел

Содержание

1	Цель работы	5
2	Выполнение	6
2.1	Настройка сервера и клиента сетевого журнала с использованием rsyslog	6
2.2	Настройка клиента сетевого журнала	7
2.3	Просмотр и анализ журналов	8
2.4	Внесение изменений в настройки внутреннего окружения виртуальных машин	11
3	Вывод	14
4	Контрольные вопросы	15

Список иллюстраций

2.1	Настройка приёма журналов по ТСР	6
2.2	Настройка межсетевого экрана для ТСР 514	7
2.3	Настройка пересылки журналов на сервер	8
2.4	Просмотр системных журналов на сервере	8
2.5	Ошибка установки lnav	10
2.6	Подготовка конфигурации netlog на сервере	11
2.7	Provisioning-скрипт netlog.sh для сервера	12
2.8	Подготовка конфигурации netlog на клиенте	12
2.9	Provisioning-скрипт netlog.sh для клиента	13

Список таблиц

1 Цель работы

Получение навыков по работе с журналами системных событий.

2 Выполнение

2.1 Настройка сервера и клиента сетевого журнала с использованием rsyslog

1. На серверной виртуальной машине в каталоге `/etc/rsyslog.d` был создан файл конфигурации сетевого хранения журналов `netlog-server.conf`. Данный файл предназначен для вынесения настроек приёма сетевых журналов в отдельный конфигурационный модуль `rsyslog`.
2. В файле `/etc/rsyslog.d/netlog-server.conf` была выполнена настройка приёма журналов по TCP-протоколу. С помощью директивы `$ModLoad imtcp` загружен модуль приёма сообщений по TCP, а директива `$InputTCPServerRun 514` активирует прослушивание стандартного порта 514 для сетевого журналирования.

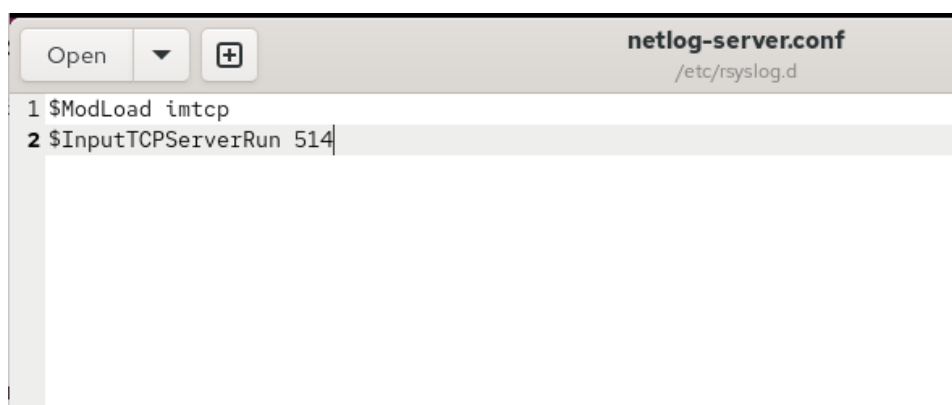


Рис. 2.1: Настройка приёма журналов по TCP

3. После внесения изменений служба rsyslog была перезапущена. Для проверки корректности запуска сервиса и подтверждения факта прослушивания сетевого порта выполнен анализ активных TCP-соединений. В выводе зафиксировано, что процесс rsyslogd находится в состоянии LISTEN на TCP-порту 514, что подтверждает успешную активацию сетевого приёма журналов.
4. Для обеспечения доступа клиентов к серверу сетевого журнала была выполнена настройка межсетевого экрана. TCP-порт 514 был открыт как временно, так и на постоянной основе, что обеспечивает приём сетевых журналов после перезагрузки системы.

```

smbd[192. 18952          elsaiedadel 33u  IPv4           120104      0t0      TCP www.elsaiedadel.
net:microsoft-ds->client.elsaiedadel.net:47188 (ESTABLISHED)
rsyslogd 20540          root    4u  IPv4           146032      0t0      TCP *:shell (LISTEN)
rsyslogd 20540          root    5u  IPv6           146033      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20543 in:imjour root    4u  IPv4           146032      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20543 in:imjour root    5u  IPv6           146033      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20544 in:imtcp root    4u  IPv4           146032      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20544 in:imtcp root    5u  IPv6           146033      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20545 w0/imtcp root    4u  IPv4           146032      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20545 w0/imtcp root    5u  IPv6           146033      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20546 w1/imtcp root    4u  IPv4           146032      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20546 w1/imtcp root    5u  IPv6           146033      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20547 rs:main root    4u  IPv4           146032      0t0      TCP *:shell (LISTEN)
rsyslogd 20540 20547 rs:main root    5u  IPv6           146033      0t0      TCP *:shell (LISTEN)
[root@server.elsaiedadel.net rsyslog.d]# firewall-cmd --add-port=514/tcp
success
[root@server.elsaiedadel.net rsyslog.d]# firewall-cmd --add-port=514/tcp --permanent
success
[root@server.elsaiedadel.net rsyslog.d]#

```

Рис. 2.2: Настройка межсетевого экрана для TCP 514

2.2 Настройка клиента сетевого журнала

5. На клиентской виртуальной машине в каталоге /etc/rsyslog.d был создан файл конфигурации netlog-client.conf, предназначенный для настройки перенаправления журналов на удалённый сервер.
6. В файле /etc/rsyslog.d/netlog-client.conf была добавлена директива `*.* @server.elsaiedadel.net:514`, которая обеспечивает отправку всех сообщений журналов по TCP-протоколу на сервер сетевого журнала. Использование двойного символа @@ гарантирует надёжную доставку сообщений.



Рис. 2.3: Настройка пересылки журналов на сервер

7. После внесения изменений служба rsyslog на клиенте была перезапущена. В результате клиентская система начала передачу системных сообщений на сервер сетевого журналирования.

2.3 Просмотр и анализ журналов

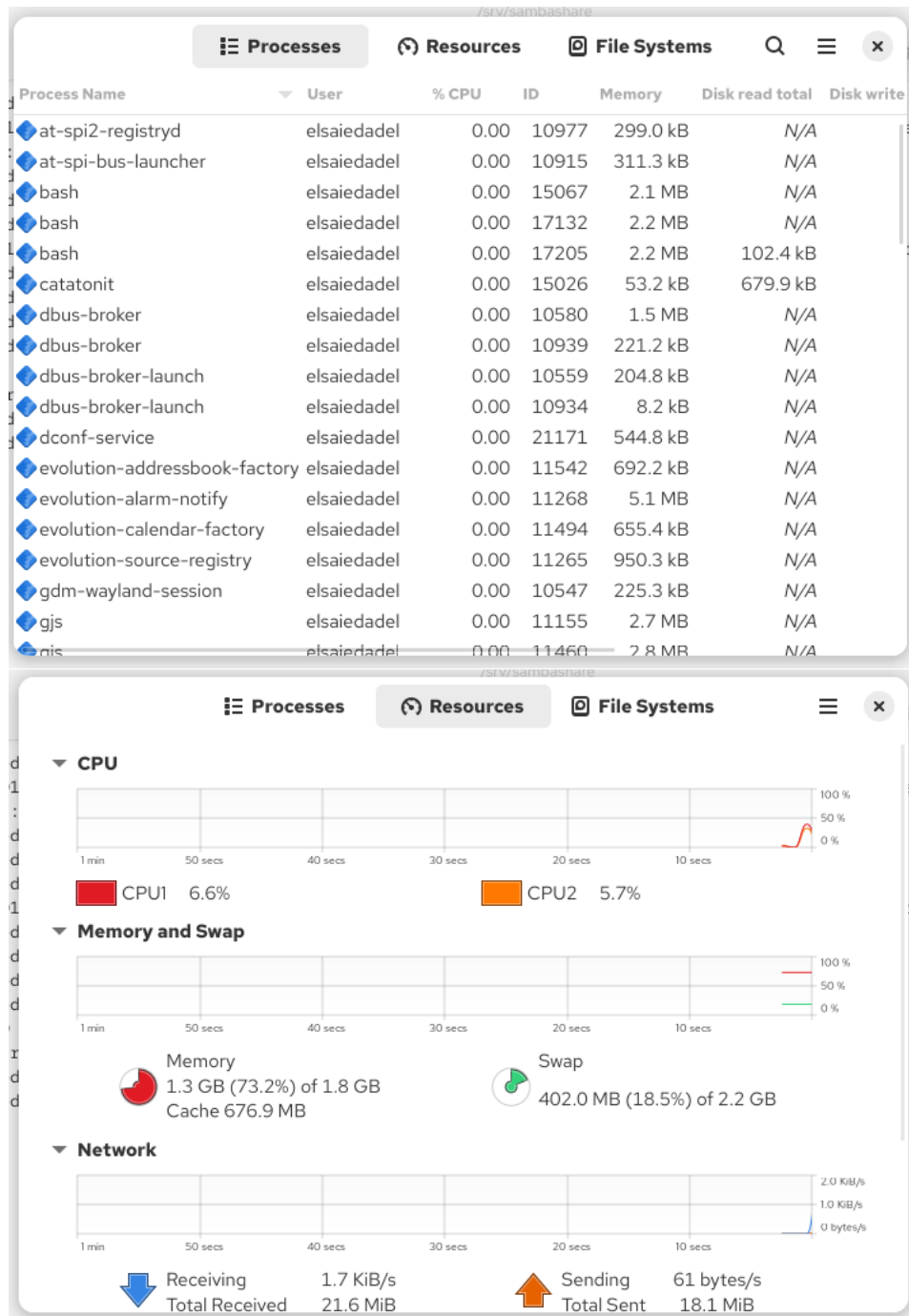
8. На сервере был выполнен просмотр файла системных сообщений /var/log/messages. В процессе наблюдения за журналом были зафиксированы записи, поступающие как от сервера, так и от клиента. В сообщениях отображается имя хоста клиента и информация о работе системных служб, что подтверждает корректную настройку сетевого журналирования.

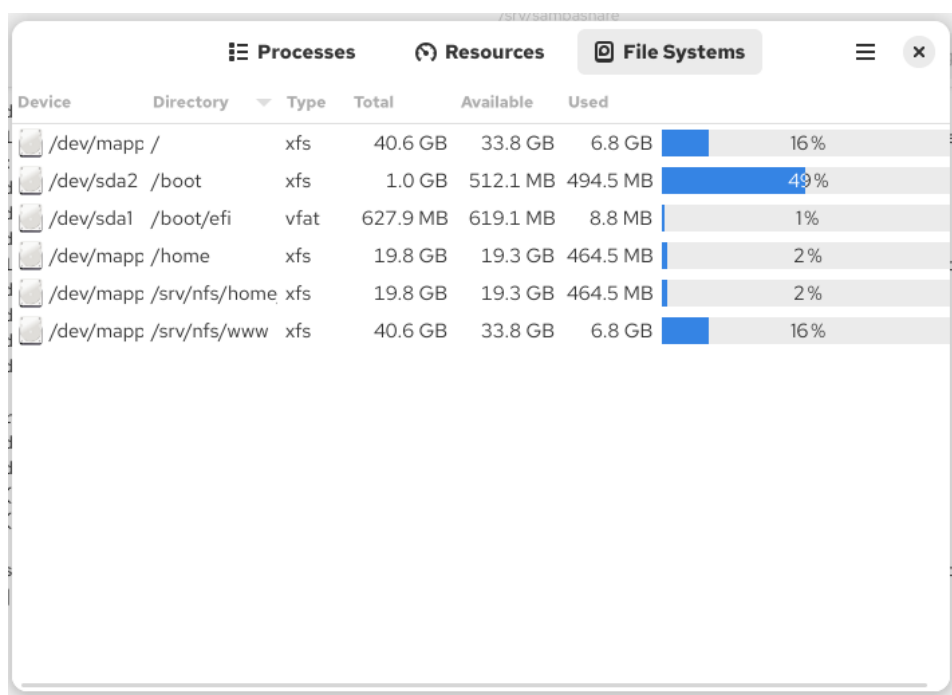
```
Jan 7 08:28:12 client systemd[1]: Started systemd-coredump@480-19045-0.service - Process Core Dump (PID 19045/UID 0).
Jan 7 08:28:12 client systemd-coredump[19046]: Process 19041 (VBoxClient) of user 1001 dumped core.#012#012Module lib
Xau.so.6 from rpm libXau-1.0.11-8.el10.x86_64#012Module libxcb.so.1 from rpm libxcb-1.17.0-3.el10.x86_64#012Module lib
X11.so.6 from rpm libX11-1.8.10-1.el10.x86_64#012Module libffi.so.8 from rpm libffi-3.4.4-9.el10.x86_64#012Module libw
ayland-client.so.0 from rpm wayland-1.23.0-2.el10.x86_64#012Stack trace of thread 19044:#012#0 0x00000000041dd1b n/a
(n/a + 0x0)#012#1 0x0000000000041dc94 n/a (n/a + 0x0)#012#2 0x0000000000045041c n/a (n/a + 0x0)#012#3 0x00000000000043
55d0 n/a (n/a + 0x0)#012#4 0x00007f5ddec15b68 start_thread (libc.so.6 + 0x94b68)#012#5 0x00007f5ddec866bc __clone3 (
libc.so.6 + 0x1056bc)#012#012Stack trace of thread 19041:#012#0 0x00007f5ddec844bd syscall (libc.so.6 + 0x1034bd)#012
#1 0x000000000004344e2 n/a (n/a + 0x0)#012#2 0x00000000000450066 n/a (n/a + 0x0)#012#3 0x00000000000405123 n/a (n/a +
0x0)#012#4 0x00007f5ddebab30e __libc_start_call_main (libc.so.6 + 0x2a30e)#012#5 0x00007f5ddebab3c9 __libc_start_mai
n@GLIBC_2.34 (libc.so.6 + 0x2a3c9)#012#6 0x000000000004044aa n/a (n/a + 0x0)#012ELF object binary architecture: AMD x
86-64
Jan 7 08:28:12 client systemd[1]: systemd-coredump@480-19045-0.service: Deactivated successfully.
Jan 7 08:28:13 server systemd[1]: serial-getty@tty50.service: Deactivated successfully.
Jan 7 08:28:13 server kernel: traps: VBoxClient[20839] trap int3 ip:41dd1b sp:7f6d3a635cd0 error:0 in VBoxClient[1dd1
b,400000+bb000]
Jan 7 08:28:13 server systemd-coredump[20840]: Process 20836 (VBoxClient) of user 1001 terminated abnormally with sig
nal 5/TRAP, processing...
```

Рис. 2.4: Просмотр системных журналов на сервере

9. Для анализа текущего состояния системы под пользовательской сессией

была запущена графическая утилита мониторинга ресурсов. С её помощью выполнен просмотр списка активных процессов, загрузки процессора, использования оперативной памяти, сетевой активности и состояния файловых систем.





- Для расширенного анализа журналов предпринята попытка установки консольного просмотрщика системных сообщений `lnav`. В процессе установки получено сообщение об отсутствии пакета в доступных репозиториях, что указывает на необходимость подключения дополнительных источников программного обеспечения либо использования альтернативных средств анализа журналов.

```
[root@server.elsaiedadel.net rsyslog.d]#
[root@server.elsaiedadel.net rsyslog.d]# dnf -y install lnav
Last metadata expiration check: 0:11:37 ago on Wed 07 Jan 2026 08:19:40 AM UTC.
No match for argument: lnav
Error: Unable to find a match: lnav
[root@server.elsaiedadel.net rsyslog.d]#
```

Рис. 2.5: Ошибка установки `lnav`

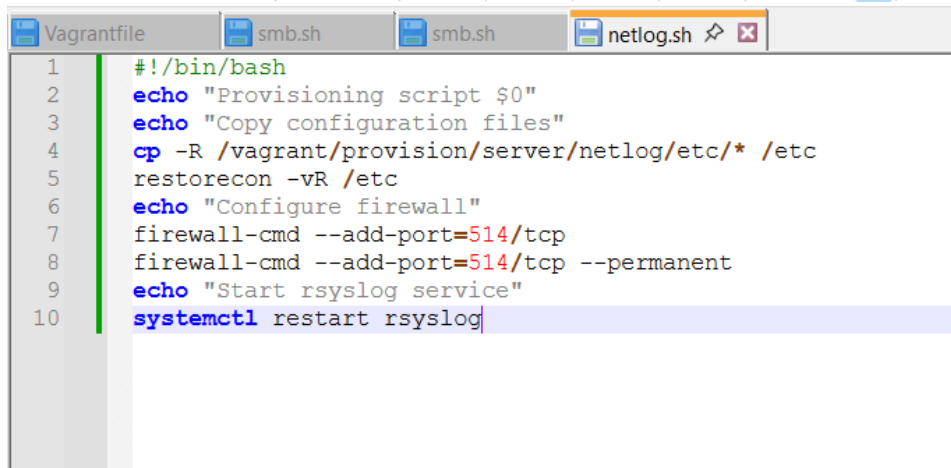
2.4 Внесение изменений в настройки внутреннего окружения виртуальных машин

1. На виртуальной машине **server** выполнен переход в каталог `/vagrant/provision/server`, предназначенный для размещения provisioning-скриптов и конфигурационных файлов внутреннего окружения. В данном каталоге создана иерархия `netlog/etc/rsyslog.d`, в которую был скопирован файл конфигурации сетевого журнала сервера `netlog-server.conf`. Это обеспечивает централизованное хранение конфигурации и возможность её автоматического применения при разворачивании виртуальной машины.

```
[root@server.elsaiedadel.net rsyslog.d]#  
[root@server.elsaiedadel.net rsyslog.d]# cd /vagrant/provision/server/  
[root@server.elsaiedadel.net server]# mkdir -p /vagrant/provision/server/netlog/etc/rsyslog.d  
[root@server.elsaiedadel.net server]# cp -R /etc/rsyslog.d/netlog-server.conf /vagrant/provision/server/netlog/etc/rsyslog.d  
[root@server.elsaiedadel.net server]# touch netlog.sh  
[root@server.elsaiedadel.net server]# chmod +x netlog.sh  
[root@server.elsaiedadel.net server]#
```

Рис. 2.6: Подготовка конфигурации netlog на сервере

2. В каталоге `/vagrant/provision/server` создан исполняемый файл `netlog.sh`, предназначенный для автоматизации настройки сетевого журналирования на сервере. Файл был помечен как исполняемый и отредактирован. В скрипте реализованы операции копирования конфигурационных файлов в системный каталог `/etc`, восстановления контекста SELinux, настройки межсетевого экрана для TCP-порта 514 и перезапуска службы `rsyslog`.



```
Vagrantfile  smb.sh  smb.sh  netlog.sh
1  #!/bin/bash
2  echo "Provisioning script $0"
3  echo "Copy configuration files"
4  cp -R /vagrant/provision/server/netlog/etc/* /etc
5  restorecon -vR /etc
6  echo "Configure firewall"
7  firewall-cmd --add-port=514/tcp
8  firewall-cmd --add-port=514/tcp --permanent
9  echo "Start rsyslog service"
10 systemctl restart rsyslog
```

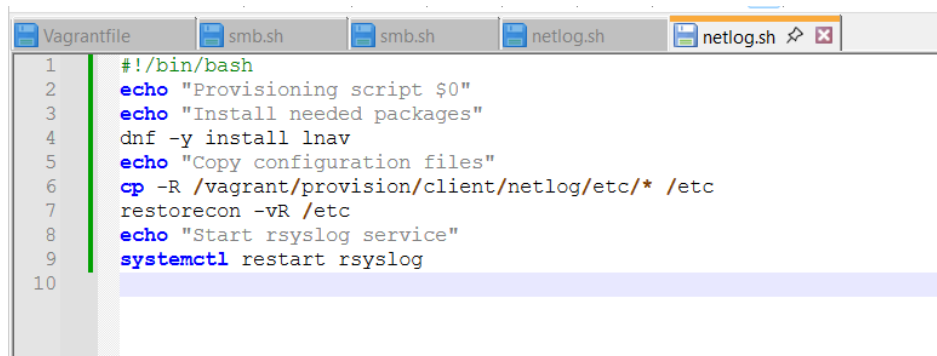
Рис. 2.7: Provisioning-скрипт netlog.sh для сервера

3. На виртуальной машине **client** выполнен переход в каталог `/vagrant/provision/client`. Аналогично серверу, была создана структура каталогов `netlog/etc/rsyslog.d`, в которую скопирован файл конфигурации клиента `netlog-client.conf`. Данная структура используется для автоматического применения клиентских настроек сетевого журналирования.

```
[root@client.elsaiedadel.net rsyslog.d]#
[root@client.elsaiedadel.net rsyslog.d]# cd /vagrant/provision/client/
[root@client.elsaiedadel.net client]# mkdir -p /vagrant/provision/client/netlog/etc/rsyslog.d
[root@client.elsaiedadel.net client]# cp -R /etc/rsyslog.d/netlog-client.conf /vagrant/provision/client/netlog/etc
/rsyslog.d
[root@client.elsaiedadel.net client]# touch netlog.sh
[root@client.elsaiedadel.net client]# chmod +x netlog.sh
[root@client.elsaiedadel.net client]# █
```

Рис. 2.8: Подготовка конфигурации netlog на клиенте

4. В каталоге `/vagrant/provision/client` создан исполняемый provisioning-скрипт `netlog.sh`. Скрипт содержит команды для установки дополнительных пакетов, включая просмотрщик журналов, копирования конфигурационных файлов в каталог `/etc`, восстановления контекста безопасности SELinux и перезапуска службы `rsyslog`. Это обеспечивает корректную инициализацию клиента при каждой загрузке виртуальной машины.

A screenshot of a terminal window with a tabbed interface. The tabs are labeled 'Vagrantfile', 'smb.sh', 'smb.sh', 'netlog.sh', and 'netlog.sh'. The active tab is 'netlog.sh'. The terminal shows a shell script with line numbers 1 through 10 on the left. The script content is: 1: #!/bin/bash, 2: echo "Provisioning script \$0", 3: echo "Install needed packages", 4: dnf -y install lnav, 5: echo "Copy configuration files", 6: cp -R /vagrant/provision/client/netlog/etc/* /etc, 7: restorecon -vR /etc, 8: echo "Start rsyslog service", 9: systemctl restart rsyslog, 10: (blank line).

```
1  #!/bin/bash
2  echo "Provisioning script $0"
3  echo "Install needed packages"
4  dnf -y install lnav
5  echo "Copy configuration files"
6  cp -R /vagrant/provision/client/netlog/etc/* /etc
7  restorecon -vR /etc
8  echo "Start rsyslog service"
9  systemctl restart rsyslog
10
```

Рис. 2.9: Provisioning-скрипт netlog.sh для клиента

5. Для автоматического выполнения созданных provisioning-скриптов во время запуска виртуальных машин были внесены изменения в конфигурационный файл Vagrantfile. В разделы конфигурации виртуальных машин **server** и **client** добавлены инструкции shell-провижининга с указанием путей к соответствующим скриптам netlog.sh и сохранением порядка выполнения. Это обеспечивает автоматическую настройку сетевого журналирования при каждом развертывании окружения.

3 Вывод

В ходе выполнения лабораторной работы была выполнена настройка централизованного сетевого журналирования на базе службы rsyslog в клиент-серверной архитектуре. Реализован приём системных сообщений по TCP-протоколу на сервере и их перенаправление с клиентской виртуальной машины. Дополнительно выполнена автоматизация настройки внутреннего окружения виртуальных машин с использованием provisioning-скриптов Vagrant, что обеспечило воспроизводимость конфигурации при каждом запуске виртуальных машин. Проверка журналов подтвердила корректную передачу и обработку сообщений, а также возможность централизованного мониторинга состояния сервисов и системы в целом.

4 Контрольные вопросы

1. Какой модуль rsyslog вы должны использовать для приёма сообщений от journald?

Для приёма сообщений от journald в rsyslog используется модуль `imjournal`. Данный модуль обеспечивает прямое взаимодействие rsyslog с `systemd-journald` и позволяет получать сообщения из бинарного журнала `systemd` без промежуточных файлов.

2. Как называется устаревший модуль, который можно использовать для включения приёма сообщений журнала в rsyslog?

Устаревшим модулем для приёма сообщений журнала является `imuxsock`. Он осуществляет приём сообщений через UNIX-сокет `/dev/log`, однако не обеспечивает полного и корректного взаимодействия с `journald`, поэтому в современных системах рекомендуется использовать `imjournal`.

3. Чтобы убедиться, что устаревший метод приёма сообщений из journald в rsyslog не используется, какой дополнительный параметр следует использовать?

Для исключения использования устаревшего метода необходимо использовать параметр `SystemLogSocketName=` в конфигурации `journald`, установив его в пустое значение. Это предотвращает передачу сообщений через сокет `/dev/log` и исключает использование модуля `imuxsock`.

4. В каком конфигурационном файле содержатся настройки, которые позволяют вам настраивать работу журнала?

Основные настройки работы системного журнала `journald` содержатся в конфи-

густационном файле `/etc/systemd/journald.conf`. В нём задаются параметры хранения журналов, их ротации и пересылки сообщений в `rsyslog`.

5. Каким параметром управляется пересылка сообщений из `journald` в `rsyslog`?

Пересылка сообщений из `journald` в `rsyslog` управляется параметром `ForwardToSyslog` в файле `/etc/systemd/journald.conf`. При значении `yes` сообщения передаются в `rsyslog`, при `no` — пересылка отключается.

6. Какой модуль `rsyslog` вы можете использовать для включения сообщений из файла журнала, не созданного `rsyslog`?

Для чтения и обработки сообщений из файла журнала, не созданного `rsyslog`, используется модуль `imfile`. Он позволяет отслеживать произвольные текстовые файлы и включать их содержимое в систему журналирования.

7. Какой модуль `rsyslog` вам нужно использовать для пересылки сообщений в базу данных `MariaDB`?

Для пересылки сообщений в базу данных `MariaDB` используется модуль `ommysql`. Он обеспечивает запись сообщений `rsyslog` в таблицы базы данных `MySQL/MariaDB` для последующего анализа и хранения.

8. Какие две строки вам нужно включить в `rsyslog.conf`, чтобы позволить текущему журнальному серверу получать сообщения через `TCP`?

Для приёма сообщений через `TCP` необходимо загрузить модуль `TCP` и активировать серверный порт: - строка загрузки модуля приёма `TCP`-сообщений; - строка запуска `TCP`-сервера на порту 514.

Эти директивы позволяют `rsyslog` принимать сетевые журналы по `TCP`-протоколу.

9. Как настроить локальный брандмауэр, чтобы разрешить приём сообщений журнала через порт `TCP 514`?

Для разрешения приёма сообщений журнала необходимо открыть `TCP`-порт 514 в настройках межсетевого экрана. Порт добавляется в активную конфигурацию и фиксируется в постоянных правилах брандмауэра, чтобы разрешение сохранялось после перезагрузки системы.