

Отчёт по третьему разделу внешнего курса

Артём Дмитриевич Петлин

Содержание

1 Цель работы	5
2 Задание	6
3 Теоретическое введение	7
4 Выполнение практических заданий	8
5 Выполнение тестовых заданий	30
6 Оценки тестов	55
7 Выводы	72
Список литературы	73

Список иллюстраций

4.1 модуль 8	8
4.2 модуль 8	8
4.3 модуль 8	9
4.4 модуль 8	9
4.5 модуль 8	10
4.6 модуль 8	10
4.7 модуль 8	11
4.8 модуль 8	12
4.9 модуль 8	12
4.10 модуль 8	12
4.11 модуль 8	13
4.12 модуль 8	13
4.13 модуль 8	13
4.14 модуль 8	14
4.15 модуль 9	14
4.16 модуль 9	14
4.17 модуль 9	15
4.18 модуль 9	15
4.19 модуль 9	16
4.20 модуль 9	16
4.21 модуль 9	16
4.22 модуль 10	17
4.23 модуль 10	17
4.24 модуль 10	18
4.25 модуль 10	18
4.26 модуль 10	18
4.27 модуль 10	19
4.28 модуль 10	19
4.29 модуль 10	19
4.30 модуль 10	20
4.31 модуль 10	20
4.32 модуль 10	21
4.33 модуль 10	21
4.34 модуль 10	22
4.35 модуль 10	23
4.36 модуль 10	24

4.37 модуль 10	24
4.38 модуль 11	25
4.39 модуль 11	25
4.40 модуль 11	25
4.41 модуль 11	26
4.42 модуль 11	26
4.43 модуль 11	26
4.44 модуль 11	27
4.45 модуль 11	27
4.46 модуль 11	27
4.47 модуль 11	27
4.48 модуль 11	28
4.49 модуль 11	28
4.50 модуль 11	28
4.51 модуль 11	29
5.1 тест	30
5.2 тест	31
5.3 тест	32
5.4 тест	33
5.5 тест	34
5.6 тест	35
5.7 тест	36
5.8 тест	37
5.9 тест	38
5.10 тест	39
5.11 тест	40
5.12 тест	41
5.13 тест	42
5.14 тест	43
5.15 тест	44
5.16 тест	45
5.17 тест	46
5.18 тест	47
5.19 тест	48
5.20 тест	49
5.21 тест	50
5.22 тест	51
5.23 тест	52
5.24 тест	53
5.25 тест	54
6.1 тест 1	55

6.2 тест 2	56
6.3 тест 3	57
6.4 тест 4	58
6.5 тест 5	59
6.6 тест 6	60
6.7 тест 7	61
6.8 тест 8	62
6.9 тест 9	63
6.10 тест 10	64
6.11 тест 11	65
6.12 тест 12	66
6.13 тест 13	67
6.14 тест 14	68
6.15 тест 15	69
6.16 тест 16	70
6.17 тест 17	71

Список таблиц

1 Цель работы

Выполнить третий раздел внешнего курса «Системный администратор Linux с нуля».

2 Задание

Задания восьмого, девятого, десятого и одиннадцатого модулей, а также тесты.

3 Теоретическое введение

- Модуль 8. Настройка сети и SSH
- Модуль 9: Управление пакетами
- Модуль 10: Управление логами
- Модуль 10: Управление логами

4 Выполнение практических заданий

```
adpetlin@adpetlin:~$ sudo ip a add 192.168.122.2/24 dev enp0s3
RTNETLINK answers: File exists
adpetlin@adpetlin:~$
```

Рисунок 4.1: модуль 8

Настраиваем статический IP на тестовом сервере вручную.

```
auto enp0s3
iface enp0s3 inet static
    address 192.168.122.2
    netmask 255.255.255.0
```

Рисунок 4.2: модуль 8

В файле /etc/network/interfaces прописываем конфигурацию для интерфейса eth0.

```
adpetlin@adpetlin:~$ sudo ip a add 192.168.122.2/24 dev enp0s3
RTNETLINK answers: File exists
adpetlin@adpetlin:~$ su -
Пароль:
root@adpetlin:~# ping selectel.ru
PING selectel.ru (85.119.149.3) 56(84) bytes of data.
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=1 ttl=255 time=9.75 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=2 ttl=255 time=8.08 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=3 ttl=255 time=10.3 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=4 ttl=255 time=5.92 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=5 ttl=255 time=9.41 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=6 ttl=255 time=13.4 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=7 ttl=255 time=8.71 ms
64 bytes from 85.119.149.3 (85.119.149.3): icmp_seq=8 ttl=255 time=13.0 ms
^C
--- selectel.ru ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7137ms
rtt min/avg/max/mdev = 5.921/9.814/13.357/2.299 ms
root@adpetlin:~# ip a del 192.168.122.2/24 dev enp0s3
root@adpetlin:~#
```

Рисунок 4.3: модуль 8

Проверяем доступность внешних ресурсов через ping.

```
adpetlin@adpetlin:~$ sudo systemctl start nginx
adpetlin@adpetlin:~$ ss -tulnp | grep :80
tcp  LISTEN 0      511          0.0.0.0:80          0.0.0.0:*
tcp  LISTEN 0      511          [::]:80            [::]:*
adpetlin@adpetlin:~$ nc -vz 192.168.122.2 80
nc: connect to 192.168.122.2 port 80 (tcp) failed: Connection refused
adpetlin@adpetlin:~$ nc -vz 192.168.122.1 80
nc: connect to 192.168.122.1 port 80 (tcp) failed: Connection refused
adpetlin@adpetlin:~$ nc -vz 0.0.0.0 80
Connection to 0.0.0.0 80 port [tcp/http] succeeded!
adpetlin@adpetlin:~$ _
```

Рисунок 4.4: модуль 8

Удаляем текущий IP-адрес с интерфейса. Запускаем любой веб-сервер на 80 порту (например, nginx) и проверяем, что он работает.

```
adpetlin@adpetlin:~$ dig vk.com
;; communications error to 10.0.2.3#53: timed out
;; communications error to 10.0.2.3#53: timed out
;; communications error to 10.0.2.3#53: timed out

; <>> DiG 9.18.28-1~deb12u2-Debian <>> vk.com
;; global options: +cmd
;; no servers could be reached

adpetlin@adpetlin:~$ dig google.com
;; communications error to 10.0.2.3#53: timed out
;; communications error to 10.0.2.3#53: timed out
;; communications error to 10.0.2.3#53: timed out

; <>> DiG 9.18.28-1~deb12u2-Debian <>> google.com
;; global options: +cmd
;; no servers could be reached

adpetlin@adpetlin:~$ dig selectel.ru
;; communications error to 10.0.2.3#53: timed out
;; communications error to 10.0.2.3#53: timed out
;; communications error to 10.0.2.3#53: timed out

; <>> DiG 9.18.28-1~deb12u2-Debian <>> selectel.ru
;; global options: +cmd
;; no servers could be reached

adpetlin@adpetlin:~$
```

Рисунок 4.5: модуль 8

С помощью утилиты dig узнаем IP-адреса популярных сервисов.

```
adpetlin@adpetlin:~$ ss -tuln | grep :22
tcp  LISTEN 0      128          0.0.0.0:22          0.0.0.0:*
tcp  LISTEN 0      128          [::]:22            [::]:*
adpetlin@adpetlin:~$
```

Рисунок 4.6: модуль 8

Проверяем, слушает ли SSH-порт на сервере.

```
Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin no
```

Рисунок 4.7: модуль 8

Изменяем порт SSH и запрещаем вход по паролю.

```
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/adpetlin/.ssh/id_ed25519):
Created directory '/home/adpetlin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/adpetlin/.ssh/id_ed25519
Your public key has been saved in /home/adpetlin/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:xwSgX63bN8qspnjQJ+b508zRtruKfke7b6x2p+4aJ94 adpetlin@adpetlin
The key's randomart image is:
+-- [ED25519 256] --+
|   ...
|   . o
|   . . o
|   . . +
|   .. S o.
|   . + .+. +
|   + +.+.=+.
|   .+ o+=+=**+
|   ...==+=+8%Eo
+--- [SHA256] ---+
```

Рисунок 4.8: модуль 8

Настраиваем подключение по ключу и авторизуемся.

```
root@adpetlin:~# ssh -p 2222 user1@192.168.122.2
ssh: connect to host 192.168.122.2 port 2222: Connection refused
root@adpetlin:~# ssh -p 2222 user1@192.168.122.0
ssh: connect to host 192.168.122.0 port 2222: Connection refused
root@adpetlin:~# ssh -p 2222 user1@192.168.122.1
ssh: connect to host 192.168.122.1 port 2222: Connection refused
root@adpetlin:~# _
```

Рисунок 4.9: модуль 8

Разрешаем SSH-доступ только для определенных пользователей.

Убеждаемся, что для другого пользователя подключение не сработает.

```
root@adpetlin:~# ssh -p 2222 trex@192.168.122.222
ssh: connect to host 192.168.122.222 port 2222: Connection refused
root@adpetlin:~# ssh -p 2222 user2@192.168.122.222
ssh: connect to host 192.168.122.222 port 2222: Connection refused
root@adpetlin:~# _
```

Рисунок 4.10: модуль 8

Настраиваем UFW для ограничения доступа только для SSH.

```
adpetlin@adpetlin:~$ ufw deny ssh
Команда 'ufw' доступна в следующих местах
* /sbin/ufw
* /usr/sbin/ufw
Команда не может быть найдена, потому что '/usr/sbin:/sbin' не включена в переменную окружения PATH
Вероятно, причиной является отсутствие прав администратора у вашей учетной записи.
ufw: команда не найдена
adpetlin@adpetlin:~$ su -
Пароль:
root@adpetlin:~# ufw deny ssh
Skipping adding existing rule
Skipping adding existing rule (v6)
root@adpetlin:~# ufw allow from 192.168.122.2/24 to any port 22 proto tcp
WARN: Rule changed after normalization
Skipping adding existing rule
root@adpetlin:~#
```

Рисунок 4.11: модуль 8

Настраиваем UFW, чтобы доступ по SSH был только из вашей локальной сети.

```
/etc/fail2ban/filter.d/nginx-req-limit.conf  [----] 14 L:[ 1+ 2
:[Definition]
failregex = ^<HOST> -.*"(GET|POST).*" 200
ignoreregex =._
```

Рисунок 4.12: модуль 8

```
/etc/fail2ban/jail.d/nginx.conf  [----] 13 L:[ 1+ 7   8/  8] *(152 / 152b) <EOF>
[nginx-req-limit]
enabled = true
port = http,https
filter = nginx-req-limit
logpath = /var/log/nginx/access.log
maxretry = 3
findtime = 60
bantime = 600_
```

Рисунок 4.13: модуль 8

Используя fail2ban, создаем jail, который будет блокировать IP-адрес после трех успешных запросов к nginx.

```
root@adpetlin:~# systemctl restart fail2ban.service
root@adpetlin:~# _
```

Рисунок 4.14: модуль 8

Проверяем, как работает блокировка при множественных запросах.

```
adpetlin@adpetlin:~$ sudo apt update
[sudo] пароль для adpetlin:
Сущ:1 https://stable.see.selectel.ru selectos InRelease
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Может быть обновлено 93 пакета. Запустите «apt list --upgradable» для их показа.
adpetlin@adpetlin:~$ sudo apt install htop
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Уже установлен пакет htop самой новой версии (3.2.2-2).
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 93 пакетов не обновлено.
adpetlin@adpetlin:~$ _
```

Рисунок 4.15: модуль 9

Находим и устанавливаем утилиту htop.

```
adpetlin@adpetlin:~$ sudo apt purge htop
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Следующие пакеты устанавливались автоматически и больше не требуются:
 libnl-3-200 libnl-genl-3-200 libunwind8
Для их удаления используйте «sudo apt autoremove».
Следующие пакеты будут УДАЛЕНЫ:
   htop*
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 1 пакетов,
После данной операции объём занятого дискового пространства уменьшится на 358 kB.
Хотите продолжить? [Д/н] у
(Чтение базы данных ... на данный момент установлено 44856 файлов и каталогов.)
Удаляется htop (3.2.2-2) ...
Обрабатываются триггеры для mailcap (3.70+nmu1) ...
Обрабатываются триггеры для man-db (2.11.2-2) ...
adpetlin@adpetlin:~$
```

Рисунок 4.16: модуль 9

Удаляем утилиту htop, затем устанавливаем заново с полной очисткой (purge).

```
adpetlin@adpetlin:~$ sudo apt autoremove --purge
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Следующие пакеты будут УДАЛЕНЫ:
  libnl-3-200* libnl-genl-3-200* libunwind8*
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 3 пакетов,
После данной операции объём занятого дискового пространства уменьшится на 442 kB.
Хотите продолжить? [Д/Н] у
(Чтение базы данных ... на данный момент установлено 44845 файлов и каталогов.)
Удаляется libnl-genl-3-200:amd64 (3.7.0-0.2) ...
Удаляется libnl-3-200:amd64 (3.7.0-0.2) ...
Удаляется libunwind8:amd64 (1.6.2-3) ...
Обрабатываются триггеры для libc-bin (2.36-9+deb12u8) ...
(Чтение базы данных ... на данный момент установлено 44818 файлов и каталогов.)
Вычищаются файлы настройки пакета libnl-3-200:amd64 (3.7.0-0.2) ...
adpetlin@adpetlin:~$
```

Рисунок 4.17: модуль 9

Выполняем полное обновление системы.

```
Настраивается пакет libgssapi-krb5-2:amd64 (1.20.1-2+deb12u2) ...
Настраивается пакет groff-base (1.22.4-10) ...
Настраивается пакет libtirpc3:amd64 (1.3.3+ds-1) ...
Настраивается пакет iproute2 (6.1.0-3) ...
Настраивается пакет isc-dhcp-client (4.4.3-P1-2) ...
Настраивается пакет libnsl2:amd64 (1.3.0-2) ...
Настраивается пакет libpython3.11-stdlib:amd64 (3.11.2-6+deb12u4) ...
Настраивается пакет python3.11 (3.11.2-6+deb12u4) ...
Настраивается пакет libdevmapper1.02.1:amd64 (2:1.02.185-2) ...
Настраивается пакет dmsetup (2:1.02.185-2) ...
Настраивается пакет libcryptsetup12:amd64 (2:2.6.1-4~deb12u2) ...
Обрабатываются триггеры для debianutils (5.7-0.5~deb12u1) ...
Обрабатываются триггеры для install-info (6.8-6) ...
Обрабатываются триггеры для mailcap (3.70+nmui1) ...
Обрабатываются триггеры для initramfs-tools (0.142+deb12u1) ...
update-initramfs: Generating /boot/initrd.img-6.1.0-27-amd64
Обрабатываются триггеры для libc-bin (2.36-9+deb12u8) ...
Обрабатываются триггеры для systemd (252.31-1~deb12u1) ...
Обрабатываются триггеры для man-db (2.11.2-2) ...
Обрабатываются триггеры для ca-certificates (20250419) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
adpetlin@adpetlin:~$ sudo apt upgrade
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Расчёт обновлений... Готово
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.
adpetlin@adpetlin:~$
```

Рисунок 4.18: модуль 9

Настраиваем автоматическую установку обновлений безопасности.

```
root@adpetlin:~# exit
выход
адретлин@адретлин:~$ sudo apt install unattended-upgrades
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состояниях... Готово
E: Невозможно найти пакет unattended-upgrades
адретлин@адретлин:~$ sudo dpkg-reconfigure unattended-upgrades
dpkg-query: пакет «unattended-upgrades» не установлен, информация о нём недоступна
Для проверки файлов архивов используйте команду dpkg --info (dpkg-deb --info).
/usr/sbin/dpkg-reconfigure: Пакет unattended-upgrades не установлен
адретлин@адретлин:~$ sudo apt autoremove -y
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состояниях... Готово
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.
адретлин@адретлин:~$ _
```

Рисунок 4.19: модуль 9

Удаляем устаревшие и неиспользуемые зависимости.

```
root@adpetlin:~# wget http://ftp.us.debian.org/debian/pool/main/h/htop/htop_3.4.1-5_amd64.deb
--2025-11-15 00:00:40-- http://ftp.us.debian.org/debian/pool/main/h/htop/htop_3.4.1-5_amd64.deb
Распознаётся ftp.us.debian.org (ftp.us.debian.org)... 64.50.236.52, 208.80.154.139, 64.50.233.100, ...
Подключение к ftp.us.debian.org (ftp.us.debian.org)|64.50.236.52|:80... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: 171412 (167K) [application/vnd.debian.binary-package]
Сохранение в: «htop_3.4.1-5_amd64.deb»

htop_3.4.1-5_amd64.deb           100%[=====] 2025-11-15 00:00:41 (249 KB/s) - «htop_3.4.1-5_amd64.deb» сохранён [171412/171412]

root@adpetlin:~# sudo dpkg -i htop_3.4.1-5_amd64.deb
(Чтение базы данных ... на данный момент установлено 44877 файлов и каталогов.)
Подготовка к распаковке htop_3.4.1-5_amd64.deb ...
Распаковывается htop (3.4.1-5) на замену (3.2.2-2) ...
dpkg: зависимости пакетов не позволяют настроить пакет htop:
  htop зависит от libc6 (>= 2.38), однако:
    Версия libc6:amd64 в системе – 2.36-9+deb12u8.

dpkg: ошибка при обработке пакета htop (--install):
  проблемы зависимости – оставляем не настроенным
Обрабатываются триггеры для mailcap (3.70+nmu1) ...
Обрабатываются триггеры для man-db (2.11.2-2) ...
При обработке следующих пакетов произошли ошибки:
  htop
root@adpetlin:~# _
```

Рисунок 4.20: модуль 9

Устанавливаем пакет вручную через dpkg, предварительно скачав его с сайта.

```
root@adpetlin:~# dpkg -I htop_3.4.1-5_amd64.deb | grep Depends
  Depends: libc6 (>= 2.38), libncursesw6 (>= 6), libtinfo6 (>= 6)
root@adpetlin:~# _
```

Рисунок 4.21: модуль 9

Анализируем, какие зависимости потребуются.

```
Ноя 15 00:04:05 adpetlin (python3)[32525]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
Ноя 15 00:04:05 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:05 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
Ноя 15 00:04:05 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
Ноя 15 00:04:15 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 777.
Ноя 15 00:04:15 adpetlin (python3)[32526]: mydaemon.service: Failed to determine user credentials: No such process
Ноя 15 00:04:15 adpetlin (python3)[32526]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
Ноя 15 00:04:15 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:15 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
Ноя 15 00:04:15 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
Ноя 15 00:04:25 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 778.
Ноя 15 00:04:25 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:25 adpetlin (python3)[32534]: mydaemon.service: Failed to determine user credentials: No such process
Ноя 15 00:04:25 adpetlin (python3)[32534]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
Ноя 15 00:04:25 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:25 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
Ноя 15 00:04:25 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
Ноя 15 00:04:35 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 779.
Ноя 15 00:04:35 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:35 adpetlin (python3)[32535]: mydaemon.service: Failed to determine user credentials: No such process
Ноя 15 00:04:35 adpetlin (python3)[32535]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
Ноя 15 00:04:35 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:35 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
Ноя 15 00:04:35 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
Ноя 15 00:04:46 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 780.
Ноя 15 00:04:46 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:46 adpetlin (python3)[32537]: mydaemon.service: Failed to determine user credentials: No such process
Ноя 15 00:04:46 adpetlin (python3)[32537]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
Ноя 15 00:04:46 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
Ноя 15 00:04:46 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
Ноя 15 00:04:46 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
(ЕНД)
```

Рисунок 4.22: модуль 10

```
Ноя 14 19:05:01 adpetlin CRON[1081]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
Ноя 14 19:05:01 adpetlin CRON[1080]: pam_unix(cron:session): session closed for user root
Ноя 14 19:10:13 adpetlin systemd[1]: Starting sysstat-collect.service - system activity accounting tool...
Ноя 14 19:10:13 adpetlin systemd[1]: sysstat-collect.service: Deactivated successfully.
Ноя 14 19:10:13 adpetlin systemd[1]: Finished sysstat-collect.service - system activity accounting tool.
Ноя 14 19:13:18 adpetlin crontab[1121]: (root) BEGIN EDIT (root)
Ноя 14 19:13:49 adpetlin crontab[1121]: (root) REPLACE (root)
Ноя 14 19:13:49 adpetlin crontab[1121]: (root) END EDIT (root)
Ноя 14 19:13:58 adpetlin crontab[1128]: (root) BEGIN EDIT (root)
Ноя 14 19:14:01 adpetlin cron[538]: (root) RELOAD (crontabs/root)
Ноя 14 19:14:07 adpetlin crontab[1128]: (root) END EDIT (root)
Ноя 14 19:15:01 adpetlin CRON[1136]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
Ноя 14 19:15:01 adpetlin CRON[1137]: (root) CMD (command -v debian-sa1 > /dev/null && debian-sa1 1 1)
Ноя 14 19:15:01 adpetlin CRON[1136]: pam_unix(cron:session): session closed for user root
Ноя 14 19:17:01 adpetlin CRON[1173]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
Ноя 14 19:17:01 adpetlin CRON[1174]: (root) CMD (cd / && run-parts --report /etc/cron.hourly)
Ноя 14 19:17:01 adpetlin CRON[1173]: pam_unix(cron:session): session closed for user root
Ноя 14 19:20:06 adpetlin systemd[1]: Starting sysstat-collect.service - system activity accounting tool...
Ноя 14 19:20:06 adpetlin systemd[1]: sysstat-collect.service: Deactivated successfully.
Ноя 14 19:20:06 adpetlin systemd[1]: Finished sysstat-collect.service - system activity accounting tool.
(ЕНД)
```

Рисунок 4.23: модуль 10

Просматриваем содержимое основного системного журнала и журнала аутентификации, используя утилиту для постраничного просмотра.

```

root@adpetlin:~# journalctl -n 20
ноя 15 00:05:48 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
ноя 15 00:05:48 adpetlin (python3)[32554]: mydaemon.service: Failed to determine user credentials: No such process
ноя 15 00:05:48 adpetlin (python3)[32554]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such proc
ноя 15 00:05:48 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
ноя 15 00:05:48 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
ноя 15 00:05:48 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
ноя 15 00:05:58 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 787.
ноя 15 00:05:58 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
ноя 15 00:05:58 adpetlin (python3)[32555]: mydaemon.service: Failed to determine user credentials: No such process
ноя 15 00:05:58 adpetlin (python3)[32555]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such proc
ноя 15 00:05:58 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
ноя 15 00:05:58 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
ноя 15 00:05:58 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
ноя 15 00:06:08 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 788.
ноя 15 00:06:08 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
ноя 15 00:06:08 adpetlin (python3)[32556]: mydaemon.service: Failed to determine user credentials: No such process
ноя 15 00:06:08 adpetlin (python3)[32556]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such proc
ноя 15 00:06:08 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
ноя 15 00:06:08 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
ноя 15 00:06:08 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
root@adpetlin:~# 

```

Рисунок 4.24: модуль 10

Выводим последние 20 записей из бинарного системного журнала с помощью journalctl.

```

root@adpetlin:~# ls -l /var/log/nginx/error.log
-rw-r----- 1 www-data adm 76 ноя 14 21:21 /var/log/nginx/error.log
root@adpetlin:~# journalctl -n 1
ноя 15 00:08:52 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
root@adpetlin:~# 

```

Рисунок 4.25: модуль 10

Проверяем наличие и расположение файла журнала ошибок веб-сервера Nginx.

```

ноя 14 23:45:01 adpetlin CRON[32184]: pam_unix(cron:session): session closed for user root
ноя 14 23:55:01 adpetlin CRON[32331]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
ноя 14 23:55:01 adpetlin CRON[32332]: (root) CMD (command -v debian-sai > /dev/null && debian-sai 1 1)
ноя 14 23:55:01 adpetlin CRON[32331]: pam_unix(cron:session): session closed for user root
ноя 14 23:59:01 adpetlin CRON[32367]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
ноя 14 23:59:01 adpetlin CRON[32368]: (root) CMD (command -v debian-sai > /dev/null && debian-sai 60 2)
ноя 14 23:59:01 adpetlin CRON[32367]: pam_unix(cron:session): session closed for user root
ноя 15 00:00:01 adpetlin CRON[32386]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
ноя 15 00:00:01 adpetlin CRON[32387]: (root) CMD ([ -d "/run/systemd/system" ] || /usr/share/atop/atop.daily&)
ноя 15 00:00:01 adpetlin CRON[32386]: pam_unix(cron:session): session closed for user root
ноя 15 00:05:01 adpetlin CRON[32542]: pam_unix(cron:session): session opened for user root(uid=0) by (uid=0)
ноя 15 00:05:01 adpetlin CRON[32543]: (root) CMD (command -v debian-sai > /dev/null && debian-sai 1 1)
ноя 15 00:05:01 adpetlin CRON[32542]: pam_unix(cron:session): session closed for user root
Lines 146-194/194 (END).

```

Рисунок 4.26: модуль 10

Изучаем любую одну строку из файла /var/log/auth.log и вручную определяем в ней: временную метку, имя сервиса (процесса) и описание события.

```
root@adpetlin:~# systemctl status rsyslog.service
Unit rsyslog.service could not be found.
root@adpetlin:~# systemctl status systemd-journald.service
● systemd-journald.service - Journal Service
   Loaded: loaded (/lib/systemd/system/systemd-journald.service; static)
     Active: active (running) since Fri 2025-11-14 21:51:12 MSK; 2h 22min ago
TriggeredBy: • systemd-journald-audit.socket
              • systemd-journald-dev-log.socket
              • systemd-journald.socket
    Docs: man:systemd-journald.service(8)
          man:journald.conf(5)
   Main PID: 251 (systemd-journal)
      Status: "Processing requests..."
        Tasks: 1 (limit: 4652)
       Memory: 18.0M
         CPU: 944ms
      CGroup: /system.slice/systemd-journald.service
               └─251 /lib/systemd/systemd-journald

ноя 14 21:51:12 adpetlin systemd-journald[251]: Journal started
ноя 14 21:51:12 adpetlin systemd-journald[251]: Runtime Journal (/run/log/journal/10afde83d1d74c48857b6c7f78f2523b) is 4
ноя 14 21:51:12 adpetlin systemd-journald[251]: Time spent on flushing to /var/log/journal/10afde83d1d74c48857b6c7f78f2523b is 72
ноя 14 21:51:12 adpetlin systemd-journald[251]: System Journal (/var/log/journal/10afde83d1d74c48857b6c7f78f2523b) is 72
ноя 14 21:51:12 adpetlin systemd-journald[251]: Received client request to flush runtime journal.
Notice: journal has been rotated since unit was started, output may be incomplete.
root@adpetlin:~#
```

Рисунок 4.27: модуль 10

Проверяем текущий статус служб rsyslog и systemd-journald с помощью systemctl.

```
root@adpetlin:~# logger "Это тестовое сообщение"
root@adpetlin:~# journalctl -n 1
ноя 15 00:15:04 adpetlin root[32741]: Это тестовое сообщение
root@adpetlin:~# _
```

Рисунок 4.28: модуль 10

```
root@adpetlin:~# grep "Это тестовое сообщение" /var/log/syslog
ноя 15 00:15:04 adpetlin root[32741]: Это тестовое сообщение
root@adpetlin:~# grep 'auth.log' /etc/rsyslog.d/50-default.conf
grep: /etc/rsyslog.d/50-default.conf: Нет такого файла или каталога
root@adpetlin:~# logger -p user.warn "Это специальное тестовое сообщение"
root@adpetlin:~# journalctl -p warning | grep "специальное тестовое сообщение"
ноя 15 00:17:34 adpetlin root[32777]: Это специальное тестовое сообщение
root@adpetlin:~# _
```

Рисунок 4.29: модуль 10

Генерируем тестовое сообщение с помощью утилиты logger. Затем находим это сообщение сначала в выводе journalctl, а потом в файле /var/log/syslog. Используя journalctl, фильтруем и выводим только те события, которые имеют уровень важности error (err) или более критичный. Заглядываем в

конфигурационный файл rsyslog (например, /etc/rsyslog.d/50-default.conf) и находим строку, отвечающую за направление сообщений от категорий (facility) auth и authpriv в файл /var/log/auth.log. Отправляем в системный журнал сообщение с явно указанным уровнем важности warning. Убеждаемся, что оно появляется в выводе journalctl при фильтрации по этому уровню.

```
root@adpetlin:~# grep -i "Failed password" /var/log/auth.log
ноя 14 23:17:49 adpetlin sshd[3062]: Failed password for invalid user user1 from 192.168.122.2 port 39086 ssh2
root@adpetlin:~# grep "Accepted password" /var/log/auth.log | awk '{print $11}'
root@adpetlin:~# grep "Failed password" /var/log/auth.log | awk '{print $11}' | sort | uniq
user1
root@adpetlin:~#
```

Рисунок 4.30: модуль 10

Находим все строки в файле /var/log/auth.log, в которых упоминается неудачная попытка входа по паролю (Failed password), не обращая внимания на регистр символов. Строим конвейер команд, который сначала найдет все успешные SSH-подключения (Accepted password) в /var/log/auth.log, а затем извлечет из этих строк только IP-адреса подключившихся клиентов. Составляем список уникальных IP-адресов, с которых были зафиксированы неудачные попытки входа в систему. Собираем статистику и выведите пять IP-адресов, с которых было зафиксировано наибольшее количество успешных входов по SSH.

```
ноя 15 00:24:25 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
ноя 15 00:24:25 adpetlin (python3)[32853]: mydaemon.service: Failed to determine user credentials: No such process
ноя 15 00:24:25 adpetlin (python3)[32853]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
ноя 15 00:24:25 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
ноя 15 00:24:25 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
ноя 15 00:24:25 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
ноя 15 00:24:36 adpetlin systemd[1]: mydaemon.service: Scheduled restart job, restart counter is at 896.
ноя 15 00:24:36 adpetlin systemd[1]: Stopped mydaemon.service - Мой тестовый сервис.
ноя 15 00:24:36 adpetlin (python3)[32856]: mydaemon.service: Failed to determine user credentials: No such process
ноя 15 00:24:36 adpetlin (python3)[32856]: mydaemon.service: Failed at step USER spawning /usr/bin/python3: No such process
ноя 15 00:24:36 adpetlin systemd[1]: Started mydaemon.service - Мой тестовый сервис.
ноя 15 00:24:36 adpetlin systemd[1]: mydaemon.service: Main process exited, code=exited, status=217/USER
ноя 15 00:24:36 adpetlin systemd[1]: mydaemon.service: Failed with result 'exit-code'.
root@adpetlin:~# grep "Accepted password" /var/log/auth.log | awk '{print $11}' | sort | uniq -c | sort -nr | head -n 5
root@adpetlin:~#
```

Рисунок 4.31: модуль 10

Используя journalctl, отображаем все системные журналы за последние 15 минут.

```

root@adpetlin:~# grep "Invalid user" /var/log/auth.log
ноя 14 22:40:06 adpetlin sshd[1432]: Invalid user user1 from 127.0.0.1 port 41544
ноя 14 22:41:27 adpetlin sshd[1453]: Invalid user user1 from 127.0.0.1 port 51282
ноя 14 23:02:47 adpetlin sshd[2301]: Invalid user user1 from 192.168.122.2 port 56812
ноя 14 23:02:47 adpetlin sshd[2308]: Invalid user user1 from 192.168.122.2 port 56818
ноя 14 23:02:48 adpetlin sshd[2321]: Invalid user user1 from 192.168.122.2 port 56826
ноя 14 23:03:11 adpetlin sshd[2330]: Invalid user user1 from 192.168.122.2 port 45466
ноя 14 23:03:31 adpetlin sshd[2357]: Invalid user user1 from 192.168.122.2 port 40186
ноя 14 23:03:31 adpetlin sshd[2365]: Invalid user user1 from 192.168.122.2 port 40190
ноя 14 23:03:31 adpetlin sshd[2378]: Invalid user user1 from 192.168.122.2 port 40204
ноя 14 23:03:34 adpetlin sshd[2382]: Invalid user user1 from 192.168.122.2 port 50602
ноя 14 23:05:19 adpetlin sshd[2456]: Invalid user user1 from 192.168.122.2 port 47978
ноя 14 23:05:20 adpetlin sshd[2463]: Invalid user user1 from 192.168.122.2 port 47982
ноя 14 23:05:20 adpetlin sshd[2476]: Invalid user user1 from 192.168.122.2 port 47984
ноя 14 23:06:01 adpetlin sshd[2603]: Invalid user user1 from 192.168.122.2 port 42052
ноя 14 23:06:01 adpetlin sshd[2610]: Invalid user user1 from 192.168.122.2 port 42058
ноя 14 23:06:01 adpetlin sshd[2623]: Invalid user user1 from 192.168.122.2 port 42060
ноя 14 23:08:07 adpetlin sshd[2653]: Invalid user user1 from 192.168.122.2 port 49358
ноя 14 23:08:51 adpetlin sshd[2670]: Invalid user user1 from 192.168.122.2 port 50360
ноя 14 23:09:41 adpetlin sshd[2703]: Invalid user user1 from 192.168.122.2 port 59788
ноя 14 23:09:41 adpetlin sshd[2710]: Invalid user user1 from 192.168.122.2 port 59804
ноя 14 23:09:41 adpetlin sshd[2723]: Invalid user user1 from 192.168.122.2 port 59816
ноя 14 23:10:45 adpetlin sshd[2757]: Invalid user user1 from 192.168.122.2 port 60218
ноя 14 23:10:45 adpetlin sshd[2766]: Invalid user user1 from 192.168.122.2 port 49548
ноя 14 23:10:45 adpetlin sshd[2779]: Invalid user user1 from 192.168.122.2 port 49550
ноя 14 23:10:55 adpetlin sshd[2784]: Invalid user user1 from 192.168.122.2 port 53492
ноя 14 23:11:10 adpetlin sshd[2788]: Invalid user user1 from 192.168.122.2 port 44496
ноя 14 23:11:15 adpetlin sshd[2792]: Invalid user user1 from 192.168.122.2 port 36278
ноя 14 23:12:49 adpetlin sshd[2830]: Invalid user user1 from 192.168.122.2 port 38210
ноя 14 23:12:49 adpetlin sshd[2837]: Invalid user user1 from 192.168.122.2 port 38226
ноя 14 23:12:49 adpetlin sshd[2850]: Invalid user user1 from 192.168.122.2 port 38232
ноя 14 23:13:17 adpetlin sshd[2882]: Invalid user trex from 192.168.122.2 port 36440
ноя 14 23:13:17 adpetlin sshd[2889]: Invalid user trex from 192.168.122.2 port 36448
ноя 14 23:13:17 adpetlin sshd[2902]: Invalid user trex from 192.168.122.2 port 36450
ноя 14 23:13:40 adpetlin sshd[2908]: Invalid user trex from 192.168.122.2 port 32772
ноя 14 23:14:20 adpetlin sshd[2942]: Invalid user user1 from 192.168.122.2 port 35986
ноя 14 23:14:20 adpetlin sshd[2949]: Invalid user user1 from 192.168.122.2 port 35988
ноя 14 23:14:21 adpetlin sshd[2962]: Invalid user user1 from 192.168.122.2 port 36002
ноя 14 23:17:32 adpetlin sshd[3042]: Invalid user user1 from 192.168.122.2 port 39060
ноя 14 23:17:32 adpetlin sshd[3049]: Invalid user user1 from 192.168.122.2 port 39070
ноя 14 23:17:32 adpetlin sshd[3062]: Invalid user user1 from 192.168.122.2 port 39086
root@adpetlin:~#

```

Рисунок 4.32: модуль 10

Находим в журнале аутентификации /var/log/auth.log все попытки входа от имени несуществующих пользователей.

```

root@adpetlin:~# grep "Failed password" /var/log/auth.log | awk '{print $11}' | sort | uniq -c | sort -nr | head -n 1
    1 user1
root@adpetlin:~# grep "192.168.122.2" /var/log/auth.log > incident_report.log
root@adpetlin:~# sha256sum incident_report.log
cfec6ac47fad8626b61d5abc0ec0c017a0ccc676bcadb9e5b272654e12c35cca  incident_report.log
root@adpetlin:~# grep "sudo:" /var/log/auth.log | grep "USER=$(whoami)"
root@adpetlin:~# 

```

Рисунок 4.33: модуль 10

Определяем IP-адрес, с которого было совершено наибольшее количество неудачных попыток подбора пароля (Failed password). После определения

наиболее подозрительного IP-адреса из предыдущего шага, извлекаем из /var/log/auth.log абсолютно все записи, связанные с этим IP, и сохраняем их в отдельный файл incident_report.log. Рассчитываем контрольную сумму SHA256 для созданного файла incident_report.log, чтобы зафиксировать его целостность и доказать, что он не изменялся после сбора. Проверяем, какие команды выполнялись с правами суперпользователя (через sudo) вашим текущим пользователем.

```
root@adpetlin:~# cat /etc/logrotate.d/apt
/var/log/apt/term.log {
    rotate 12
    monthly
    compress
    missingok
    notifempty
}

/var/log/apt/history.log {
    rotate 12
    monthly
    compress
    missingok
    notifempty
}

root@adpetlin:~# _
```

Рисунок 4.34: модуль 10

Изучаем существующую конфигурацию logrotate для системного менеджера пакетов (apt или dpkg) в директории /etc/logrotate.d/. Определяем, как часто происходит ротация, сколько архивных копий хранится и используется ли сжатие.

```
GNU nano 7.2
/var/log/testapp.log {
    daily
    rotate 4
    compress
    missingok
    notifempty
}
:
```

Рисунок 4.35: модуль 10

Создаем собственный конфигурационный файл /etc/logrotate.d/testapp для управления вымышленным лог-файлом /var/log/testapp.log. Настраиваем его на ежедневную ротацию, хранение четырех архивных копий и сжатие старых логов.

```
root@adpetlin:~# logrotate -d /etc/logrotate.d/testapp
warning: logrotate in debug mode does nothing except printing debug messages! Consider using verbose mode (-v) instead

reading config file /etc/logrotate.d/testapp
Reading state from file: /var/lib/logrotate/status
Allocating hash table for state file, size 64 entries
Creating new state
Handling 1 logs

rotating pattern: /var/log/testapp.log after 1 days (4 rotations)
empty log files are not rotated, old logs are removed
considering log /var/log/testapp.log
Creating new state
    Now: 2025-11-15 00:41
    Last rotated at 2025-11-15 00:00
    log does not need rotating (log has already been rotated)
root@adpetlin:~# _
```

Рисунок 4.36: модуль 10

Проверяем созданную конфигурацию logrotate на синтаксические ошибки, выполнив «сухой запуск» в режиме отладки.

```
root@adpetlin:~# sudo logrotate -f /etc/logrotate.d/testapp
root@adpetlin:~# ls -l /var/log/testapp*
-rw-r--r-- 1 root root 0 ноя 15 00:39 /var/log/testapp.log
root@adpetlin:~# *.* @@logs.example.com:514
archive_test.tar.gz: команда не найдена
root@adpetlin:~#
```

Рисунок 4.37: модуль 10

Принудительно запускаем ротацию для лога testapp и проверьте результат: убеждаемся, что старый лог был сжат и переименован, а на его месте появился новый пустой файл. Пишем строку конфигурации для rsyslog, которая будет пересыпать абсолютно все логи (.) по протоколу TCP на удаленный сервер с адресом logs.example.com и стандартным портом 514.

```
adpetlin@adpetlin:~$ sudo apt install podman
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состояниях... Готово
Уже установлен пакет podman самой новой версии (4.3.1+ds1-8+deb12u1).
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 197 пакетов не обновлено.
adpetlin@adpetlin:~$
```

Рисунок 4.38: модуль 11

Устанавливаем Podman и проверяем установку.

```
adpetlin@adpetlin:~$ podman run -it --rm alpine sh
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `loginctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `loginctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
Resolved "alpine" as an alias (/etc/containers/registries.conf.d/shortnames.conf)
Trying to pull docker.io/library/alpine:latest...
Getting image source signatures
Copying blob 2d35ebdb57d9 done
Copying config 706db57fb2 done
Writing manifest to image destination
Storing signatures
[10679.128824] BPF: [100413] STRUCT
[10679.128838] BPF: size=4 vlen=2
[10679.128843] BPF:
[10679.128846] BPF: Invalid name
[10679.128850] BPF:
/ #
```

Рисунок 4.39: модуль 11

```
adpetlin@adpetlin:~$ podman run -it --rm debian
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `loginctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `loginctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
Resolved "debian" as an alias (/etc/containers/registries.conf.d/shortnames.conf)
Trying to pull docker.io/library/debian:latest...
Getting image source signatures
Copying blob 13cc39f8244a done
Copying config 3ab615c1937 done
Writing manifest to image destination
Storing signatures
root@a0h18687982c:/#
```

Рисунок 4.40: модуль 11

Запускаем контейнер.

```
adpetlin@adpetlin:~$ podman run -d --name my-nginx -p 8080:80 docker.io/library/nginx
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
Trying to pull docker.io/library/nginx:latest...
Getting image source signatures
Copying blob 52bc359bcb7 done
Copying blob 266626526d42 done
Copying blob d7ecded7702a done
Copying blob d921c57c6a81 done
Copying blob 320b0949be89 done
Copying blob 9def9e3993e4 done
Copying blob e2f8e296d9df done
Copying config d261fd19cb done
Writing manifest to image destination
Storing signatures
6b10c8631706ab6942ea471f4e4eed8dc00e1aa188db432f70841a5dfa86dc8
adpetlin@adpetlin:~$ _
```

Рисунок 4.41: модуль 11

Запускаем Nginx и проверяем, что он работает.

```
adpetlin@adpetlin:~$ podman run -d --name my-nginx -p 8080:80 -v /home/adpetlin/site:/usr/share/nginx/html:ro docker.io/
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
0edcc2e7c30dbf0db0142ea2435253bd3218b8ddf9a6a7235a48e62c62d79a94
adpetlin@adpetlin:~$ curl http://localhost:8080
curl: (3) URL using bad/illegal format or missing URL
adpetlin@adpetlin:~$ curl http://localhost:8080
<h1>Hello from Podman</h1>
adpetlin@adpetlin:~$ _
```

Рисунок 4.42: модуль 11

Разворачиваем сайт в Nginx, используя локальную папку с HTML-файлами.

ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET IO	BLOCK IO	PIDS	CPU TIME
5a9057f69742	nginx-limited	733.18%	30.72MB / 104.9MB	29.30%	430B / 110B	0B / 0B	28	2m0.695512s

Рисунок 4.43: модуль 11

Запускаем контейнер Nginx с ограничением памяти в 100 МБ и проверяем его работу.

```

adpetlin@adpetlin:~$ podman run --memory=100m --name stress-test docker.io/programm/stress --vm 1 --vm-bytes 150m --vm-hang 0
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroups
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
Trying to pull docker.io/programm/stress:latest...
Getting image source signatures
Copying blob 7d04afe1405 done
Copying blob a3ed95caebe2 done
Copying blob 871c32dbbb53 done
Copying blob d14088925c56 done
Copying blob doe7819a64dd done
Copying blob 58026d51efc4 done
Copying blob 1775fca35fb6 done
Copying blob 5c819e267900 done
Copying blob 1775fca35fb6 done
Copying blob 1775fca35fb6 done
Writing manifest to image destination
Storing signatures
[Cadpetlin@adpetlin:~$]
adpetlin@adpetlin:~$ podman logs stress-test
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroups
adpetlin@adpetlin:~$
```

Рисунок 4.44: модуль 11

ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET IO	BLOCK IO	PIDS	CPU TIME	Avg C
434814c98ff7	stress-test	48.00%	214.2MB / 104.9MB	204.27%	430B / 220B	0B / 0B	43	2m23.602133s	48.00

Рисунок 4.45: модуль 11

```

Error: no container with name or ID "stress-test" found: no such container
adpetlin@adpetlin:~$ journalctl -u podman --no-pager | grep OOM
Hint: You are currently not seeing messages from other users and the system.
      Users in groups 'adm', 'systemd-journal' can see all messages.
      Pass -q to turn off this notice.
adpetlin@adpetlin:~$ _
```

Рисунок 4.46: модуль 11

Исследуем поведение контейнера при исчерпании выделенной памяти.

```

adpetlin@adpetlin:~$ echo '<h1>Hello from my custom container!</h1>' > index.html
adpetlin@adpetlin:~$ ls
```

Рисунок 4.47: модуль 11

```
GNU nano 7.2
FROM nginx
COPY ./index.html /usr/share/nginx/html/index.html
```

Рисунок 4.48: модуль 11

```
adpetlin@adpetlin:~$ podman build -t mysite1 .
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
STEP 1/2: FROM nginx
STEP 2/2: COPY ./index.html /usr/share/nginx/html/index.html
COMMIT mysite1
--> fd27b46fea0
Successfully tagged localhost/mysite1:latest
fd27b46fea06811841ad46ca0d7644b35979bee550acf361071ed1345ec96d7c
```

Рисунок 4.49: модуль 11

```
adpetlin@adpetlin:~$ podman run -d -p 8080:80 mysite1
Error: requires at least 1 arg(s), only received 0
adpetlin@adpetlin:~$ podman run -d -p 8080:80 mysite1
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupfs
3eec3bab2da59bc9074121b1f7ba00a36524830b3f506a98f6ebbb1a44a4e4fe0
adpetlin@adpetlin:~$ curl http://localhost:8080
<h1>Hello from my custom container!</h1>
adpetlin@adpetlin:~$ _
```

Рисунок 4.50: модуль 11

Создаем и запускаем кастомный веб-сайт на базе nginx с использованием Podman.

```
adpetlin@adpetlin:~$ podman save -o mysite.tar mysite1
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupufs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupufs
Copying blob 36d06fe0cbc6 done
Copying blob 6e19587ac541 done
Copying blob 8feb164cd673 done
Copying blob 2ced4cd78a7b done
Copying blob 99cd1b1b6a43 done
Copying blob d81dff94f8d07 done
Copying blob d7217c60dca4 done
Copying blob 76b348108d34 done
Copying config fd27b46fea done
Writing manifest to image destination
Storing signatures
adpetlin@adpetlin:~$ scp mysite.tar user@remote:/tmp/
ssh: Could not resolve hostname remote: Name or service not known
scp: Connection closed
adpetlin@adpetlin:~$ scp mysite.tar user1@192.168.122.2:/tmp/
ssh: connect to host 192.168.122.2 port 22: Connection refused
scp: Connection closed
adpetlin@adpetlin:~$ scp mysite.tar user1@0.0.0.0:/tmp/
ssh: connect to host 0.0.0.0 port 22: Connection refused
scp: Connection closed
adpetlin@adpetlin:~$ scp -p 22 mysite.tar adpetlin@0.0.0.0:/tmp/
ssh: connect to host 0.0.0.0 port 22: Connection refused
scp: Connection closed
adpetlin@adpetlin:~$ podman run -d -p 8080:80 localhost/mysite1
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupufs
WARN[0000] The cgroupv2 manager is set to systemd but there is no systemd user session available
WARN[0000] For using systemd, you may need to login using an user session
WARN[0000] Alternatively, you can enable lingering with: `logindctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroupufs
Error: rootlessport listen tcp 0.0.0.0:8080: bind: address already in use
adpetlin@adpetlin:~$ curl http://localhost:8080
<h1>Hello from my custom container!</h1>
adpetlin@adpetlin:~$ _
```

Рисунок 4.51: модуль 11

Сохраняем образ и переносим его на другой сервер.

5 Выполнение тестовых заданий

Какой командой можно назначить IP-адрес вручную?

- а) ip add dev eth0 192.168.122.2
- б) ip a add 192.168.122.2/24 dev eth0
- в) ip set eth0 192.168.122.2
- г) ip config dev eth0 address 192.168.122.2

Верный ответ: ip a add 192.168.122.2/24 dev eth0

Что делает директива auto eth0 в /etc/network/interfaces?

- а) Запускает dhclient при подключении интерфейса
- б) Настраивает интерфейс при появлении линка
- в) Поднимает интерфейс автоматически при загрузке системы
- г) Назначает статический IP при старте

Верный ответ: Поднимает интерфейс автоматически при загрузке системы

Какая команда используется для удаления IP с интерфейса?

- а) ip addr flush dev eth0
- б) ip del addr 192.168.122.2 dev eth0
- в) ip a del 192.168.122.2/24 dev eth0
- г) ip a down 192.168.122.2 dev eth0

Верный ответ: ip a del 192.168.122.2/24 dev eth0

Рисунок 5.1: тест

Мы используем команду ip a add для временного добавления IP-адреса к сетевому интерфейсу. Директива auto в файле /etc/network/interfaces указывает

системе, какие интерфейсы нужно автоматически активировать (поднимать) во время загрузки. Для удаления конкретного IP-адреса с интерфейса мы используем команду `ip a del` с указанием полного адреса с маской и имени интерфейса.

Что произойдет при перезагрузке, если IP-адрес был задан только через ip?

- а) Система создаст интерфейс заново
- б) IP-адрес будет автоматически восстановлен
- в) Настройка сохранится в `/etc/network/interfaces`
- г) IP-адрес исчезнет

Верный ответ: IP-адрес исчезнет

Где задается шлюз по умолчанию в конфигурации интерфейса?

- а) В `/etc/hosts`
- б) В `resolv.conf`
- в) В поле `gateway` в `/etc/network/interfaces`
- г) В настройках DNS-сервера

Верный ответ: В поле `gateway` в `/etc/network/interfaces`

Рисунок 5.2: тест

Настройки, заданные с помощью команды `ip`, являются временными и действуют только до перезагрузки системы. При статической настройке сети в файле `/etc/network/interfaces` мы указываем шлюз по умолчанию с помощью директивы `gateway` в блоке настроек интерфейса.

Какая команда показывает открытые TCP-порты и процессы, которые их слушают?

a) netstat -an

б) ping

в) ss -tulnp

г) nc -l

Верный ответ: ss -tulnp

Какой флаг в ss включает отображение номеров портов и PID/имен процессов?

а) -n

б) -l

в) -p

г) -t

Верный ответ: -p

Рисунок 5.3: тест

Мы используем современную утилиту ss с ключами -t , -u , -l, -n, -p для получения полного списка открытых портов и связанных с ними процессов. Ключ -р в команде ss заставляет ее отображать идентификатор процесса (PID) и его имя, которое использует данный сокет.

Что делает команда dig?

- а) Определяет местоположение хоста
- б) Диагностирует сетевые маршруты
- в) Отправляет эхо-запросы
- г) Выполняет DNS-запрос и отображает IP-адреса

Верный ответ: Выполняет DNS-запрос и отображает IP-адреса

Какая команда может использоваться для проверки подключения к удаленному TCP-порту без передачи данных?

- а) nc -vz
- б) curl
- в) scp
- г) wget

Верный ответ: nc -vz

Рисунок 5.4: тест

Мы используем команду dig для выполнения DNS-запросов и получения подробной информации о доменных именах, включая их IP-адреса. Для быстрой проверки доступности удаленного TCP-порта мы используем утилиту netcat с ключами -v и -z.

Какой порт использует SSH по умолчанию?

- a) 20
- б) 21
- в) 22
- г) 443

Верный ответ: 22

Где находится основной конфигурационный файл демона SSH-сервера?

- a) /etc/ssh/ssh_config
- б) /etc/ssh/sshd_config
- в) ~/.ssh/config
- г) /etc/hosts

Верный ответ: /etc/ssh/sshd_config

Какой командой можно временно остановить службу SSH (systemd)?

- a) systemctl disable ssh
- б) systemctl stop ssh
- в) service ssh restart
- г) killall sshd

Верный ответ: systemctl stop ssh

Рисунок 5.5: тест

По умолчанию демон SSH-сервера прослушивает входящие подключения на TCP-порту 22. Главный файл конфигурации для серверной части SSH находится по пути /etc/ssh/sshd_config. Файл ssh_config предназначен для клиентской части. Для временной остановки системного сервиса мы используем команду systemctl stop.

Какой файл публичного ключа нужно добавить на сервер для авторизации по ключу?

- a) id_rsa
- б) authorized_keys
- в) id_rsa.pub
- г) known_hosts

Верный ответ: id_rsa.pub

Какой параметр в sshd_config отключает вход пользователя root по SSH?

- а) PermitRootLogin no
- б) AllowUsers root
- в) PasswordAuthentication no
- г) Port 2222

Верный ответ: PermitRootLogin no

Рисунок 5.6: тест

Для настройки аутентификации по SSH-ключу мы копируем содержимое файла публичного ключа в файл authorized_keys на сервере. Директива PermitRootLogin no в файле /etc/ssh/sshd_config запрещает прямое подключение к серверу по SSH под учетной записью root, что является важной мерой безопасности.

Какой командой включить (activate) файрвол UFW?

- a) ufw start
- б) ufw enable
- в) systemctl start ufw
- г) service ufw on

Верный ответ: ufw enable

В каком файле fail2ban хранит свои jail-конфигурации по умолчанию?

- a) /etc/fail2ban/fail2ban.conf
- б) /etc/fail2ban/jail.conf
- в) /etc/fail2ban/jail.local
- г) /etc/fail2ban/filters/jail.conf

Верный ответ: /etc/fail2ban/jail.conf

Какой флаг UFW позволяет указать конкретный номер правила для удаления?

- а) --remove
- б) --delete
- в) --num
- г) --dry-run

Верный ответ: в

Рисунок 5.7: тест

После настройки правил мы активируем межсетевой экран UFW командой ufw enable. Это применяет правила и включает автозагрузку файрвола при старте системы. Основные настройки jails Fail2ban, которые определяют

правила блокировки, хранятся в файле /etc/fail2ban/jail.conf. Чтобы удалить правило по его номеру из списка, выведенного командой ufw status numbered, мы используем команду ufw delete [номер].

Какой параметр в jail-файле fail2ban задает время блокировки IP в секундах?

- a) maxretry
- б) bantime
- в) findtime
- г) backend

Верный ответ: bantime

Какая команда добавит в UFW разрешение на SSH (порт 22) только с подсети 192.168.1.0/24?

- а) ufw allow 22
- б) ufw allow from 192.168.1.0/24 to any port 22
- в) ufw allow in 192.168.1.0/24 22
- г) ufw allow 22/tcp 192.168.1.0/24

Верный ответ: ufw allow from 192.168.1.0/24 to any port 22

Рисунок 5.8: тест

Параметр bantime в конфигурации jail Fail2ban определяет длительность блокировки IP-адреса в секундах после превышения лимита попыток. Для ограничения доступа к порту по источнику мы используем синтаксис ufw allow from [источник] to any port [порт]. Это разрешает подключения только с указанной подсети.

Если нужно удалить пакет и его зависимости, оставив только конфигурационные файлы, какую команду будете использовать?

- a) depends
- б) autoremove
- в) purge
- г) remove

Верный ответ: autoremove

Какую команду нужно использовать, чтобы удалить пакет и конфигурационные файлы, но оставить зависимости?

- а) purge
- б) autopurge
- в) remove
- г) delete

Верный ответ: purge

Какую команду нужно выполнять регулярно?

- а) apt show
- б) man sources.list
- в) apt update
- г) apt install

Верный ответ: apt update

Рисунок 5.9: тест

Команда apt autoremove удаляет пакеты, которые были установлены автоматически как зависимости и больше не нужны. Сами целевые пакеты удаляются командой remove, а их конфиги остаются. Команда apt purge удаляет пакет вместе с его конфигурационными файлами. Зависимости, установленные с ним, при этом остаются в системе. Мы регулярно выполняем apt update, чтобы обновить локальную базу данных пакетов. Без этого система не будет знать о новых версиях пакетов.

С помощью какого пакета можно настроить автоматическое обновление системы?

- a) htop
- б) gzip
- в) update
- г) unattended-upgrades

Верный ответ: unattended-upgrades

Какую команду нужно использовать для установки новых версий пакетов?

- а) update
- б) autoupdate
- в) upgrade
- г) clean

Верный ответ: upgrade

С помощью какой команды можно выполнить полное обновление системы с моментальным удалением ненужных по мнению apt пакетов?

- а) full-upgrade
- б) unattended-upgrades
- в) clean-upgrade
- г) full-update

Верный ответ: full-upgrade

Рисунок 5.10: тест

Пакет unattended-upgrades позволяет нам настроить автоматическую установку обновлений безопасности и других пакетов без ручного вмешательства. После apt update мы выполняем apt upgrade, чтобы установить все доступные обновления для установленных пакетов. Команда apt full-upgrade выполняет более интеллектуальное обновление, которое может удалять obsolete пакеты или устанавливать новые зависимости, что иногда необходимо для полного обновления системы.

Какую команду нужно использовать, чтобы системы попробовала установить недостающие зависимости?

- a) --fix-broken install
- б) list --installed
- в) --fix-broken delete
- г) grep

Верный ответ: --fix-broken install

Что случится, если вы удалите какой-то пакет, от которого зависят другие пакеты?

- а) Сервер выключится
- б) Сработает apt-cache search
- в) Разрыв зависимостей

Верный ответ: Разрыв зависимостей

Что нужно применить в выводе, чтобы найти информацию по конкретному пакету?

- а) grep
- б) list
- в) apt-cache

Верный ответ: grep

Рисунок 5.11: тест

При возникновении проблем с зависимостями мы используем команду apt –fix-broken install, чтобы попытаться автоматически исправить нарушенные зависимости. Принудительное удаление пакета, от которого зависят другие программы, приводит к «разрыву зависимостей». Для фильтрации вывода других команд и поиска конкретного пакета мы используем конвейер с grep.

Какой вариант установки .deb-файлов рекомендован?

- a) apt
- б) dpkg
- в) sudo

Верный ответ: apt

Через что apt проверяет подлинность пакетов?

- a) GPG-подпись
- б) Имя пользователя
- в) SSH

Верный ответ: GPG-подпись

Что лучше сделать, если в репозитории SelectOS нет нужного пакета?

- а) Установить самостоятельно
- б) Создать тикет через панель управления
- в) Написать в комьюнити в Telegram

Верный ответ: Создать тикет через панель управления

Рисунок 5.12: тест

Мы предпочтаем устанавливать .deb файлы через apt, так как он автоматически разрешает и устанавливает все зависимости. APT проверяет целостность и подлинность пакетов из репозиториев с помощью GPG-ключей, что защищает систему от установки модифицированных или вредоносных

пакетов. В корпоративной среде, если нужного пакета нет в утвержденных репозиториях, правильным действием является запрос на его добавление через систему тикетов, а не самостоятельная установка из непроверенных источников.

Какие три ключевые задачи системного администратора решаются с помощью анализа логов, согласно материалу урока?

- а) Установка обновлений, настройка сети, резервное копирование
- б) Диагностика сбоев, расследование инцидентов безопасности, аудит производительности
- в) Управление пользователями, настройка файрвола, мониторинг дискового пространства
- г) Компиляция ядра, написание скриптов, управление пакетами

Верный ответ: Диагностика сбоев, расследование инцидентов безопасности, аудит производительности

Веб-сайт перестал открываться, показывая ошибку «502 Bad Gateway». Основываясь на примере из урока, какой первый шаг будет наиболее эффективным для диагностики проблемы?

- а) Немедленно перезагрузить весь сервер
- б) Проверить сетевое подключение к серверу с помощью ping
- в) Изучить файл журнала ошибок веб-сервера (например, /var/log/nginx/error.log)
- г) Переустановить веб-сервер и PHP-FPM

Верный ответ: Изучить файл журнала ошибок веб-сервера (например, /var/log/nginx/error.log)

Где в современной Linux-системе хранятся системные журналы?

- а) Только в текстовых файлах в директории /etc/logs/
- б) Только в бинарном формате, доступном через journalctl
- в) В текстовых файлах в /var/log/ и в бинарном журнале systemd, доступном через journalctl
- г) В специальной базе данных в домашней директории пользователя root

Верный ответ: В текстовых файлах в /var/log/ и в бинарном журнале systemd, доступном через journalctl

Рисунок 5.13: тест

Мы используем логи в первую очередь для этих трех целей: найти причину неисправности, выяснить обстоятельства взлома и определить «узкие места» в системе. Ошибка 502 обычно указывает на проблему связи веб-сервера с

backend-процессом. В современных системах логи хранятся в двух основных местах: классические текстовые файлы в `/var/log/` и централизованный бинарный журнал `systemd`, который мы просматриваем с помощью `journalctl`.

Каково типичное взаимодействие между `systemd-journald` и `rsyslog` в современных дистрибутивах Linux?

- a) `rsyslog` собирает все логи и передает их в `systemd-journald`
- б) `systemd-journald` и `rsyslog` работают полностью независимо и не взаимодействуют
- в) `systemd-journald` собирает все системные сообщения и может перенаправлять их в `rsyslog` для записи в текстовые файлы
- г) `rsyslog` является устаревшей технологией и полностью заменен на `systemd-journald`

Верный ответ: `systemd-journald` собирает все системные сообщения и может перенаправлять их в `rsyslog` для записи в текстовые файлы

Служба не смогла запуститься при старте системы. Согласно стандартной иерархии уровней важности (priority), какой уровень, скорее всего, будет присвоен этому событию?

- a) `info` (информационное сообщение)
- б) `debug` (отладочное сообщение)
- в) `emergency` (система неработоспособна)
- г) `error` (ошибка)

Верный ответ: `error` (ошибка)

Каково назначение «категории» (facility) в сообщениях `syslog`?

- а) Указывать на уровень критичности события (например, ошибка или предупреждение)
- б) Указывать на источник сообщения (например, ядро, служба аутентификации, планировщик cron) для маршрутизации
- в) Содержать основной текст лога с описанием произошедшего
- г) Определять точное время, когда произошло событие

Верный ответ: Указывать на источник сообщения (например, ядро, служба аутентификации, планировщик cron) для маршрутизации

Рисунок 5.14: тест

Обычно `systemd-journald` действует как первичный сборщик логов, а затем, при наличии настроек, перенаправляет сообщения в демон `syslog` для постоянного хранения в привычных текстовых файлах в `/var/log/`. Сбой запуска

критичной системной службы — это значимое негативное событие, которое классифицируется уровнем error, а не просто информационным сообщением. Категория в syslog указывает на подсистему или программу-источник сообщения.

У вас есть файл с IP-адресами, многие из которых повторяются. Какой конвейер команд правильно подсчитает количество вхождений каждого уникального IP-адреса?

- a) uniq -c | sort -nr ip_list.txt
- б) sort ip_list.txt | uniq -c
- в) uniq ip_list.txt | sort
- г) awk '{print \$1}' ip_list.txt | uniq

Верный ответ: sort ip_list.txt | uniq -c

Что означает конструкция \$3 в команде awk '{print \$3}'?

- а) Вывести третью строку из входных данных
- б) Вывести третий символ каждой строки
- в) Вывести третье поле (столбец) каждой строки, разделенное пробелами
- г) Вывести переменную с именем 3

Верный ответ: Вывести третье поле (столбец) каждой строки, разделенное пробелами

Вам нужно найти в /var/log/syslog все записи, содержащие слово "error", но при этом исключить из вывода строки, где упоминается healthcheck. Какая команда подходит для этой задачи?

- а) grep "error" /var/log/syslog
- б) grep "error" /var/log/syslog | grep -v "healthcheck"
- в) egrep "error|healthcheck" /var/log/syslog
- г) grep -v "error" /var/log/syslog | grep "healthcheck"

Верный ответ: grep "error" /var/log/syslog | grep -v "healthcheck"

Рисунок 5.15: тест

Чтобы uniq мог корректно подсчитать повторяющиеся строки, они должны следовать друг за другом. В awk \$1, \$2, \$3 и т.д. обозначают первое, второе, третье и последующие поля в строке. Мы используем конвейер: первый grep

отфильтровывает строки с «error», а второй grep с ключом -v удаляет из этого результата строки, содержащие «healthcheck».

Вы наблюдаете в логах сотни попыток входа за короткое время для пользователей admin, root, test, user с одного и того же IP-адреса. Какой тип активности это, скорее всего, означает?

- а) Системный сбой, вызвавший повторные попытки подключения
- б) Пользователь, который забыл свой логин и пароль
- в) Автоматизированная атака по подбору пароля (brute-force) и перебору имен пользователей
- г) Нормальная активность службы мониторинга

Верный ответ: Автоматизированная атака по подбору пароля (brute-force) и перебору имен пользователей

После сбора релевантных логов в отдельный файл для расследования инцидента, какой следующий шаг является критически важным для обеспечения целостности доказательств?

- а) Открыть файл в текстовом редакторе и добавить свои комментарии
- б) Отправить файл по электронной почте руководителю
- в) Рассчитать и сохранить криптографическую контрольную сумму (хеш) файла, например, SHA256
- г) Сжать файл в архив для экономии места

Верный ответ: Рассчитать и сохранить криптографическую контрольную сумму (хеш) файла, например, SHA256

В ходе расследования вам необходимо выяснить, какие команды мог выполнить злоумышленник с повышенными привилегиями. Какой шаблон в логах является наиболее релевантным для этого поиска?

- а) Поиск записей Accepted password в /var/log/auth.log
- б) Поиск записей, содержащих sudo:, в /var/log/auth.log или syslog
- в) Поиск записей COMMAND= в логах веб-сервера
- г) Поиск записей CRON в /var/log/cron.log

Верный ответ: Поиск записей, содержащих sudo:, в /var/log/auth.log или syslog

Рисунок 5.16: тест

Большое количество неудачных попыток входа за короткий промежуток времени, особенно для стандартных имен пользователей, является классическим признаком автоматизированной brute-force атаки. Расчет хеша фиксирует текущее состояние файла. Команды, выполненные через sudo, подробно логируются в журналах аутентификации.

Какую основную и наиболее насущную проблему решает утилита logrotate?

- а) Анализ логов на предмет угроз безопасности
- б) Централизованный сбор логов с нескольких серверов
- в) Предотвращение исчерпания свободного места на диске из-за неконтролируемого роста лог-файлов
- г) Уведомление администратора об ошибках в реальном времени

Верный ответ: Предотвращение исчерпания свободного места на диске из-за неконтролируемого роста лог-файлов

Какое ключевое преимущество в области безопасности дает отправка логов на выделенный удаленный сервер?

- а) Логи начинают занимать меньше места на диске
- б) Ускоряется работа приложений на исходном сервере
- в) Злоумышленник, получивший доступ к серверу, не сможет легко скрыть свои следы, удалив локальные логи
- г) Упрощается синтаксис команд для поиска по логам

Верный ответ: Злоумышленник, получивший доступ к серверу, не сможет легко скрыть свои следы, удалив локальные логи

Компания разворачивает приложение в Kubernetes и ищет наиболее экономичное решение для логирования. Основная потребность – фильтрация по меткам (имя пода, неймспейс), а скорость полнотекстового поиска не является приоритетом. Какая система из рассмотренных в уроке лучше всего подходит под эти требования?

- а) ELK Stack / OpenSearch, так как он индексирует все содержимое логов
- б) Grafana Loki, так как он индексирует только метки и хранит тексты логов в сжатом виде
- в) Graylog, так как он предоставляет готовые дашборды для SIEM
- г) rsyslog, настроенный на локальную запись в файлы

Верный ответ: Grafana Loki, так как он индексирует только метки и хранит тексты логов в сжатом виде

Рисунок 5.17: тест

Основная задача logrotate — ротация, архивация и удаление старых лог-файлов по заданному расписанию и правилам. Удаленный сбор логов обеспечивает целостность журналов. Grafana Loki спроектирован именно для этого сценария.

Какая команда позволяет проверить конфигурацию Podman?

- а) podman run --check
- б) podman system
- в) podman info
- г) podman status

Верный ответ: podman info

Чем Podman отличается от Docker?

- а) Не поддерживает rootless-режим
- б) Не требует фонового демона
- в) Использует только свою ОС
- г) Не поддерживает CLI

Верный ответ: Не требует фонового демона

Что делает ключ --rm в podman run

- а) Перезапускает контейнер при сбое
- б) Подключает к сети хоста
- в) Удаляет контейнер после завершения
- г) Обновляет образ

Верный ответ: Удаляет контейнер после завершения

Рисунок 5.18: тест

Команда podman info выводит подробную информацию о среде Podman: версию, конфигурацию, хранилища, сеть и т.д., что помогает нам проверить корректность его настройки. Ключевое архитектурное отличие Podman в том, что он использует архитектуру без демона. Ключ –rm автоматически удаляет контейнер сразу после того, как он завершит свою работу.

Почему Podman безопаснее для многопользовательских систем?

- а) Работает только с root-доступом
- б) Контейнеры не используют ядро хоста
- в) Поддерживает запуск без root и демона
- г) Требует отдельную ОС на каждый контейнер

Верный ответ: Поддерживает запуск без root и демона

Какой компонент не нужен при работе с Podman?

- а) Системный демон
- б) Образ
- в) Контейнер
- г) Командная строка

Верный ответ: Системный демон

Рисунок 5.19: тест

Безопасность Podman для многопользовательских сред обусловлена его способностью работать в rootless-режиме. Как уже было отмечено, Podman не требует постоянно работающего фонового демона.

Какая команда запускает контейнер с пробросом порта 8080 на 80?

- а) podman expose 8080:80 nginx
- б) podman publish -p 8080:80 nginx
- в) podman run -p 8080:80 nginx
- г) podman exec -p 8080:80 nginx

Верный ответ: podman run -p 8080:80 nginx

Что делает флаг -v при запуске контейнера?

- а) Пробрасывает сетевой интерфейс
- б) Монтирует директорию с хоста внутрь контейнера
- в) Задает версию образа
- г) Устанавливает переменную окружения

Верный ответ: Монтирует директорию с хоста внутрь контейнера

Что произойдет при использовании флага -it?

- а) Контейнер будет запущен в фоне
- б) Откроется shell внутри контейнера
- в) Образ будет автоматически загружен
- г) Контейнер завершится сразу после запуска

Верный ответ: Откроется shell внутри контейнера

Рисунок 5.20: тест

Мы используем команду `podman run` для запуска нового контейнера. Флаг `-v` используется для монтирования директорий или файлов с хостовой машины внутрь контейнера. Комбинация флагов `-i` и `-t` позволяет нам запустить контейнер в интерактивном режиме с псевдо-TTY.

Как просмотреть все (включая остановленные) контейнеры?

- а) podman ps
- б) podman ls
- в) podman list
- г) podman ps -a

Верный ответ: podman ps -a

Что делает флаг --rm при запуске контейнера?

- а) Удаляет образ после использования
- б) Автоматически удаляет контейнер после завершения
- в) Перезапускает контейнер при сбое
- г) Запускает контейнер в фоновом режиме

Верный ответ: Автоматически удаляет контейнер после завершения

Рисунок 5.21: тест

По умолчанию podman ps показывает только работающие контейнеры. Чтобы увидеть все контейнеры, включая остановленные, мы добавляем флаг -a. Флаг -rm заставляет Podman автоматически удалять контейнер сразу после его остановки.

Какая команда позволяет отслеживать использование ресурсов контейнера в реальном времени?

- а) podman stats
- б) podman ps
- в) podman run
- г) podman list

Верный ответ: podman stats

Что делает флаг -d при запуске контейнера?

- а) Монтирует директорию с хоста внутрь контейнера
- б) Удаляет контейнер после завершения
- в) Запускает контейнер в фоновом режиме
- г) Устанавливает переменную окружения

Верный ответ: Запускает контейнер в фоновом режиме

Что делает команда systemctl daemon-reload?

- а) Запускает сервис
- б) Останавливает сервис
- в) Запускает автозагрузку сервиса
- г) Перечитывает unit-файлы

Верный ответ: Перечитывает unit-файлы

Рисунок 5.22: тест

Для мониторинга потребления ресурсов работающими контейнерами в реальном времени мы используем команду podman stats. Флаг -d запускает контейнер в фоновом режиме. После запуска управление возвращается в терминал, а контейнер продолжает работать независимо. После изменения unit-файлов systemd мы выполняем systemctl daemon-reload.

Как просмотреть логи контейнера?

- а) journalctl ps имя.service
- б) journalctl -u имя.service
- в) journalctl list имя.service
- г) systemctl enable имя.service

Верный ответ: journalctl -u имя.service

Что делает флаг --memory=512m при запуске контейнера?

- а) Устанавливает жесткое ограничение оперативной памяти (RAM) для контейнера
- б) Гарантирует, что контейнер получит 512 МБ RAM, даже если на сервере недостаточно памяти
- в) Автоматически увеличивает лимит памяти до 512 МБ, если контейнеру не хватает ресурсов
- г) Резервирует 512 МБ RAM исключительно для контейнера, запрещая другим процессам хоста использовать эту память

Верный ответ: Устанавливает жесткое ограничение оперативной памяти (RAM) для контейнера

Рисунок 5.23: тест

Когда контейнер запущен как systemd-сервис, его логи интегрируются в общий журнал systemd. Флаг –memory устанавливает максимальный лимит оперативной памяти, который может использовать контейнер.

Какая команда используется для загрузки образа из реестра?

- a) podman load
- б) podman fetch
- в) podman pull
- г) podman clone

Верный ответ: podman pull

Где хранится политика доверия для Podman?

- а) /etc/podman/trust.json
- б) /etc/containers/policy.json
- в) /usr/share/podman/trust
- г) /etc/pki/policy.json

Верный ответ: /etc/containers/policy.json

Что делает команда podman build -t myapp .?

- а) Собирает образ из файла YAML
- б) Загружает образ с тегом myapp
- в) Собирает образ из Dockerfile в текущей директории
- г) Удаляет образ с тегом myapp

Верный ответ: Собирает образ из Dockerfile в текущей директории

Рисунок 5.24: тест

Для загрузки Docker-образов из удаленного реестра мы используем команду podman pull. Файл policy.json определяет политику доверия для образов контейнеров. В нем мы настраиваем, из каких реестров разрешено скачивать образы, требуется ли для них цифровая подпись и какие ключи являются

доверенными. Команда podman build используется для сборки собственного образа контейнера.

Что делает команда podman build -t myapp .?

- а) Собирает образ из файла YAML
- б) Загружает образ с тегом myapp
- в) Собирает образ из Dockerfile в текущей директории
- г) Удаляет образ с тегом myapp

Верный ответ: Собирает образ из Dockerfile в текущей директории

Что происходит при использовании неподписанного образа, если настроен signedBy?

- а) Образ будет принят с предупреждением
- б) Подпись будет добавлена автоматически
- в) Загрузка завершится ошибкой
- г) Подпись будет проигнорирована

Верный ответ: Загрузка завершится ошибкой

Рисунок 5.25: тест

Мы используем podman build -t myapp . для создания собственного образа из инструкций в Dockerfile, который находится в текущей директории. Результатом будет образ с именем myapp. Если в политике доверия (policy.json) для определенного реестра или образа указана директива signedBy, то Podman будет проверять наличие и валидность цифровой подписи у образа.

6 Оценки тестов

Тест по теме «Основы сетевой конфигурации в Linux»

Результат тестирования

Тест пройден

5 из 5

Рисунок 6.1: тест 1

Тест по теме «Базовая диагностика сети»

Результат тестирования

Тест пройден

4 из 4

Рисунок 6.2: тест 2

Тест по теме «Настройка SSH-доступа и его защита»

Результат тестирования

Тест пройден

5 из 5

Рисунок 6.3: тест 3

Тест по теме «Повышение безопасности сетевого взаимодействия»

Результат тестирования

Тест пройден

4 из 5

Рисунок 6.4: тест 4

Тест по теме «Что такое пакеты и как они устроены»

Результат тестирования

Тест пройден

2 из 3

Рисунок 6.5: тест 5

Тест по теме «Обновление системы и безопасность»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.6: тест 6

Тест по теме « Работа с зависимостями и решение конфликтов»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.7: тест 7

Тест по теме «Работа с локальными .deb-пакетами и сторонними источниками»

Результат тестирования

Тест пройден

1 из 3

Рисунок 6.8: тест 8

Тест по теме «Знакомство с логами»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.9: тест 9

Тест по теме «Система логирования в Linux»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.10: тест 10

Тест по теме «Поиск и фильтрация логов под конкретные задачи»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.11: тест 11

Тест по теме «Расследование инцидентов по логам»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.12: тест 12

Тест по теме «Управление жизненным циклом логов и ротация»

Результат тестирования

Тест пройден

3 из 3

Рисунок 6.13: тест 13

Тест по теме «Контейнеризация как подход»

Результат тестирования

Тест пройден

5 из 5

Рисунок 6.14: тест 14

Тест по теме «Работа с контейнерами в Podman»

Результат тестирования

Тест пройден

5 из 5

Рисунок 6.15: тест 15

Тест по теме «Управление ресурсами контейнеров»

Результат тестирования

Тест пройден

5 из 5

Рисунок 6.16: тест 16

Тест по теме «Образы, реестры и базовая безопасность»

Результат тестирования

Тест пройден

4 из 4

Рисунок 6.17: тест 17

7 Выводы

Мы выполнили третий раздел внешнего курса «Системный администратор Linux с нуля».

Список литературы

1. <https://study.selectel.ru/members/courses/course756726784647>