

Внешний курс. Раздел 3

Артём Дмитриевич Петлин

2025-11-22

Содержание i

1. Информация
2. Цель работы
3. Задание
4. Теоретическое введение
5. Выполнение практических заданий
6. Выполнение тестовых заданий
7. Оценки тестов
8. Выводы

1. Информация

1.1 Докладчик

- Петлин Артём Дмитриевич
- студент
- группа НПИбд-02-24
- Российский университет дружбы народов
- 1132246846@pfur.ru
- https://github.com/hikrim/study_2025-2026_os2



2. Цель работы

2.1 Цель работы

Выполнить третий раздел внешнего курса “Системный администратор Linux с нуля”.

3. Задание

3.1 Задание

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

4. Теоретическое введение

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

5. Выполнение практических заданий

5.1 Ход работы

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

Рисунок 1: модуль 8

Настраиваем статический IP на тестовом сервере вручную.

5.2 Ход работы

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

Рисунок 2: модуль 8

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

5.3 Ход работы

Проверяем доступность внешних ресурсов
через ping.

```
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 3: модуль 8

5.4 Ход работы

```
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: модуль 8

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

5.5 Ход работы

С помощью утилиты dig узнаем IP-адреса популярных сервисов.

```
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:~$
```

5.6 Ход работы

```
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 6: модуль 8

Проверяем, слушает ли SSH-порт на сервере.

5.7 Ход работы

Изменяем порт SSH и запрещаем вход по паролю.

```
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
#PubkeyAuthentication yes
```

Рисунок 7: модуль 8

5.8 Ход работы

Настраиваем подключение по ключу и авторизуемся.

```
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:xwSgX6bN8qspnjQJ+b508zRtruKfke7b6x2p+4aJ94 adpetlin@adpetlin
The key's randomart image is:
+-- [ED25519 256] --+
|   ...
|   . o
|   . . o
|   . . +
|   .. S o.
|   . + .+. +
|   + +,+.+=+,
|   .+ o+=+*=+
|   ...==+=+B%Eo
+--- [SHA256] ---+
```

Рисунок 8: модуль 8

5.9 Ход работы

```
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 9: модуль 8

Разрешаем SSH-доступ только для определенных пользователей. Убеждаемся, что для другого пользователя подключение не сработает.

5.10 Ход работы

```
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 10: модуль 8

Настраиваем UFW для ограничения доступа только для SSH.

5.11 Ход работы

```
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 11: модуль 8

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

5.12 Ход работы

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

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

Рисунок 12: модуль 8

```
/etc/fail2ban/filter.d/nginx.conf [----] 11 L:[ 1+ 2
nginx-req-limit
enabled = true
port = http,https
filter = nginx-req-limit
logpath = /var/log/nginx/access.log
maxretry = 3
jailtime = 60
bantime = 600
```

Рисунок 13: модуль 8

5.13 Ход работы

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

Рисунок 14: модуль 8

Проверяем, как работает блокировка при множественных запросах.

5.14 Ход работы

```
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 15: модуль 9

Находим и устанавливаем утилиту htop.

5.15 Ход работы

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

```
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+nmui1) ...
Обрабатываются триггеры для man-db (2.11.2-2) ...
adpetlin@adpetlin:~$
```

Рисунок 16: модуль 9

5.16 Ход работы

Выполняем полное обновление системы.

```
adpetlin@adpetlin:~$ sudo apt autoremove --purge
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состояниях... 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 томгеры 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 17: модуль 9

5.17 Ход работы

Настраиваем автоматическую установку обновлений безопасности.

```
настраивается пакет libbz2api-krb5-2:amd64 (1.20.1-2+deb12u2) ...
настраивается пакет groff-base (1.22.4-19) ...
настраивается пакет liblircpc3: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 libns12:amd64 (1.3.9-2) ...
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 libcryptsetup2:amd64 (2:2.6.1-4~deb12u2) ...
обрабатывается триггер для debianutils (5.7-0.5~deb12u1) ...
обрабатывается триггер для install-info (6.8-6) ...
обрабатывается триггер для mailcap (3.70+nmu1) ...
обрабатывается триггер для initramfs-tools (0.142+deb12u1) ...
update-initramfs: Generating /boot/initrd.lz-6.1.0-27-amd64
обрабатывается триггер для libc-bin (2.36-9+deb12u8) ...
обрабатывается триггер для systemd (252.31-1~deb12u1) ...
обрабатывается триггер для man-db (2.11.2-2) ...
обрабатывается триггер для ca-certificates (20250419) ...
updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
running hooks in /etc/ca-certificates/update.d...
done.
adretlin@adretlin:~$ sudo apt upgrade
 чтение списков пакетов... Готово
 Построение дерева зависимостей... Готово
 чтение информации о состояниях... Готово
 Расчет обновлений... Готово
 Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.
adretlin@adretlin:~$
```

Рисунок 18: модуль 9

5.18 Ход работы

```
root@adpetlin:~# exit
Выход
adpetlin@adpetlin:~$ sudo apt install unattended-upgrades
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
E: Невозможно найти пакет unattended-upgrades
adpetlin@adpetlin:~$ 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
adpetlin@adpetlin:~$ sudo apt autoremove -y
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.
adpetlin@adpetlin:~$ _
```

Рисунок 19: модуль 9

Удаляем устаревшие и неиспользуемые зависимости.

5.19 Ход работы

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

```
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, 208.00.154.139, 64.50.233.108, ...  
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 зависимостей – оставляем не настроенным  
Обрабатываются триггеры для mailcap (3.79+multi) ...  
Обрабатываются триггеры для man-db (2.11.2-2) ...  
При обработке следующих пакетов произошли ошибки:  
    htop  
root@adpetlin:~#
```

Рисунок 20: модуль 9

5.20 Ход работы

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

Рисунок 21: модуль 9

Анализируем, какие зависимости потребуются.

5.21 Ход работы

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

```
[root@rhel7 ~]# ps aux | grep myhaem
root      1  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 bash -c /usr/bin/python /opt/myhaem/main.py
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Main process exited, code=exited, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed with result exit-code
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 777,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_service - My haem received copies.
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed with result exit-code
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 776,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_service - My haem received copies.
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 775,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 774,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 773,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 772,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 771,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 770,
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Failed to determine user credentials! No such process
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Started myhaem_main, status=217/USER
75 99999 100  0.0  0.0  1000  100 ? Ss Jan01 00:00:00 myhaem_main Job restart counter is at 769,
```

Рисунок 22: модуль 10

Рисунок 23: модуль 10

5.22 Ход работы

```
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/bun/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/bun/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/bun/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 24: модуль 10

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

5.23 Ход работы

```
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 25: модуль 10

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

5.24 Ход работы

```
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-sa1 > /dev/null && debian-sa1 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-sa1 > /dev/null && debian-sa1 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-sa1 > /dev/null && debian-sa1 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 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.

5.25 Ход работы

```
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/10afde83d1d74c48857b6c7f78f25
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 27: модуль 10

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

5.26 Ход работы

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

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

Рисунок 28: модуль 10

```
root@adpetlin:~# grep "Это тестовое сообщение" /var/log/syslog
Jan 15 00:15:04 adpetlin root[32741]: Это тестовое сообщение
root@adpetlin:~# grep auth.log /var/log/syslog.d/50-default.conf
auth  authpriv {/var/log/auth.log} authpriv;
root@adpetlin:~# logger -p user.warn -t test "Специальное тестовое сообщение"
root@adpetlin:~# journalctl -r -warning | grep "Специальное тестовое сообщение"
Jan 15 00:17:14 adpetlin root[32771]: 9% специальное тестовое сообщение
root@adpetlin:~#
```

Рисунок 29: модуль 10

5.27 Ход работы

```
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 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.

5.28 Ход работы

```
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/bun/python3: No such proc
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/bun/python3: No such proc
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 31: модуль 10

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

5.29 Ход работы

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

```
root@adpetlin:~# grep "invalid user" /var/log/auth.log
HOR 14 22:40:06 adpetlin sshd[1432]: Invalid user user1 from 127.0.0.1 port 41544
HOR 14 22:41:27 adpetlin sshd[1453]: Invalid user user1 from 127.0.0.1 port 51282
HOR 14 23:02:47 adpetlin sshd[2301]: Invalid user user1 from 192.168.122.2 port 56812
HOR 14 23:02:47 adpetlin sshd[2300]: Invalid user user1 from 192.168.122.2 port 56818
HOR 14 23:02:48 adpetlin sshd[2321]: Invalid user user1 from 192.168.122.2 port 56826
HOR 14 23:03:11 adpetlin sshd[2300]: Invalid user user1 from 192.168.122.2 port 45466
HOR 14 23:03:11 adpetlin sshd[2357]: Invalid user user1 from 192.168.122.2 port 40186
HOR 14 23:03:31 adpetlin sshd[2365]: Invalid user user1 from 192.168.122.2 port 40198
HOR 14 23:03:31 adpetlin sshd[2378]: Invalid user user1 from 192.168.122.2 port 40204
HOR 14 23:03:34 adpetlin sshd[2302]: Invalid user user1 from 192.168.122.2 port 50602
HOR 14 23:05:19 adpetlin sshd[2456]: Invalid user user1 from 192.168.122.2 port 47978
HOR 14 23:05:20 adpetlin sshd[2453]: Invalid user user1 from 192.168.122.2 port 47982
HOR 14 23:05:20 adpetlin sshd[2476]: Invalid user user1 from 192.168.122.2 port 47984
HOR 14 23:06:01 adpetlin sshd[2603]: Invalid user user1 from 192.168.122.2 port 42052
HOR 14 23:06:01 adpetlin sshd[2610]: Invalid user user1 from 192.168.122.2 port 42058
HOR 14 23:06:01 adpetlin sshd[2628]: Invalid user user1 from 192.168.122.2 port 42068
HOR 14 23:08:07 adpetlin sshd[2653]: Invalid user user1 from 192.168.122.2 port 49358
HOR 14 23:08:51 adpetlin sshd[2670]: Invalid user user1 from 192.168.122.2 port 50368
HOR 14 23:09:41 adpetlin sshd[2703]: Invalid user user1 from 192.168.122.2 port 59788
HOR 14 23:09:41 adpetlin sshd[2710]: Invalid user user1 from 192.168.122.2 port 59804
HOR 14 23:09:41 adpetlin sshd[2723]: Invalid user user1 from 192.168.122.2 port 59816
HOR 14 23:10:45 adpetlin sshd[2757]: Invalid user user1 from 192.168.122.2 port 60218
HOR 14 23:10:45 adpetlin sshd[2766]: Invalid user user1 from 192.168.122.2 port 49548
HOR 14 23:10:45 adpetlin sshd[2779]: Invalid user user1 from 192.168.122.2 port 49558
HOR 14 23:10:55 adpetlin sshd[2784]: Invalid user user1 from 192.168.122.2 port 53492
HOR 14 23:11:10 adpetlin sshd[2788]: Invalid user user1 from 192.168.122.2 port 44496
HOR 14 23:11:15 adpetlin sshd[2792]: Invalid user user1 from 192.168.122.2 port 36278
HOR 14 23:12:49 adpetlin sshd[2830]: Invalid user user1 from 192.168.122.2 port 38210
HOR 14 23:12:49 adpetlin sshd[2837]: Invalid user user1 from 192.168.122.2 port 38226
HOR 14 23:12:49 adpetlin sshd[2850]: Invalid user user1 from 192.168.122.2 port 38232
HOR 14 23:13:17 adpetlin sshd[2882]: Invalid user trex from 192.168.122.2 port 36440
HOR 14 23:13:17 adpetlin sshd[2889]: Invalid user trex from 192.168.122.2 port 36448
HOR 14 23:13:17 adpetlin sshd[2902]: Invalid user trex from 192.168.122.2 port 36450
HOR 14 23:13:40 adpetlin sshd[2908]: Invalid user trex from 192.168.122.2 port 32772
HOR 14 23:14:20 adpetlin sshd[2942]: Invalid user user1 from 192.168.122.2 port 35986
HOR 14 23:14:20 adpetlin sshd[2949]: Invalid user user1 from 192.168.122.2 port 35988
HOR 14 23:14:21 adpetlin sshd[2962]: Invalid user user1 from 192.168.122.2 port 36002
HOR 14 23:17:32 adpetlin sshd[3042]: Invalid user user1 from 192.168.122.2 port 35968
HOR 14 23:17:32 adpetlin sshd[3049]: Invalid user user1 from 192.168.122.2 port 35970
HOR 14 23:17:32 adpetlin sshd[3062]: Invalid user user1 from 192.168.122.2 port 35986
root@adpetlin:~#
```

Рисунок 32: модуль 10

5.30 Ход работы

```
root@adpetlin:~# grep "Failed password" /var/log/auth.log | awk '{print $11}' | sort | uniq -c | sort -nr | head -n 1
      1 user1
root@adpetlin:~# grep "192.168.122.2" /var/log/auth.log > incident_report.log
root@adpetlin:~# sha256sum incident_report.log
cfec6ac47fad8626b61d5abc0ec0c017a0ccc676bcadb905b272654e12c35cca  incident_report.log
root@adpetlin:~# grep "sudo:" /var/log/auth.log | grep "USER=$(whoami)"
root@adpetlin:~# -
```

Рисунок 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 изменился после сбора. Проверяем, какие команды выполнялись с правами суперпользователя (через sudo) вашим текущим пользователем.

5.31 Ход работы

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.

```
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 34: модуль 10

5.32 Ход работы

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.

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

Рисунок 35: модуль 10

5.33 Ход работы

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

```
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 (14 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 36: модуль 10

5.34 Ход работы

```
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 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.

5.35 Ход работы

```
adpetlin@adpetlin:~$ sudo apt install podman
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... 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 38: модуль 11

Устанавливаем Podman и проверяем установку.

5.36 Ход работы

Запускаем контейнер.

```
root@localhost: ~$ docker run -it --rm alpine sh
alpine[1]@alpine: ~$ The cgroup2 manager is set to systemd but there is no systemd user session available
alpine[1]@alpine: ~$ Alternatively, you can enable llinger with: 'logind.enable-linger 1000' (possibly as root)
alpine[1]@alpine: ~$ The cgroup2 manager is set to systemd but there is no systemd user session available
alpine[1]@alpine: ~$ For using systemd, you may need to login using an user session
alpine[1]@alpine: ~$ Alternatively, you can enable llinger with: 'logind.enable-linger 1000' (possibly as root)
alpine[1]@alpine: ~$ Falling back to '--cgroup-manager=cgroups'
resolved 'alpine' is an alias /etc/containers/registries.conf.d/shortnames.conf
alpine[1]@alpine: ~$ Using --rm to remove container after stopping...
setting image source signatures
copying blob 13e3978244a done
alpine[1]@alpine: ~$ Storing image destination
Storing signatures
1:0679..129941 0PF: [100413] STRUCT
1:0679..129941 0PF: [100413] class4 vimage2
1:0679..129941 0PF: Invalid name
1:0679..129941 0PF:
```

Рисунок 39: модуль 11

```
root@localhost: ~$ docker run -it --rm debian
alpine[1]@alpine: ~$ The cgroup2 manager is set to systemd but there is no systemd user session available
alpine[1]@alpine: ~$ Alternatively, you can enable llinger with: 'logind.enable-linger 1000' (possibly as root)
alpine[1]@alpine: ~$ Falling back to '--cgroup-manager=cgroups'
alpine[1]@alpine: ~$ The cgroup2 manager is set to systemd but there is no systemd user session available
alpine[1]@alpine: ~$ Alternatively, you can enable llinger with: 'logind.enable-linger 1000' (possibly as root)
alpine[1]@alpine: ~$ Alternatively, you can enable llinger with: 'logind.enable-linger 1000' (possibly as root)
alpine[1]@alpine: ~$ resolved 'debain' is an alias /etc/containers/registries.conf.d/shortnames.conf
alpine[1]@alpine: ~$ Using --rm to remove container after stopping...
setting image source signatures
copying blob 13e3978244a done
alpine[1]@alpine: ~$ Storing image destination
Storing signatures
1:0679..129941 0PF: [100413] STRUCT
1:0679..129941 0PF: [100413] class4 vimage2
1:0679..129941 0PF: Invalid name
1:0679..129941 0PF:
```

Рисунок 40: модуль 11

5.37 Ход работы

Запускаем Nginx и проверяем, что он работает.

```
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: `loginctl 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: `loginctl enable-linger 1000` (possibly as root)
WARN[0000] Falling back to --cgroup-manager=cgroups
Trying to pull docker.io/library/nginx:latest...
Getting Image source signatures
Copying blob 52bc359e0cd7 done
Copying blob 266628526d42 done
Copying blob d7ecded77e2a done
Copying blob 921c57c6a81 done
Copying blob 320b6949be89 done
Copying blob 9def903393e4 done
Copying blob e2f8e236ed9df done
Copying config d261f019cb done
Writing manifest to image destination
Storing signatures
6b10c8631766ab6942ea71f4e4eed8dc00e1aa1b8db432f70841a5dfa86dc8
adpetlin@adpetlin:~$ _
```

Рисунок 41: модуль 11

5.38 Ход работы

```
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: `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
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 42: модуль 11

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

5.39 Ход работы

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

ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET IO	BLOCK IO	PIDS	CPU TIME
609057e69742	nginx-limited	733.18%	30.72MB / 104.3MB	29.30%	430B / 110B	0B / 0B	28	2m3.695512s

Рисунок 43: модуль 11

5.40 Ход работы

Исследуем поведение контейнера при исчерпании выделенной памяти.

Рисунок 44: модуль 11

Рисунок 45: модуль 11

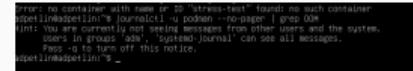


Рисунок 46: модуль 11

5.41 Ход работы

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



Рисунок 47: модуль
11



Рисунок 48: модуль
11



Рисунок 49: модуль
11



Рисунок 50: модуль
11

5.42 Ход работы

Сохраняем образ и переносим его на другой сервер.

```
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 system, you may need to login using an user session
[warn] [0000] Alternatively, you can enable lingering with: 'logind 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 system, you may need to login using an user session
[warn] [0000] Alternatively, you can enable lingering with: 'logind enable-linger 1000' (possibly as root)
[warn] [0000] Falling back to '--cgroup-manager=cgroups'
Copying blob 36d08fe0cb65 done
Copying blob e1e9587ac541 done
Copying blob 8fbeb165cd673 done
Copying blob 2ceddc4cf78a7b done
Copying blob 99cd1b1b6ad3 done
Copying blob d81df9ef4f0897 done
Copying blob d7217c5edca4 done
Copying blob 76b348108d34 done
Copying config f4d7b46fea 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 user@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 user@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 system, you may need to login using an user session
[warn] [0000] Alternatively, you can enable lingering with: 'logind 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 system, you may need to login using an user session
[warn] [0000] Alternatively, you can enable lingering with: 'logind enable-linger 1000' (possibly as root)
[warn] [0000] Falling back to '--cgroup-manager=cgroups'
Error: rootlessport listen tcp 0.0.0.0:8080: bind: address already in use
adpetlin@adpetlin:~$ curl http://localhost:8080
<html><body>from my custom container!</body></html>
adpetlin@adpetlin:~$ _
```

Рисунок 51: модуль 11

6. Выполнение тестовых заданий

6.1 Тестовые задания

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.

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 52: тест

6.2 Тестовые задания

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.

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

- a) Система создаст интерфейс заново
- b) IP-адрес будет автоматически восстановлен
- c) Настройка сохранится в `/etc/network/interfaces`
- d) IP-адрес исчезнет

Верный ответ: IP-адрес исчезнет

Где задается шлюз по умолчанию в конфигурации интерфейса?

- a) В `/etc/hosts`
- b) В `resolv.conf`
- c) В поле `gateway` в `/etc/network/interfaces`
- d) В настройках DNS-сервера

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

Рисунок 53: тест

6.3 Тестовые задания

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

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

- a) netstat -an
- б) ping
- в) ss -tuInp
- г) nc -l

Верный ответ: `ss -tuInp`

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

- а) -n
- б) -l
- в) -p
- г) -t

Верный ответ: `-p`

Рисунок 54: тест

6.4 Тестовые задания

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`.

Что делает команда `dig`?

- a) 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 55: тест

6.5 Тестовые задания

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 порт использует SSH по умолчанию?

- а) 20
- б) 21
- в) 22
- г) 443

Верный ответ: 22

Где находится основной конфигурационный файл демона SSH-сервера?

- а) `/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`)?

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

Верный ответ: `systemctl stop ssh`

Рисунок 56: тест

6.6 Тестовые задания

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

- a) id_rsa
- b) authorized_keys
- c) id_rsa.pub
- d) known_hosts

Верный ответ: id_rsa.pub

Какой параметр в sshd_config отключает вход пользователя root по SSH?

- a) PermitRootLogin no
- b) AllowUsers root
- c) PasswordAuthentication no
- d) Port 2222

Верный ответ: PermitRootLogin no

Рисунок 57: тест

6.7 Тестовые задания

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 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?

- а) /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

6.8 Тестовые задания

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.

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

- a) maxretry
- b) bantime
- c) findtime
- d) backend

Верный ответ: bantime

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

- a) ufw allow 22
- b) ufw allow from 192.168.1.0/24 to any port 22
- c) ufw allow in 192.168.1.0/24 22
- d) ufw allow 22/tcp 192.168.1.0/24

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

Рисунок 59: тест

6.9 Тестовые задания

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.

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

- a) depends
- b) autoremove
- c) purge
- d) remove

Верный ответ: autoremove

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

- a) purge
- b) autoremove
- c) remove
- d) delete

Верный ответ: purge

Какую команду нужно выполнять регулярно?

- a) apt show
- b) man sources.list
- c) apt update
- d) apt install

Верный ответ: apt update

Рисунок 60: тест

6.10 Тестовые задания

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.

С помощью какого пакета можно настроить автоматическое обновление системы?

- a) htop
- b) gzip
- c) update
- d) unattended-upgrades

Верный ответ: unattended-upgrades

Какую команду нужно использовать для установки новых версий пакетов?

- a) update
- b) autoupdate
- c) upgrade
- d) clean

Верный ответ: upgrade

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

- a) full-upgrade
- b) unattended-upgrades
- c) clean-upgrade
- d) full-update

Верный ответ: full-upgrade

Рисунок 61: тест

6.11 Тестовые задания

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`.

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

- a) `--fix-broken install`
- b) `list --installed`
- c) `--fix-broken delete`
- d) `grep`

Верный ответ: `--fix-broken install`

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

- a) Сервер выключится
- b) Сработает `apt-cache search`
- c) Разрыв зависимостей

Верный ответ: Разрыв зависимостей

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

- a) `grep`
- b) `list`
- c) `apt-cache`

Верный ответ: `grep`

Рисунок 62: тест

6.12 Тестовые задания

Мы предпочитаем 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.

Какой вариант установки .deb-файлов рекомендован?

- a) apt
- б) dpkg
- в) sudo

Верный ответ: apt

Через что apt проверяет подлинность пакетов?

- а) 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

6.13 Тестовые задания

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`.

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 64: тест

6.14 Тестовые задания

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.

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

- а) `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?

- а) `info` (информационное сообщение)
- б) `debug` (отладочное сообщение)
- в) `emergency` (системанеработоспособна)
- г) `errort` (ошибка)

Верный ответ: `errort` (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 65: тест

6.15 Тестовые задания

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”.

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 66: тест

6.16 Тестовые задания

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.

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 67: тест

6.17 Тестовые задания

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.

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 68: тест

6.18 Тестовые задания

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.

Какая команда позволяет проверить конфигурацию Podman?

- a) `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 69: тест

6.19 Тестовые задания

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

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 70: тест

6.20 Тестовые задания

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.

Какая команда запускает контейнер с пробросом порта 8080 на 80?

- a) podman expose 8080:80 nginx
- b) podman publish -p 8080:80 nginx
- c) podman run -p 8080:80 nginx
- d) podman exec -p 8080:80 nginx

Верный ответ: podman run -p 8080:80 nginx

Что делает флаг -v при запуске контейнера?

- a) Пробрасывает сетевой интерфейс
- b) Монтирует директорию с хоста внутрь контейнера
- c) Задает версию образа
- d) Устанавливает переменную окружения

Верный ответ: Монтирует директорию с хоста внутрь контейнера

Что произойдет при использовании флага -it?

- a) Контейнер будет запущен в фоне
- b) Откроется shell внутри контейнера
- c) Образ будет автоматически загружен
- d) Контейнер завершится сразу после запуска

Верный ответ: Откроется shell внутри контейнера

Рисунок 71: тест

6.21 Тестовые задания

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

Как просмотреть все (включая остановленные) контейнеры?

- a) podman ps
- b) podman ls
- c) podman list
- d) podman ps -a

Верный ответ: podman ps -a

Что делает флаг --rm при запуске контейнера?

- a) Удаляет образ после использования
- b) Автоматически удаляет контейнер после завершения
- c) Перезапускает контейнер при сбое
- d) Запускает контейнер в фоновом режиме

Верный ответ: Автоматически удаляет контейнер после завершения

Рисунок 72: тест

6.22 Тестовые задания

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.

Какая команда позволяет отслеживать использование ресурсов контейнера в реальном времени?

- a) podman stats
- b) podman ps
- c) podman run
- d) podman list

Верный ответ: podman stats

Что делает флаг -d при запуске контейнера?

- a) Монтирует директорию с хоста внутрь контейнера
- b) Удаляет контейнер после завершения
- c) Запускает контейнер в фоновом режиме
- d) Устанавливает переменную окружения

Верный ответ: Запускает контейнер в фоновом режиме

Что делает команда systemctl daemon-reload?

- a) Запускает сервис
- b) Останавливает сервис
- c) Запускает автозагрузку сервиса
- d) Перечитывает unit-файлы

Верный ответ: Перечитывает unit-файлы

Рисунок 73: тест

6.23 Тестовые задания

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

Как просмотреть логи контейнера?

- a) journalctl ps имя.service
- b) journalctl -u имя.service
- c) journalctl list имя.service
- d) systemctl enable имя.service

Верный ответ: journalctl -u имя.service

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

- a) Устанавливает жесткое ограничение оперативной памяти (RAM) для контейнера
- b) Гарантирует, что контейнер получит 512 МБ RAM, даже если на сервере недостаточно памяти
- c) Автоматически увеличивает лимит памяти до 512 МБ, если контейнеру не хватает ресурсов
- d) 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 74: тест

6.24 Тестовые задания

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.

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 75: тест

6.25 Тестовые задания

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.

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 76: тест

7. Оценки тестов

Тест по теме «Основы сетевой конфигурации в Linux»

Результат тестирования

Тест пройден

5 из 5

Тест по теме «Базовая диагностика сети»

Результат тестирования

Тест пройден

4 из 4

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

Результат тестирования

Тест пройден

5 из 5

Тест по теме «Повышение безопасности сетевого взаимодействия»

Результат тестирования

Тест пройден

4 из 5

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

Результат тестирования

Тест пройден

2 из 3

Тест по теме «Обновление системы и безопасность»

Результат тестирования

Тест пройден

3 из 3

Тест по теме «Работа с зависимостями и решение конфликтов»

Результат тестирования

Тест пройден

3 из 3

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

Результат тестирования

Тест пройден

1 из 3

81/92

Тест по теме «Знакомство с логами»

Результат тестирования

Тест пройден

3 из 3

Тест по теме «Система логирования в Linux»

Результат тестирования

Тест пройден

3 из 3

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

Результат тестирования

Тест пройден

3 из 3

Тест по теме «Расследование инцидентов по логам»

Результат тестирования

Тест пройден

3 из 3

Тест по теме «Управление жизненным циклом логов и ротация»

Результат тестирования

Тест пройден

3 из 3

Тест по теме «Контейнеризация как подход»

Результат тестирования

Тест пройден

5 из 5

Тест по теме «Работа с контейнерами в Podman»

Результат тестирования

Тест пройден

5 из 5

Тест по теме «Управление ресурсами контейнеров»

Результат тестирования

Тест пройден

5 из 5

Тест по теме «Образы, реестры и базовая безопасность»

Результат тестирования

Тест пройден

4 из 4

8. Выводы

8.1 Выводы

Мы выполнили третий раздел внешнего курса “Системный администратор Linux с нуля”.

Список литературы

Список литературы

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