

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКИЙ УНИВЕРСИТЕТ
ДРУЖБЫ НАРОДОВ**

Факультет физико-математических и естественных наук

Кафедра информационных технологий

ОТЧЕТ по лабораторной работе 6

ТЕМА «Мандатное разграничение прав в Linux»

по дисциплине «Информационная безопасность»

Выполнил:

Студент группы НПИбд-02-21

Студенческий билет № 1032205641

Сатлихана Петрити

Список содержания

[Список содержания.](#)

[Список изображений](#)

[Цель работы.](#)

[Последовательность выполнения работы](#)

[Выводы](#)

Список изображений

[рис. 1 Политика SELinux и режим Enforcing. 5](#)

[рис. 2 Установка и настройка веб-сервера Apache. 5](#)

[рис. 3 Добавьте строку. 6](#)

[рис. 4 Настройка пакета фильтра \(iptables\) 6](#)

[рис. 5 Выбор браузера. 6](#)

[рис. 6 Вход в систему и проверка режима SELinux. 7](#)

[рис. 7 статус веб-сервера. 7](#)

[рис. 8 Если сервер не запущен, запустите его. 7](#)

[рис. 9 Определение контекста безопасности веб-сервера Apache. 8](#)

[рис. 10 Просмотр переключателей SELinux для Apache. 8](#)

[рис. 11 Просмотр статистики по политике SELinux. 8](#)

[рис. 12 Определение типов файлов и директорий в /var/www.. 9](#)

[рис. 13 Определение типов файлов в /var/www/html 9](#)

[рис. 14 Проверка прав создания файлов в директории /var/www/html 9](#)

[рис. 15 Создание HTML-файла. 9](#)

[рис. 16 Проверка контекста созданного файла. 9](#)

[рис. 17 файл отображается корректно. 10](#)

[рис. 18 Изменение контекста файла. 10](#)

[рис. 19 Проверка доступа к файлу после изменения контекста. 11](#)

[рис. 20 Анализ логов. 11](#)

[рис. 21 Запуск Apache на порту 81. 11](#)

[рис. 22 Перезапуск веб-сервера. 12](#)

[рис. 23 Анализ логов после изменения порта. 12](#)

[рис. 24 Добавление нового порта в SELinux. 12](#)

[рис. 25 Восстановление контекста файла. 13](#)

[рис. 26 Перезапуск веб-сервера Apache. 13](#)

[рис. 27 Возвращение Apache на порт 80. 13](#)

[рис. 28 Удаление порта 81 из списка SELinux. 13](#)

[рис. 29 Удаление файла 13](#)

Цель работы

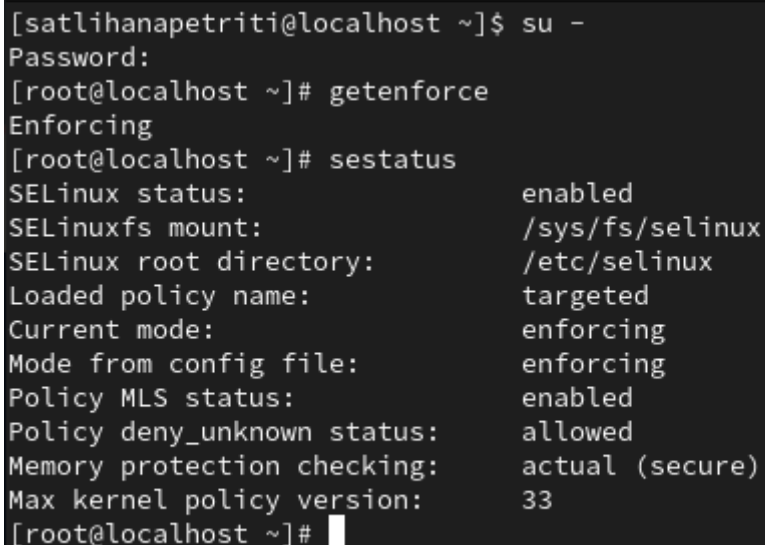
Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux.

Проверить работу SELinux на практике совместно с веб-сервером Apache.

Последовательность выполнения работы

6.3. Подготовка лабораторного стенда и методические рекомендации

1. При подготовке стенда обратите внимание, что необходимая для работы и указанная выше политика `targeted` и режим `enforcing` используются в данном дистрибутиве по умолчанию, т.е. каких-то специальных настроек не требуется. При этом следует убедиться, что политика и режим включены, особенно когда работа будет проводиться повторно и велика вероятность изменений при предыдущем использовании системы



```
[satlihanapetrity@localhost ~]$ su -
Password:
[root@localhost ~]# getenforce
Enforcing
[root@localhost ~]# sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:       /etc/selinux
Loaded policy name:            targeted
Current mode:                  enforcing
Mode from config file:         enforcing
Policy MLS status:             enabled
Policy deny_unknown status:    allowed
Memory protection checking:    actual (secure)
Max kernel policy version:     33
[root@localhost ~]#
```

рис. 1 Политика SELinux и режим Enforcing

2. При необходимости администратор должен разбираться в работе SELinux и уметь как исправить конфигурационный файл

`/etc/selinux/config`, так и проверить используемый режим и политику.

3. Необходимо, чтобы был установлен веб-сервер Apache. При установке

системы в конфигурации «рабочая станция» указанный пакет не ставится.

```
[root@localhost ~]# yum install httpd
Rocky Linux 9 - BaseOS                4.8 kB/s | 4.1 kB      00:00
Rocky Linux 9 - BaseOS                1.3 MB/s | 2.3 MB      00:01
Rocky Linux 9 - AppStream              5.1 kB/s | 4.5 kB      00:00
Rocky Linux 9 - AppStream              3.8 MB/s | 8.0 MB      00:02
Rocky Linux 9 - Extras                  5.0 kB/s | 2.9 kB      00:00
Dependencies resolved.
=====
Package                               Arch      Version                               Repository      Size
=====
Installing:
httpd                                  x86_64    2.4.57-11.el9_4.1                   appstream       44 k
```

рис. 2 Установка и настройка веб-сервера Apache

4. В конфигурационном файле /etc/httpd/httpd.conf необходимо задать параметр ServerName:

ServerName test.ru чтобы при запуске веб-сервера не выдавались лишние сообщения об ошибках, не относящихся к лабораторной работе.

```
# it explicitly to prevent problems during
#
# If your host doesn't have a registered DN
#
ServerName test.ru
#
# Deny access to the entirety of your server
```

рис. 3 Добавьте строку

5. Также необходимо проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp. Отключить фильтр можно командами iptables -F
- iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT либо добавить разрешающие правила:
- iptables -I INPUT -p tcp --dport 80 -j ACCEPT
- iptables -I INPUT -p tcp --dport 81 -j ACCEPT
- iptables -I OUTPUT -p tcp --sport 80 -j ACCEPT
- iptables -I OUTPUT -p tcp --sport 81 -j ACCEPT

```
[root@localhost ~]# iptables -F
[root@localhost ~]# iptables -p INPUT ACCEPT
iptables v1.8.10 (nf_tables): unknown protocol "input" specified
Try `iptables -h' or 'iptables --help' for more information.
[root@localhost ~]# iptables -F
[root@localhost ~]# iptables -P INPUT ACCEPT
[root@localhost ~]# iptables -P OUTPUT ACCEPT
[root@localhost ~]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
[root@localhost ~]# iptables -I INPUT -p tcp --dport 81 -j ACCEPT
[root@localhost ~]# iptables -I OUTPUT -p tcp --sport 80 -j ACCEPT
[root@localhost ~]# iptables -I OUTPUT -p tcp --sport 81 -j ACCEPT
[root@localhost ~]#
```

рис. 4 Настройка пакета фильтра (iptables)

6. Обратите внимание, что данные правила не являются «точными» и рекомендуемыми на все случаи жизни, они лишь позволяют правильно организовать работу стенда.
7. В работе специально не делается акцент, каким браузером (или какой консольной программой) будет производиться подключение к веб-серверу. По желанию могут использоваться разные программы, такие

как консольные `links`, `lynx`, `wget` и графические `konqueror`, `opera`, `firefox` или др.

```
[root@localhost ~]# yum install lynx
Last metadata expiration check: 0:12:11 ago on Mon 07 Oct 2024 04:11:51 PM CEST.
Dependencies resolved.
=====
Package      Architecture Version                      Repository      Size
=====
Installing:
lynx         x86_64        2.8.9-20.el9                appstream       1.5 M
Transaction Summary
=====
Install 1 Package

Total download size: 1.5 M
Installed size: 6.1 M
Is this ok [y/N]: y
Downloading Packages:
lynx-2.8.9-20.el9.x86_64.rpm 1.6 MB/s | 1.5 MB 00:00
```

рис. 5 Выбор браузера

6.4. Порядок выполнения работы

1. Войдите в систему с полученными учётными данными и убедитесь, что

SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`.

```
[root@localhost ~]# getenforce
Enforcing
[root@localhost ~]# sestatus
SELinux status:                enabled
SELinuxfs mount:               /sys/fs/selinux
SELinux root directory:        /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:         enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33
[root@localhost ~]#
```

рис. 6 Вход в систему и проверка режима SELinux

2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает:

`service httpd status`

или

/etc/rc.d/init.d/httpd status

Если не работает, запустите его так же, но с параметром start.

```
[root@localhost ~]# service httpd status
Redirecting to /bin/systemctl status httpd.service
• httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Mon 2024-10-07 16:39:27 CEST; 15min ago
     Docs: man:httpd.service(8)
   Main PID: 35851 (httpd)
  Status: "Total requests: 1; Idle/Busy workers 100/0; Requests/sec: 0.00108; Bytes served/sec: 0 B/s"
    Tasks: 177 (limit: 10966)
   Memory: 21.7M
      CPU: 993ms
   CGroup: /system.slice/httpd.service
           └─35851 /usr/sbin/httpd -DFOREGROUND
             └─35852 /usr/sbin/httpd -DFOREGROUND
```

рис. 7 статус веб-сервера

```
[root@localhost ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@localhost ~]#
```

рис. 8 Если сервер не запущен, запустите его

3. Найдите веб-сервер Apache в списке процессов, определите его контекст

безопасности и занесите эту информацию в отчёт. Например, можно использовать команду

```
ps auxZ | grep httpd
```

или

```
ps -eZ | grep httpd
```

```
[root@localhost ~]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 35851 0.0 0.6 20152 11304 ? Ss 16:39 0:00
bin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35852 0.0 0.3 22032 6972 ? S 16:39 0:00
bin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35853 0.0 0.6 981452 10912 ? Sl 16:39 0:00
bin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35854 0.0 0.7 1112588 13288 ? Sl 16:39 0:00
bin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35855 0.0 0.6 981452 11808 ? Sl 16:39 0:00
bin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 36317 0.0 0.4 236780 8832 pts/0 T 16:5
0 /bin/systemctl status httpd.service
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 36339 0.0 0.4 236780 8832 pts/0 T 16:5
0 /bin/systemctl status httpd.service
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 36350 0.0 0.1 221796 2304 pts/0 S+ 16:
00 grep --color=auto httpd
[root@localhost ~]# ps -eZ | grep httpd
system_u:system_r:httpd_t:s0 35851 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 35852 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 35853 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 35854 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 35855 ? 00:00:00 httpd
[root@localhost ~]#
```

рис. 9 Определение контекста безопасности веб-сервера Apache

4. Посмотрите текущее состояние переключателей SELinux для Apache с

помощью команды

```
sestatus -bigrep httpd
```

```

[abrt@localhost ~]$ sestatus -b httpd
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:       /etc/selinux
Loaded policy name:            targeted
Current mode:                  enforcing
Mode from config file:        enforcing
Policy MLS status:             enabled
Policy deny_unknown status:    allowed
Memory protection checking:    actual (secure)
Max kernel policy version:    33

Policy booleans:
abrt_anon_write                off
abrt_handle_event              off
abrt_upload_watch_anon_write   on
antivirus_can_scan_system      off
antivirus_use_jit              off
auditadm_exec_content          on
authlogin_nsswitch_use_ldap     off
authlogin_radius               off
authlogin_yubikey              off
awstats_purge_apache_log_files off
boinc_execmem                  on
cdrecord_read_content          off

```

рис. 10 Просмотр переключателей SELinux для Apache

Обратите внимание, что многие из них находятся в положении «off».

5. Посмотрите статистику по политике с помощью команды `seinfo`, также определите множество пользователей, ролей, типов.

```

[abrt@localhost ~]$ seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:            33 (MLS enabled)
Target Policy:              selinux
Handle unknown classes:    allow

Classes:                    135      Permissions:                457
Sensitivities:              1        Categories:                1024
Types:                      5145     Attributes:                 259
Users:                      8         Roles:                     15
Booleans:                   356       Cond. Expr.:               388
Allow:                      65500     Neverallow:                 0
Auditallow:                 176       Dontaudit:                 8682
Type_trans:                 271770    Type_change:                94
Type_member:                 37        Range_trans:               5931
Role allow:                  40        Role_trans:                417
Constraints:                70        Validatetrans:             0
MLS Constrains:             72        MLS Val. Tran:             0
Permissives:                4         Polcap:                    6
Defaults:                   7         Typebounds:                0
Allowxperm:                 0         Neverallowxperm:           0
Auditallowxperm:            0         Dontauditxperm:           0
Ibendportcon:               0         Ibpkeycon:                 0
Initial SIDs:               27        Fs_use:                    35
Genfscon:                   109       Portcon:                   665
Netifcon:                   0         Nodecon:                   0

```

рис. 11 Просмотр статистики по политике SELinux

6. Определите тип файлов и поддиректорий, находящихся в директории

/var/www, с помощью команды

ls -lZ /var/www

```
NetIcon: 0 Nodecon: 0
[root@localhost ~]# ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Aug  8 18:30 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Aug  8 18:30 html
[root@localhost ~]#
```

рис. 12 Определение типов файлов и директорий в /var/www

7. Определите тип файлов, находящихся в директории /var/www/html:

ls -lZ /var/www/html

```
drwxr-xr-x. 2 root root system_u:object_r:httpd_
[root@localhost ~]# ls -lZ /var/www/html
total 0
[root@localhost ~]#
```

рис. 13 Определение типов файлов в /var/www/html

8. Определите круг пользователей, которым разрешено создание файлов в

директории /var/www/html.

```
total 0
[root@localhost ~]# ls -ld /var/www/html
drwxr-xr-x. 2 root root 6 Aug  8 18:30 /var/www/html
```

рис. 14 Проверка прав создания файлов в директории /var/www/html

9. Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл

/var/www/html/test.html следующего содержания:

test

```
root@localhost:~
GNU nano 5.6.1 /var/www/html/test.html
<html>
<body>test</body>
</html>
```

рис. 15 Создание HTML-файла

10. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html.


```
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@localhost ~]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@localhost ~]#
```

рис. 16 Проверка контекста созданного файла

11. Обратитесь к файлу через веб-сервер, введя в браузере адрес

<http://127.0.0.1/test.html>. Убедитесь, что файл был успешно отображён.



рис. 17 файл отображается корректно

12. Изучите справку `man httpd_selinux` и выясните, какие контексты файлов определены для `httpd`. Сопоставьте их с типом файла

`test.html`. Проверить контекст файла можно командой `ls -Z`.

`ls -Z /var/www/html/test.html`

Рассмотрим полученный контекст детально. Обратите внимание, что так как по умолчанию пользователи CentOS являются свободными от типа (`unconfined` в переводе с англ. означает свободный), созданному нами файлу `test.html` был сопоставлен SELinux, пользователь `unconfined_u`.

Это первая часть контекста.

Далее политика ролевого разделения доступа RBAC используется процессами, но не файлами, поэтому роли не имеют никакого значения для

файлов. Роль `object_r` используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах. (В директории

`/proc` файлы, относящиеся к процессам, могут иметь роль `system_r`.

Если активна политика MLS, то могут использоваться и другие роли, например, `secadm_r`. Данный случай мы рассматривать не будем, как и предназначение `:s0`).

Тип `httpd_sys_content_t` позволяет процессу `httpd` получить доступ к файлу. Благодаря наличию последнего типа мы получили доступ к файлу при обращении к нему через браузер.

13. Измените контекст файла `/var/www/html/test.html` с

`httpd_sys_content_t` на любой другой, к которому процесс `httpd` не

должен иметь доступа, например, на `samba_share_t`: `chcon -t samba_share_t /var/www/html/test.html ls -Z /var/www/html/test.html`.

После этого проверьте, что контекст поменялся.

```
[root@localhost ~]# chcon -t samba_share_t /var/www/html/test.html
[root@localhost ~]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
[root@localhost ~]#
```

рис. 18 Изменение контекста файла

14. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Вы должны получить сообщение об ошибке:

Forbidden

You don't have permission to access /test.html on this server.

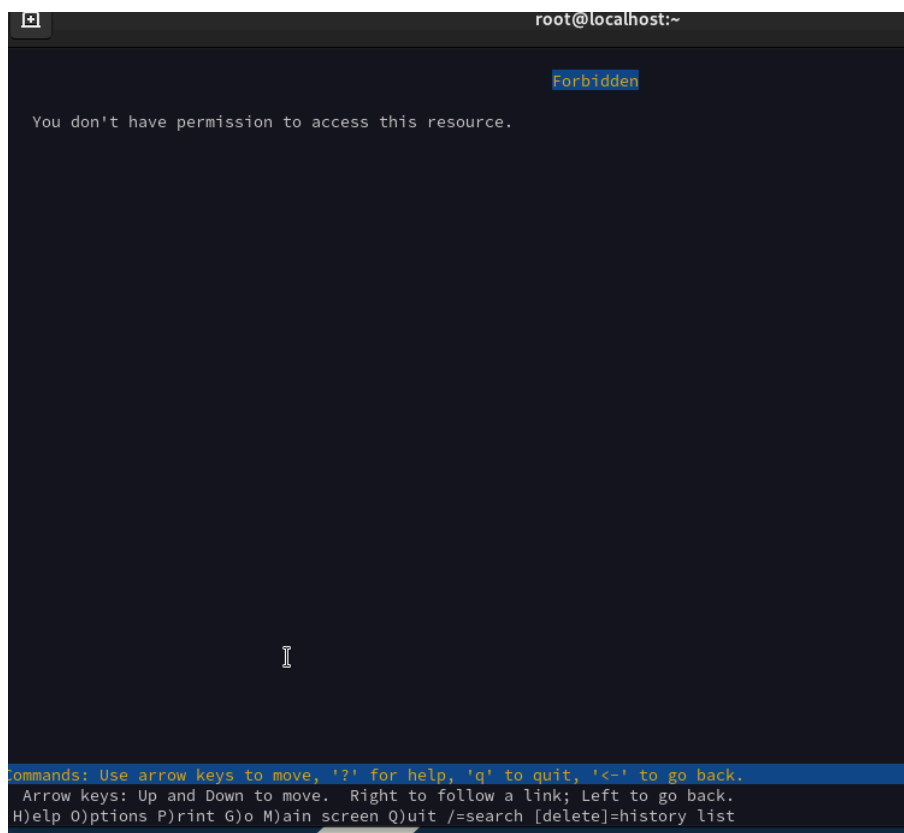


рис. 19 Проверка доступа к файлу после изменения контекста

15. Проанализируйте ситуацию. Почему файл не был отображён, если права

доступа позволяют читать этот файл любому пользователю?

ls -l /var/www/html/test.html Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл:

tail /var/log/messages Если в системе окажутся запущенными процессы setroubleshootd и auditd, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log. Проверьте это утверждение самостоятельно.

```
[root@localhost ~]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct  7 17:51 /var/www/html/test.html
[root@localhost ~]# tail /var/log/messages
Oct  7 18:03:37 localhost systemd[1]: Created slice Slice /system/dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged.
Oct  7 18:03:37 localhost systemd[1]: Started dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@0.service.
Oct  7 18:03:39 localhost setroubleshoot[36631]: SELinux is preventing /usr/sbin/httpd from getting access on the file /var/www/html/test.html. For complete SELinux messages run: sealert -l a165a3b1-7aa8-4caa-a015-83765690d18c
Oct  7 18:03:39 localhost setroubleshoot[36631]: SELinux is preventing /usr/sbin/httpd from getting access on the file /var/www/html/test.html.#012#012***** Plugin restorecon (92.2 confidence) suggests *****#012#012If you want to fix the label. #012/var/www/html/test.html
```

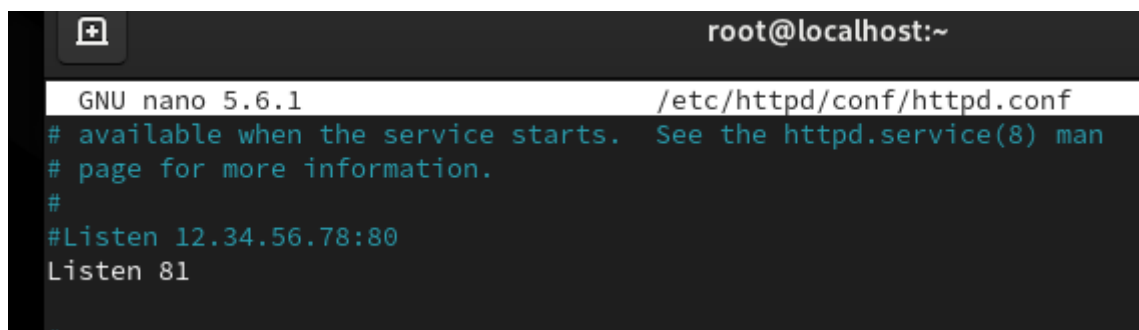
рис. 20 Анализ логов

16. Попробуйте запустить веб-сервер Apache на прослушивание TCP-порта

81 (а не 80, как рекомендует IANA и прописано в /etc/services). Для

этого в файле /etc/httpd/httpd.conf найдите строку Listen 80 и

замените её на Listen 81.



```
root@localhost:~
GNU nano 5.6.1 /etc/httpd/conf/httpd.conf
# available when the service starts. See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 81
```

рис. 21 Запуск Apache на порту 81

17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему?

```
[root@localhost ~]# nano /etc/httpd/conf/httpd.conf
[root@localhost ~]# systemctl restart httpd
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Mon 2024-10-07 18:11:13 CEST; 12s ago
     Docs: man:httpd.service(8)
  Main PID: 36670 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0"
    Tasks: 177 (limit: 10966)
   Memory: 21.6M
      CPU: 118ms
    CGroup: /system.slice/httpd.service
            └─36670 /usr/sbin/httpd -DFOREGROUND
              └─36672 /usr/sbin/httpd -DFOREGROUND
                └─36673 /usr/sbin/httpd -DFOREGROUND
                  └─36674 /usr/sbin/httpd -DFOREGROUND
                    └─36675 /usr/sbin/httpd -DFOREGROUND

Oct 07 18:11:13 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Oct 07 18:11:13 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Oct 07 18:11:13 localhost.localdomain httpd[36670]: Server configured, listening on: port 81
```

рис. 22 Перезапуск веб-сервера

18. Проанализируйте лог-файлы:

`tail -nl /var/log/messages`

`/var/log/httpd/access_log`

```
[root@localhost ~]# tail -nl /var/log/messages
Oct  7 18:12:37 localhost systemd[1]: packagekit.service: Deactivated successfully.
[root@localhost ~]# sudo tail -nl /var/log/httpd/error_log
[Mon Oct 07 18:11:13.592769 2024] [core:notice] [pid 36670:tid 36670] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[root@localhost ~]# sudo tail -nl /var/log/httpd/access_log
127.0.0.1 - - [07/Oct/2024:18:03:35 +0200] "GET /test.html HTTP/1.0" 403 199 "-" "Lynx/2.8.9rel.1 libwww-FM/2.14 SSL-MM/1.4.1 OpenSSL/3.0.7"
[root@localhost ~]# sudo tail -nl /var/log/audit/audit.log
type=USER_START msg=audit(1728318047.629:408): pid=36885 uid=0 auid=1000 ses=3 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam_keyinit,pam_limits,pam_systemd,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=sucesss'UID="root" AUID="satlihanapetrity"
[root@localhost ~]#
```

рис. 23 Анализ логов после изменения порта

19. Выполните команду

`semanage port -a -t http_port_t -p tcp 81`

После этого проверьте список портов командой `semanage port -l | grep http_port_t` Убедитесь, что порт 81 появился в списке.

```
[root@localhost ~]# semmanage port -a -t http_port_t -p tcp 81
Port tcp/81 already defined, modifying instead
[root@localhost ~]# semange port -l | grep http_port_t
bash: semange: command not found...
[root@localhost ~]# semange port -l | grep http_port_t
bash: semange: command not found...
[root@localhost ~]# semange port -l | grep http_port_t
http_port_t          tcp      81, 80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t  tcp      5988
[root@localhost ~]#
```

рис. 24 Добавление нового порта в SELinux

20. Попробуйте запустить веб-сервер Apache ещё раз.

<http://127.0.0.1:81/test.html>

21. Верните контекст `httpd_sys_content_t` к файлу `/var/www/html/ test.html`:

```
chcon -t httpd_sys_content_t /var/www/html/test.html
```

После этого попробуйте получить доступ к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1:81/test.html>.

Вы должны увидеть содержимое файла — слово «test».

```
[root@localhost ~]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@localhost ~]# lynx http://127.0.0.1:81/test.html
[root@localhost ~]#
```

рис. 25 Восстановление контекста файла

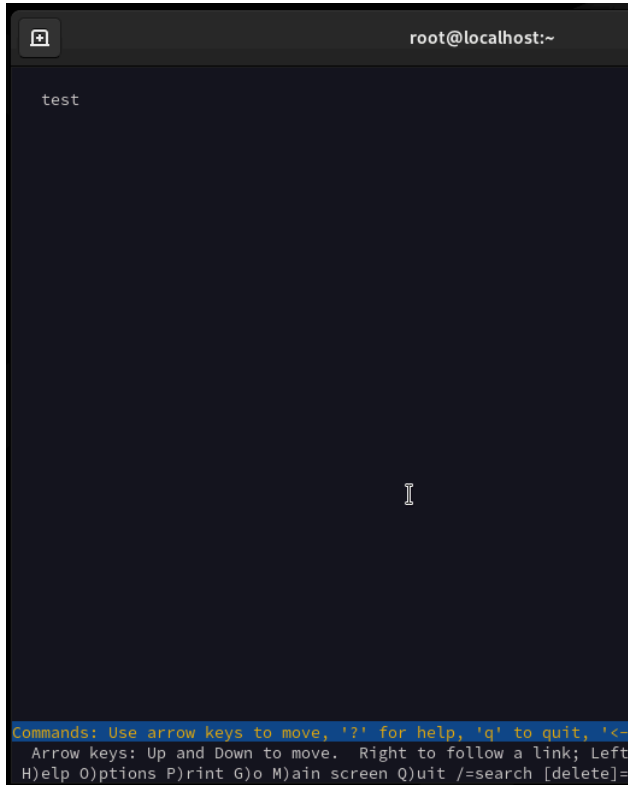


рис. 26 Перезапуск веб-сервера Apache

22. Исправьте обратно конфигурационный файл `apache`, вернув `Listen 80`.

```
# available when the service starts. See the
# page for more information.
#
#Listen 12.34.56.78:80
Listen 80
```

рис. 27 Возвращение Apache на порт 80

23. Удалите привязку `http_port_t` к 81 порту:

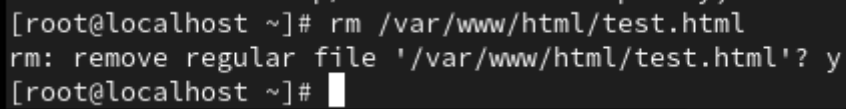
`semanage port -d -t http_port_t -p tcp 81` и проверьте, что порт 81 удалён.

```
[root@localhost ~]# sudo semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
[root@localhost ~]#
```

рис. 28 Удаление порта 81 из списка SELinux

24. Удалите файл /var/www/html/test.html:

```
rm /var/www/html/test.html
```



```
[root@localhost ~]# rm /var/www/html/test.html
rm: remove regular file '/var/www/html/test.html'? y
[root@localhost ~]#
```

рис. 29 Удаление файла

Вывод

В этой лабораторной работе мы исследовали конфигурацию и механизмы безопасности веб-сервера Apache в среде SELinux. Мы убедились, что служба Apache активна и правильно настроена для прослушивания определённых портов, таких как 80 и 81.

Изучая контексты SELinux и разрешения пользователей, мы поняли, как управляются доступы к файлам и каталогам в корневом каталоге веб-сервера. Мы столкнулись с проблемами, связанными с политиками SELinux, что подчеркнуло важность понимания взаимодействия SELinux с сетевыми службами.