

Отчет по лабораторной работе №6

Основы информационной безопасности

Швецов Михаил, НКАбд-01-23

Содержание

1	Цель работы.....	1
2	Теоретическое введение.....	1
3	Выполнение лабораторной работы.....	2
4	Выводы	11
	Список литературы	11

1 Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux¹. Проверить работу SELinux на практике совместно с веб-сервером Apache. [1]

2 Теоретическое введение

1. **SELinux (Security-Enhanced Linux)** обеспечивает усиление защиты путем внесения изменений как на уровне ядра, так и на уровне пространства пользователя, что превращает ее в действительно «непробиваемую» операционную систему. Впервые эта система появилась в четвертой версии CentOS, а в 5 и 6 версии реализация была существенно дополнена и улучшена.

SELinux имеет три основных режим работы:

- Enforcing: режим по умолчанию. При выборе этого режима все действия, которые каким-то образом нарушают текущую политику безопасности, будут блокироваться, а попытка нарушения будет зафиксирована в журнале.
- Permissive: в случае использования этого режима, информация о всех действиях, которые нарушают текущую политику безопасности, будут зафиксированы в журнале, но сами действия не будут заблокированы.
- Disabled: полное отключение системы принудительного контроля доступа.

Политика SELinux определяет доступ пользователей к ролям, доступ ролей к доменам и доступ доменов к типам. Контекст безопасности — все атрибуты SELinux — роли, типы и домены. Более подробно см. в [2].

2. **Apache** — это свободное программное обеспечение, с помощью которого можно создать веб-сервер. Данный продукт возник как доработанная версия другого HTTP-клиента от национального центра суперкомпьютерных приложений (NCSA).

Для чего нужен Apache сервер:

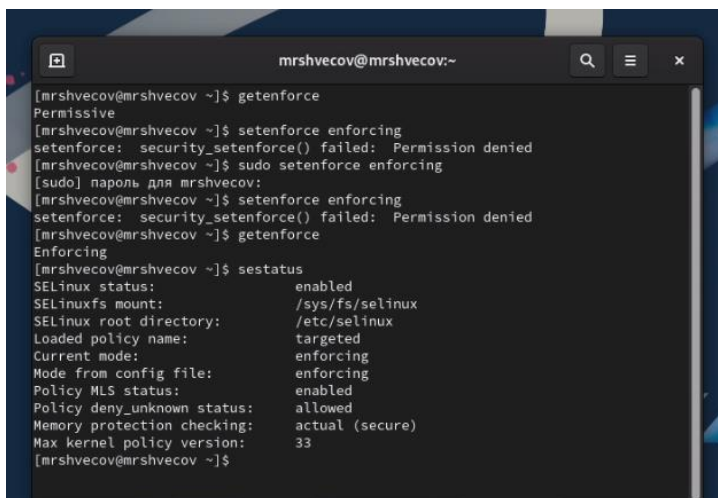
- чтобы открывать динамические PHP-страницы,
- для распределения поступающей на сервер нагрузки,
- для обеспечения отказоустойчивости сервера,
- чтобы потренироваться в настройке сервера и запуске PHP-скриптов.

Apache является кроссплатформенным ПО и поддерживает такие операционные системы, как Linux, BSD, MacOS, Microsoft, BeOS и другие.

Более подробно см. в [3].

3 Выполнение лабораторной работы

Вошла в систему под своей учетной записью. Убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (рис. 1).



```
mrshvecov@mrshvecov:~$ getenforce
Permissive
[mrshvecov@mrshvecov ~]$ setenforce enforcing
setenforce: security_setenforce() failed: Permission denied
[mrshvecov@mrshvecov ~]$ sudo setenforce enforcing
[sudo] пароль для mrshvecov:
[mrshvecov@mrshvecov ~]$ setenforce enforcing
setenforce: security_setenforce() failed: Permission denied
[mrshvecov@mrshvecov ~]$ getenforce
Enforcing
[mrshvecov@mrshvecov ~]$ 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
[mrshvecov@mrshvecov ~]$
```

Рис. 1: проверка режима работы SELinux

Запускаю сервер apache, далее обращаюсь с помощью браузера к веб-серверу, запущенному на компьютере, он работает, что видно из вывода команды `service httpd status` (рис. 2).

```
mrshvecov@mrshvecov:~ — /bin/systemctl status httpd.service
/lib/systemd/system/httpd.service.
[mrshvecov@mrshvecov ~]$ 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 2025-09-15 06:31:54 MSK; 18s ago
     Docs: man:httpd.service(8)
   Main PID: 25108 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes"
    Tasks: 177 (limit: 12146)
   Memory: 24.0M
      CPU: 92ms
   CGroup: /system.slice/httpd.service
           └─25108 /usr/sbin/httpd -DFOREGROUND
             └─27417 /usr/sbin/httpd -DFOREGROUND
               └─27418 /usr/sbin/httpd -DFOREGROUND
                 └─27421 /usr/sbin/httpd -DFOREGROUND
                   └─27423 /usr/sbin/httpd -DFOREGROUND

сен 15 06:31:51 mrshvecov systemd[1]: Starting The Apache HTTP Server...
сен 15 06:31:53 mrshvecov httpd[25108]: AH00558: httpd: Could not reliably dete
сен 15 06:31:54 mrshvecov httpd[25108]: Server configured, listening on: port 80
сен 15 06:31:54 mrshvecov systemd[1]: Started The Apache HTTP Server.
lines 1-20/20 (END)
```

Рис. 2: Проверка работы Apache

С помощью команды `ps auxZ | grep httpd` нашла веб-сервер Apache в списке процессов. Его контекст безопасности - `httpd_t` (рис. 3).

```
mrshvecov@mrshvecov:~
└─27423 /usr/sbin/httpd -DFOREGROUND

сен 15 06:31:51 mrshvecov systemd[1]: Starting The Apache HTTP Server...
сен 15 06:31:53 mrshvecov httpd[25108]: AH00558: httpd: Could not reliably dete
сен 15 06:31:54 mrshvecov httpd[25108]: Server configured, listening on: port 80
сен 15 06:31:54 mrshvecov systemd[1]: Started The Apache HTTP Server.

[1]+  Остановлен service httpd status
[mrshvecov@mrshvecov ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 25108 0.1 0.6 23356 12536 ?
Ss 06:31 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 27417 0.0 0.2 23012 5016 ?
S 06:31 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 27418 0.0 0.6 1965340 12376 ?
Sl 06:31 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 27421 0.0 0.6 2096476 12420 ?
Sl 06:31 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 27423 0.0 0.5 1965340 10376 ?
Sl 06:31 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 mrshvec+ 39977 0.0 0.4 23
7628 9088 pts/0 T 06:32 0:00 /bin/systemctl status httpd.service
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 mrshvec+ 42367 0.0 0.1 22
1820 2304 pts/0 S+ 06:32 0:00 grep --color=auto httpd
[mrshvecov@mrshvecov ~]$ sestatus
```

Рис. 3: Контекст безопасности Apache

Просмотрела текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd` (рис. 4).

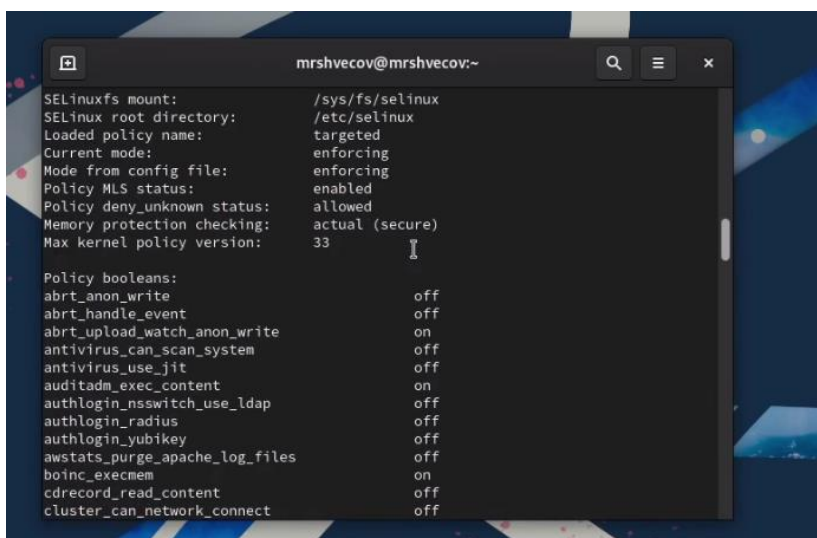


Рис. 4: Состояние переключателей SELinux

Просмотрела статистику по политике с помощью команды `seinfo`. Множество пользователей - 8, ролей - 39, типов - 5135. (рис. 5).



Рис. 5: Статистика по политике

Типы поддиректорий, находящихся в директории `/var/www`, с помощью команды `ls -lZ /var/www` следующие: владелец - `root`, права на изменения только у владельца. Файлов в директории нет (рис. 6).

```
zonefinder_run_sudo
[mrshvecov@mrshvecov ~]$ ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 сен 3 17
:59 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 сен 3 17
:59 html
[mrshvecov@mrshvecov ~]$
```

Рис. 6: Типы поддиректорий

В директории /var/www/html нет файлов. (рис. 7).

```
:59 html
[mrshvecov@mrshvecov ~]$ ls -lZ /var/www/html
итого 0
```

Рис. 7: Типы файлов

Создать файл может только суперпользователь, поэтому от его имени создаем файл touch.html со следующим содержанием:

```
<html>
<body>test</body>
</html>
```

(рис. 8).

```
итого 0
[mrshvecov@mrshvecov ~]$ sudo touch /var/www/html/test.html
[mrshvecov@mrshvecov ~]$ sudo nano /var/www/html/test.html

<html>
<body>test</body>
</html>
```

Рис. 8: Создание файла

Проверяю контекст созданного файла. По умолчанию это httpd_sys_content_t (рис. 9).

```
[mrshvecov@mrshvecov ~]$ sudo nano /var/www/html/test.html
[mrshvecov@mrshvecov ~]$ ls -lZ /var/www/html/
итого 4
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 сен 15 0
6:34 test.html
[mrshvecov@mrshvecov ~]$
```

Рис. 9: Контекст файла

Обращаюсь к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Файл был успешно отображён (рис. 10).

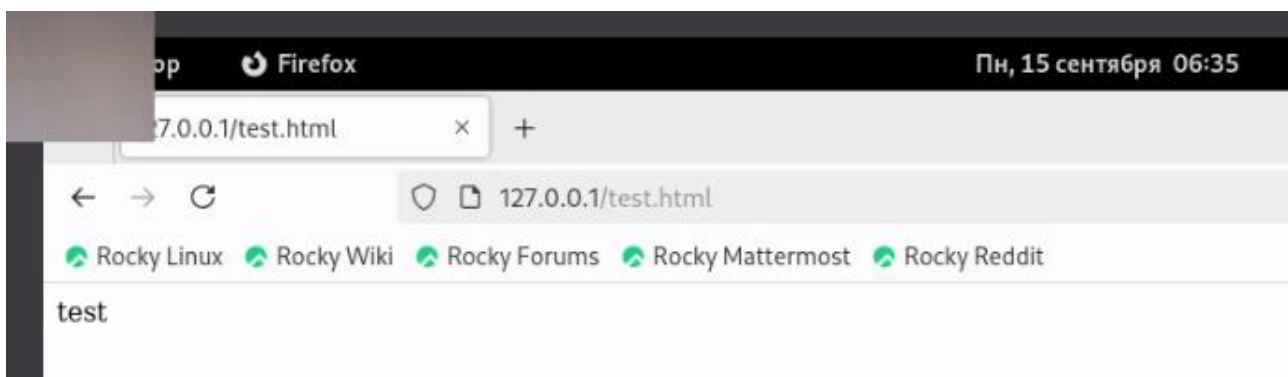


Рис. 10: Отображение файла

Изучила справку `man httpd_selinux`. Рассмотрим полученный контекст детально. 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. (рис. 11).

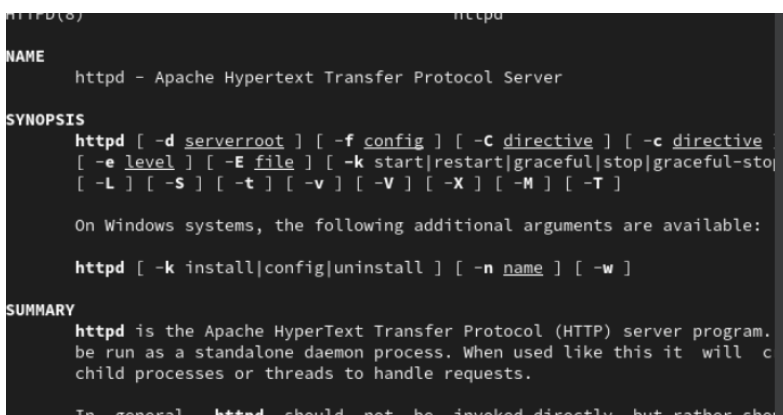


Рис. 11: Изучение справки по команде

Изменяю 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` Контекст действительно поменялся (рис. 12).

```
[mrshvecov@mrshvecov ~]$ sudo chcon -t samba_share_t /var/www/html/test.html
[mrshvecov@mrshvecov ~]$ ls -lZ /var/www/html/
итого 4
-rw-r--r--. 1 root root unconfined_u:object_r:samba_share_t:s0 33 сен 15 06:34 test.html
[mrshvecov@mrshvecov ~]$
```

Рис. 12: Изменение контекста

При попытке отображения файла в браузере получаем сообщение об ошибке (рис. 13).

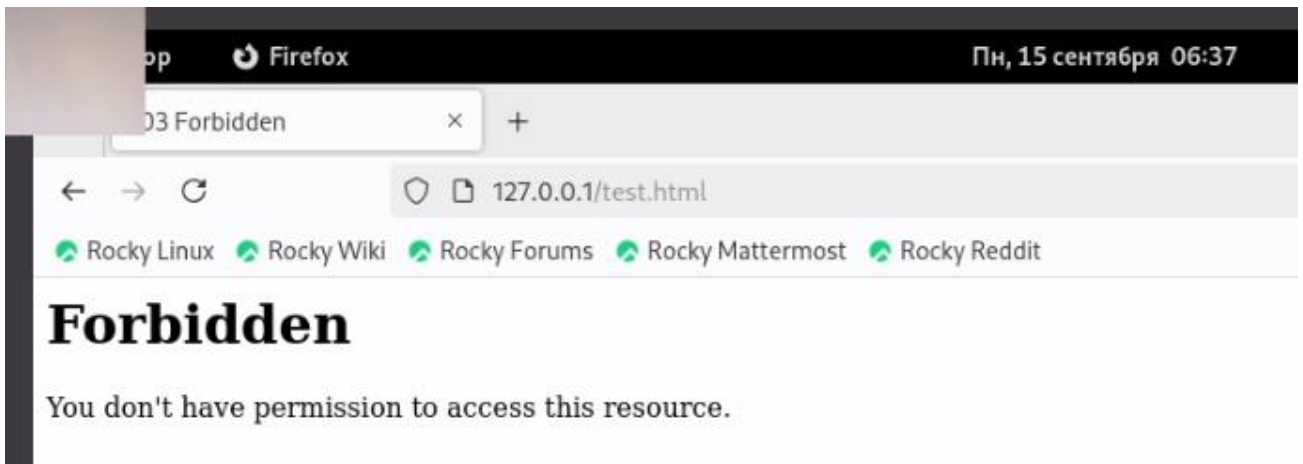
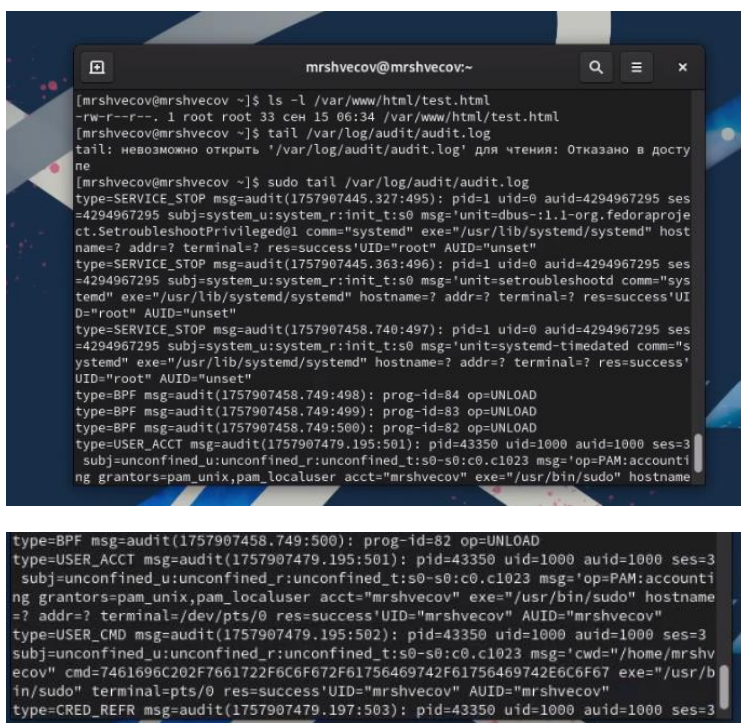


Рис. 13: Отображение файла

файл не был отображён, хотя права доступа позволяют читать этот файл любому пользователю, потому что установлен контекст, к которому процесс httpd не должен иметь доступа.

Просматриваю log-файлы веб-сервера Apache и системный лог-файл: `tail /var/log/messages`. Если в системе окажутся запущенными процессы `setroubleshootd` и `audtd`, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле `/var/log/audit/audit.log`. (рис. 14).



```
mrshvecov@mrshvecov:~$ ls -l /var/www/html/test.html
-rw-r--r-- 1 root root 33 сен 15 06:34 /var/www/html/test.html
[mrshvecov@mrshvecov ~]$ tail /var/log/audit/audit.log
tail: невозможно открыть '/var/log/audit/audit.log' для чтения: Отказано в досту
не
[mrshvecov@mrshvecov ~]$ sudo tail /var/log/audit/audit.log
type=SERVICE_STOP msg=audit(1757907445.327:495): pid=1 uid=0 auid=4294967295 ses
=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=dbus-1.1-0-org.fedoraproje
ct.SetroubleShootPrivileged@1 comm='systemd' exe='/usr/lib/systemd/systemd' host
name=? addr=? terminal=? res=success'UID='root' AUID='unset'
type=SERVICE_STOP msg=audit(1757907445.363:496): pid=1 uid=0 auid=4294967295 ses
=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=setroubleShootd comm='sys
temd' exe='/usr/lib/systemd/systemd' hostname=? addr=? terminal=? res=success'UI
D='root' AUID='unset'
type=SERVICE_STOP msg=audit(1757907458.740:497): pid=1 uid=0 auid=4294967295 ses
=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=systemd-timedated comm='s
ysd' exe='/usr/lib/systemd/systemd' hostname=? addr=? terminal=? res=success'
UID='root' AUID='unset'
type=BPF msg=audit(1757907458.749:498): prog-id=84 op=UNLOAD
type=BPF msg=audit(1757907458.749:499): prog-id=83 op=UNLOAD
type=BPF msg=audit(1757907458.749:500): prog-id=82 op=UNLOAD
type=USER_ACCT msg=audit(1757907479.195:501): pid=43350 uid=1000 auid=1000 ses=3
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:accounti
ng grantors=pam_unix,pam_localuser acct='mrshvecov' exe='/usr/bin/sudo' hostname
type=BPF msg=audit(1757907458.749:500): prog-id=82 op=UNLOAD
type=USER_ACCT msg=audit(1757907479.195:501): pid=43350 uid=1000 auid=1000 ses=3
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:accounti
ng grantors=pam_unix,pam_localuser acct='mrshvecov' exe='/usr/bin/sudo' hostname
=? addr=? terminal=/dev/pts/0 res=success'UID='mrshvecov' AUID='mrshvecov'
type=USER_CMD msg=audit(1757907479.195:502): pid=43350 uid=1000 auid=1000 ses=3
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='cwd='/home/mrshv
ecov' cmd='7461696C202F7661722F6C6F672F61756469742F61756469742E6C6F67 exe='/usr/b
in/sudo' terminal=pts/0 res=success'UID='mrshvecov' AUID='mrshvecov'
type=CRED_REFR msg=audit(1757907479.197:503): pid=43350 uid=1000 auid=1000 ses=3
```

Рис. 14: Попытка прочесть лог-файл

Чтобы запустить веб-сервер Apache на прослушивание TCP-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services) открываю файл /etc/httpd/httpd.conf для изменения. (рис. 15).



```
[mrshvecov@mrshvecov ~]$ sudo nano /etc/httpd/conf/httpd.conf
```

Рис. 15: Изменение файла

Нахожу строчку Listen 80 и заменяю её на Listen 81. (рис. 16).

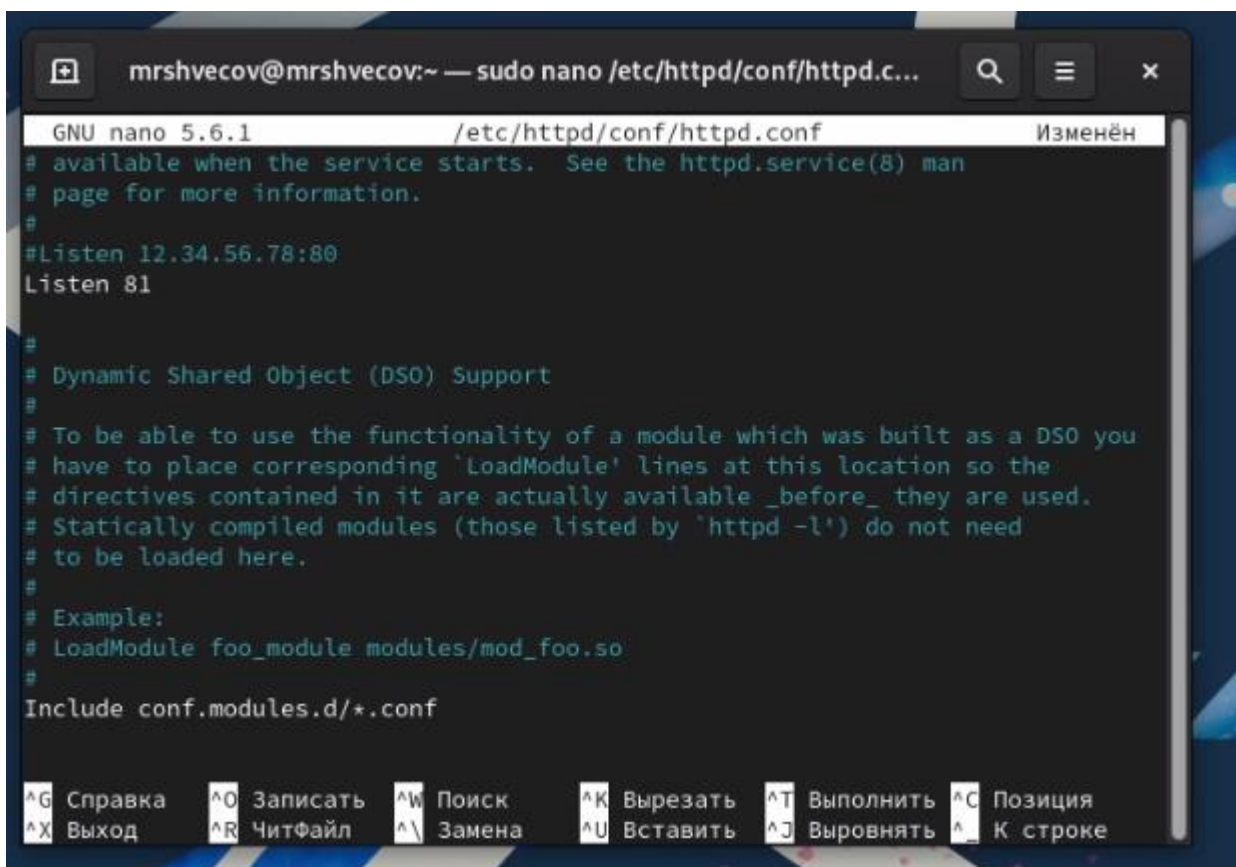


Рис. 16: Изменение порта

Выполняю перезапуск веб-сервера Apache. Произошёл сбой, потому что порт 80 для локальной сети, а 81 нет (рис. 17).

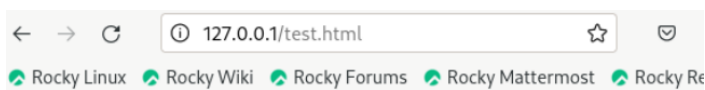


Рис. 17: Попытка прослушивания другого порта

Проанализируйте лог-файлы: `tail -nl /var/log/messages` (рис. 18).

```
[mrshvecov@mrshvecov ~]$ sudo tail -nl /var/log/messages
Sep 15 06:39:27 mrshvecov systemd[1]: systemd-timedated.service: Deactivated successfully.
[mrshvecov@mrshvecov ~]$
```

Рис. 18: Проверка лог-файлов

Просмотрите файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и выясните, в каких файлах появились записи. Запись появилась в файлу `error_log` (рис. 19).

```
[mrshvecov@mrshvecov ~]$ sudo cat /var/log/httpd/error_log
[Mon Sep 15 06:31:53.571068 2025] [core:notice] [pid 25108:tid 25108] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Mon Sep 15 06:31:53.573722 2025] [suexec:notice] [pid 25108:tid 25108] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using fe80::a00:27ff:fea3:2efe%enp0s3. Set the 'ServerName' directive globally to suppress this message
[Mon Sep 15 06:31:54.581251 2025] [lbmethod_heartbeat:notice] [pid 25108:tid 25108] AH02282: No slotmem from mod_heartbeat
[Mon Sep 15 06:31:54.586233 2025] [mpm_event:notice] [pid 25108:tid 25108] AH00489: Apache/2.4.62 (Rocky Linux) configured -- resuming normal operations
[Mon Sep 15 06:31:54.586277 2025] [core:notice] [pid 25108:tid 25108] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[Mon Sep 15 06:35:06.802571 2025] [autoindex:error] [pid 27418:tid 27590] [client 127.0.0.1:48484] AH01276: Cannot serve directory /var/www/html/: No matching DirectoryIndex (index.html) found, and server-generated directory index forbidden by Options directive
[Mon Sep 15 06:37:12.084146 2025] [core:error] [pid 27423:tid 27525] (13)Permission denied: [client 127.0.0.1:32880] AH00035: access to /test.html denied (filesystem path '/var/www/html/test.html') because search permissions are missing on a component of the path
```

Рис. 19: Проверка лог-файлов

Выполняю команду `semanage port -a -t http_port_t -p tcp 81` После этого проверяю список портов командой `semanage port -l | grep http_port_t` Порт 81 появился в списке (рис. 20).

```
[Mon Sep 15 06:39:03.570572 2025] [core:error] [pid 27423:tid 27535] (13)Permission denied: [client 127.0.0.1:32880] AH00035: access to /test.html denied (filesystem path '/var/www/html/test.html') because search permissions are missing on a component of the path
[mrshvecov@mrshvecov ~]$ sudo semanage port -a -t http_port_t -p tcp 81
Port tcp/81 already defined, modifying instead
[mrshvecov@mrshvecov ~]$ semanage port -l | grep http_port_t
ValueError: Политика SELinux не задана, или нет доступа к хранилищу.
[mrshvecov@mrshvecov ~]$ sudo semanage 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
[mrshvecov@mrshvecov ~]$
```

Рис. 20: Проверка портов

Перезапускаю сервер Apache (рис. 21).

```
[mrshvecov@mrshvecov ~]$ sudo systemctl restart httpd
[mrshvecov@mrshvecov ~]$ sudo chcon -t httpd_sys_content_t /var/www/html/test.html

[mrshvecov@mrshvecov ~]$ sudo systemctl restart httpd
```

Рис. 21: Перезапуск сервера

Теперь он работает, ведь мы внесли порт 81 в список портов httpd_port_t (рис. 22).

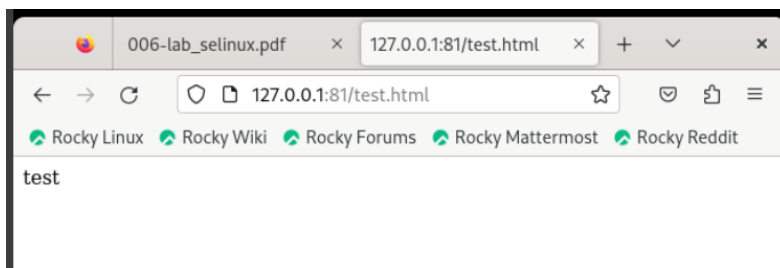


Рис. 22: Проверка сервера

Возвращаю в файле /etc/httpd/httpd.conf порт 80, вместо 81. Проверяю, что порт 81 удален, это правда. (рис. 23).

```
[mrshvecov@mrshvecov ~]$ sudo nano /etc/httpd/conf/httpd.conf
[mrshvecov@mrshvecov ~]$ sudo nano /etc/httpd/conf/httpd.conf
[mrshvecov@mrshvecov ~]$ sudo nano /etc/httpd/conf/httpd.conf
[mrshvecov@mrshvecov ~]$ semanage port -d -t httpd_port_t -p tcp 81
ValueError: Политика SELinux не задана, или нет доступа к хранилищу.
[mrshvecov@mrshvecov ~]$ sudo semanage port -d -t httpd_port_t -p tcp 81
```

Рис. 23: Проверка порта 81

Далее удаляю файл test.html, проверяю, что он удален (рис. 24).

```
[mrshvecov@mrshvecov ~]$ sudo rm /var/www/html/test.html
[mrshvecov@mrshvecov ~]$ ls -lZ /var/www/html/
итого 0
[mrshvecov@mrshvecov ~]$
```

Рис. 24: Удаление файла

4 Выводы

В ходе выполнения данной лабораторной работы были развиты навыки администрирования ОС Linux, получено первое практическое знакомство с технологией SELinux и проверена работа SELinux на практике совместно с веб-сервером Apache.

Список литературы

1. Кулябов Д. С. Г.М.Н. Королькова А. В. Лабораторная работа № 6. Мандатное разграничение прав в Linux [Электронный ресурс]. 2023. URL: https://esystem.rudn.ru/pluginfile.php/2293720/mod_resource/content/2/006-lab_selinux.pdf.

2. SELinux – описание и особенности работы с системой. Часть 1 [Электронный ресурс]. URL: <https://habr.com/ru/companies/kingservers/articles/209644/>.
3. Что такое Apache [Электронный ресурс]. URL: <https://2domains.ru/support/vps-i-servery/shto-takoye-apache>.