

Отчёт по лабораторной работе №6

Мандатное разграничение прав в Linux

Ханина Ирина Владимировна, НБИбд-02-18

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выводы	19
	Список литературы	20

List of Tables

List of Figures

4.1	Рис 1. Установка веб-сервера	9
4.2	Рис 2. Изменение конфигурационного файла /etc/httpd/httpd.conf	10
4.3	Рис 3. Отключение пакетного фильтра	10
4.4	Рис 4. SELinux работает в режиме enforcing политики targeted . . .	11
4.5	Рис 5. Команда service httpd status	11
4.6	Рис 6. Команда ps auxZ grep httpd	12
4.7	Рис 7. Команда sestatus -bigrep httpd	12
4.8	Рис 8. Команды seinfo, ls -lZ /var/www и ls -lZ /var/www/html . . .	13
4.9	Рис 9. Создание файла /var/www/html/test.html	13
4.10	Рис 10. Обращение к файлу через сервер	14
4.11	Рис 11. Изменение контекста файла /var/www/html/test.html . . .	14
4.12	Рис 12. Доступ к файлу запрещен	15
4.13	Рис 13. Команда tail /var/log/messages	15
4.14	Рис 14. Изменение Listen 80 на Listen 81	16
4.15	Рис 15. Просмотр log файлов	16
4.16	Рис 16. Команда semanage port -a -t http_port_t -p tcp 81	16
4.17	Рис 17. Изменение контекста файла /var/www/html/test.html . . .	17
4.18	Рис 18. Обращение к файлу через сервер	17
4.19	Рис 19. Изменение конфигурационного файла /etc/httpd/httpd.conf	18
4.20	Рис 20. Удаление файла /var/www/html/test.html	18

1 Цель работы

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

2 Задание

Формулировка задания представлена в разделе 6.4 “Порядок выполнения работы” в файле “Лабораторная работа № 6. Описание”.

3 Теоретическое введение

Linux с улучшенной безопасностью (SELinux) - это механизм безопасности с мандатной моделью контроля доступа (MAC), реализованный в ядре. Это разграничение контроля доступа внедряется поверх того, что уже есть в каждом дистрибутиве Linux, DAC (Discretionary Access Control). Можно сказать, что SELinux расширяет возможности стандартной системы безопасности. Первоначально он был разработан Агентством национальной безопасности США для защиты компьютерных систем от вторжения злоумышленников и взлома. Со временем SELinux появился в открытом доступе, и тогда различные дистрибутивы включили его в свой код. Он был впервые представлен в CentOS 4 и значительно улучшен в более поздних выпусках CentOS. [1]

Система SELinux – это средство для точной настройки требований контроля доступа. С помощью SELinux можно определить, что позволено делать пользователю или процессу. Она ограничивает каждый процесс своим собственным доменом, поэтому процесс может взаимодействовать только с определенными типами файлов и другими процессами из разрешенных доменов. Это предотвращает взлом любого процесса и получение хакерами общесистемного доступа. [3]

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 дей-

ствиях, которые нарушают текущую политику безопасности, будут зафиксированы в журнале, но сами действия не будут заблокированы.

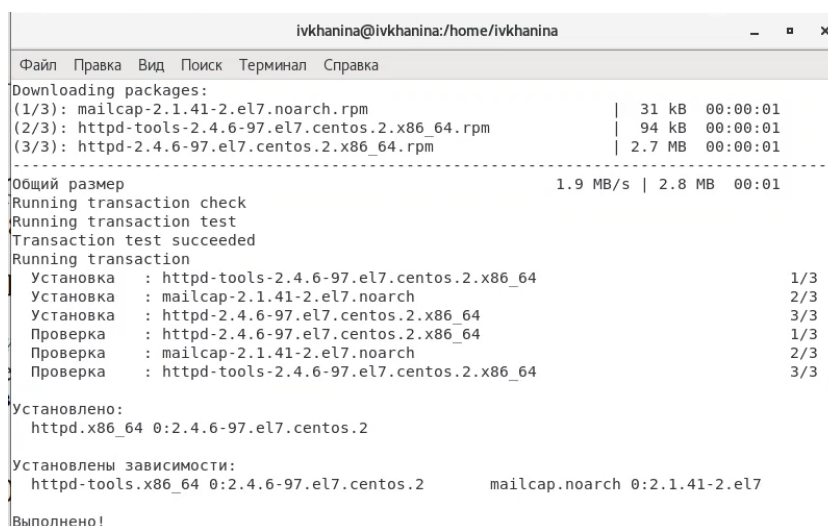
- Disabled: Полное отключение системы принудительного контроля доступа.

[2]

Для просмотра текущего режима и других настроек SELinux используется команда `sestatus`. Узнать статус SELinux можно при помощи команды `getenforce`. Команда `setenforce` позволяет быстро переключаться между режимами Enforcing и Permissive, изменения вступают в силу без перезагрузки. Но если вы включаете или отключаете SELinux, требуется перезагрузка, ведь нужно заново устанавливать метки безопасности в файловой системе. [3]

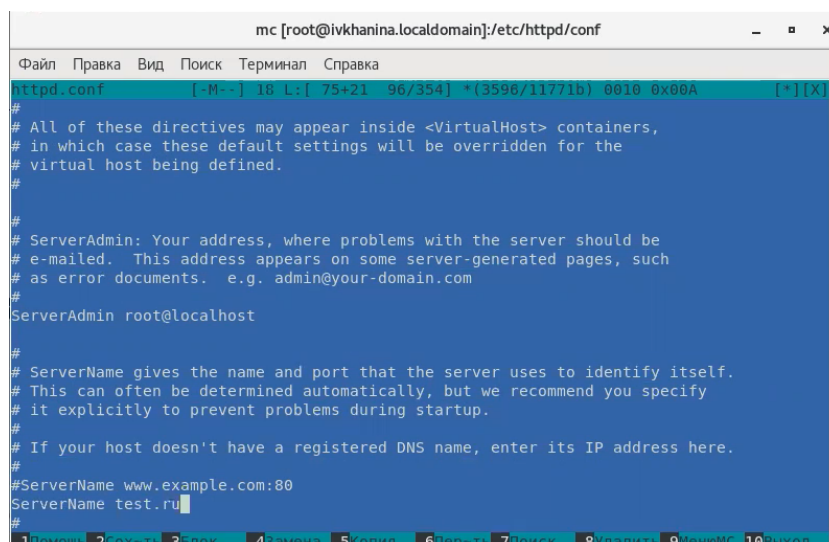
4 Выполнение лабораторной работы

1. Выполнила подготовку лабораторного стенда: вошла в систему от имени суперпользователя и запустила команду для установки веб-сервера Apache: `yum install httpd` (рис. 1), задала параметр `ServerName` в конфигурационном файле `/etc/httpd/httpd.conf` (рис. 2), отключила пакетный фильтр командами `iptables -F`, `iptables -P INPUT ACCEPT` и `iptables -P OUTPUT ACCEPT`. (рис. 3)



```
ivkhanina@ivkhanina:/home/ivkhanina
Файл  Правка  Вид  Поиск  Терминал  Справка
Downloading packages:
(1/3): mailcap-2.1.41-2.el7.noarch.rpm | 31 kB  00:00:01
(2/3): httpd-tools-2.4.6-97.el7.centos.2.x86_64.rpm | 94 kB  00:00:01
(3/3): httpd-2.4.6-97.el7.centos.2.x86_64.rpm | 2.7 MB  00:00:01
-----
Общий размер | 1.9 MB/s | 2.8 MB  00:01
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Установка : httpd-tools-2.4.6-97.el7.centos.2.x86_64 1/3
  Установка : mailcap-2.1.41-2.el7.noarch 2/3
  Установка : httpd-2.4.6-97.el7.centos.2.x86_64 3/3
  Проверка : httpd-2.4.6-97.el7.centos.2.x86_64 1/3
  Проверка : mailcap-2.1.41-2.el7.noarch 2/3
  Проверка : httpd-tools-2.4.6-97.el7.centos.2.x86_64 3/3
Установлено:
httpd.x86_64 0:2.4.6-97.el7.centos.2
Установлены зависимости:
httpd-tools.x86_64 0:2.4.6-97.el7.centos.2 mailcap.noarch 0:2.1.41-2.el7
Выполнено!
```

Figure 4.1: Рис 1. Установка веб-сервера



```
mc [root@ivkhanina.localdomain]:/etc/httpd/conf
Файл  Правка  Вид  Поиск  Терминал  Справка
httpd.conf  [-M--] 18 L: [ 75+21  96/354] *(3596/11771b) 0010 0x00A  [*][X]
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#
#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed.  This address appears on some server-generated pages, such
# as error documents.  e.g. admin@your-domain.com
#
ServerAdmin root@localhost
#
# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
#ServerName www.example.com:80
ServerName test.ru
#
1Помощь 2Сох-ть 3Залок 4Замена 5Копия 6Пер-ть 7Поиск 8Удалить 9МенюМС 10Выход
```

Figure 4.2: Рис 2. Изменение конфигурационного файла /etc/httpd/httpd.conf

```
[root@ivkhanina ~]# iptables -F
[root@ivkhanina ~]# iptables -P INPUT ACCEPT
[root@ivkhanina ~]# iptables -P OUTPUT ACCEPT
[root@ivkhanina ~]# █
```

Figure 4.3: Рис 3. Отключение пакетного фильтра

2. Я вошла в систему и убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`. (рис. 4). Далее с помощью команды `service httpd status` я обратилась с помощью браузера к веб-серверу, запущенному на компьютере, и убедилась, что он работает. (рис. 5)

```
[root@ivkhanina ~]# getenforce
Enforcing
[root@ivkhanina ~]# 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
Max kernel policy version:     31
[root@ivkhanina ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ivkhanina ~]#
```

Figure 4.4: Рис 4. SELinux работает в режиме enforcing политики targeted

```
[root@ivkhanina ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ivkhanina ~]# service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Чт 2021-11-25 22:48:58 MSK; 20s ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Main PID: 2979 (httpd)
    Status: "Total requests: 0; Current requests/sec: 0; Current traffic:  0 B/sec"
     Tasks: 6
    CGroup: /system.slice/httpd.service
            └─2979 /usr/sbin/httpd -DFOREGROUND
              └─2983 /usr/sbin/httpd -DFOREGROUND
                └─2984 /usr/sbin/httpd -DFOREGROUND
                  └─2985 /usr/sbin/httpd -DFOREGROUND
                    └─2986 /usr/sbin/httpd -DFOREGROUND
                      └─2987 /usr/sbin/httpd -DFOREGROUND

ноя 25 22:48:58 ivkhanina.localdomain systemd[1]: Starting The Apache HTTP Server...
ноя 25 22:48:58 ivkhanina.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@ivkhanina ~]#
```

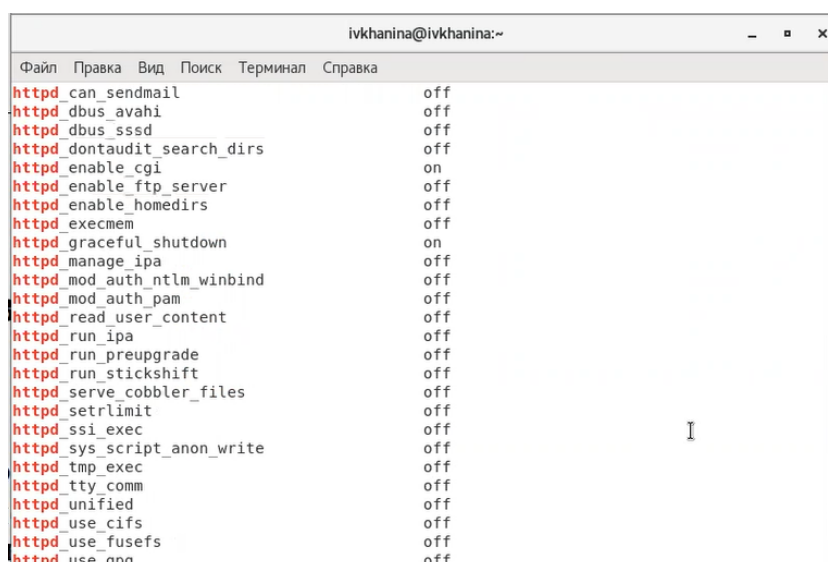
Figure 4.5: Рис 5. Команда service httpd status

- Затем я нашла веб-сервер Apache в списке процессов и определила его контекст безопасности, используя команду `ps auxZ | grep httpd`. (рис. 6)

```
[root@ivkhanina ~]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 2979 0.1 0.4 224084 5020 ? Ss 22:
48 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 2983 0.0 0.3 226168 3096 ? S 22:
48 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 2984 0.0 0.3 226168 3096 ? S 22:
48 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 2985 0.0 0.3 226168 3096 ? S 22:
48 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 2986 0.0 0.3 226168 3096 ? S 22:
48 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 2987 0.0 0.3 226168 3096 ? S 22:
48 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 3025 0.0 0.0 112832 976 pts
/0 R+ 22:49 0:00 grep --color=auto httpd
```

Figure 4.6: Рис 6. Команда ps auxZ | grep httpd

4. Я посмотрела текущее состояние переключателей SELinux для Apache с помощью команды sestatus -bigrep httpd и обратила внимание, что многие из них находятся в положении «off». (рис. 7)



```
ivkhanina@ivkhanina:~
Файл Правка Вид Поиск Терминал Справка
httpd_can_sendmail off
httpd_dbus_avahi off
httpd_dbus_sssd off
httpd_dontaudit_search_dirs off
httpd_enable_cgi on
httpd_enable_ftp_server off
httpd_enable_homedirs off
httpd_execmem off
httpd_graceful_shutdown on
httpd_manage_ipa off
httpd_mod_auth_ntlm_winbind off
httpd_mod_auth_pam off
httpd_read_user_content off
httpd_run_ipa off
httpd_run_preupgrade off
httpd_run_stickshift off
httpd_serve_cobbler_files off
httpd_setrlimit off
httpd_ssi_exec off
httpd_sys_script_anon_write off
httpd_tmp_exec off
httpd_tty_comm off
httpd_unified off
httpd_use_cifs off
httpd_use_fusefs off
httpd_use_gpg off
```

Figure 4.7: Рис 7. Команда sestatus -bigrep httpd

5. Я посмотрела статистику по политике с помощью команды seinfo, также определила множество пользователей, ролей, типов. Затем я определила тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды ls -lZ /var/www. А потом определила тип файлов, находящихся в директории /var/www/html, введя команду ls -lZ /var/www/html. Определила круг пользователей, которым разрешено создание файлов в директории /var/www/html. (рис. 8)

```
[root@ivkhanina ~]# seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version & Type: v.31 (binary, mls)

Classes: 130 Permissions: 272
Sensitivities: 1 Categories: 1024
Types: 4793 Attributes: 253
Users: 8 Roles: 14
Booleans: 316 Cond. Expr.: 362
Allow: 107834 Neverallow: 0
Auditallow: 158 Dontaudit: 10022
Type_trans: 18153 Type_change: 74
Type_member: 35 Role_allow: 37
Role_trans: 414 Range_trans: 5899
Constraints: 143 Validatetrans: 0
Initial SIDs: 27 Fs_use: 32
Genfscon: 103 Portcon: 614
Netifcon: 0 Nodecon: 0
Permissives: 0 Polcap: 5

[root@ivkhanina ~]# ls -lZ /var/www
drwxr-xr-x. root root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin
drwxr-xr-x. root root system_u:object_r:httpd_sys_content_t:s0 html
```

Figure 4.8: Рис 8. Команды seinfo, ls -lZ /var/www и ls -lZ /var/www/html

6. Далее от имени суперпользователя я создала html-файл /var/www/html/test.html. Проверила контекст созданного файла. (рис. 9)

```
[root@ivkhanina html]# vi test.html
[root@ivkhanina html]# ls
test.html
[root@ivkhanina html]# cat /var/www/html/test.html
<html>
<body>test</body>
</html>
[root@ivkhanina html]# ls -lZ /var/www/html
-rw-r--r--. root root unconfined_u:object_r:httpd_sys_content_t:s0 test.html
[root@ivkhanina html]# █
```

Figure 4.9: Рис 9. Создание файла /var/www/html/test.html

7. Обратилась к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Убедилась, что файл был успешно отображён. (рис. 10)

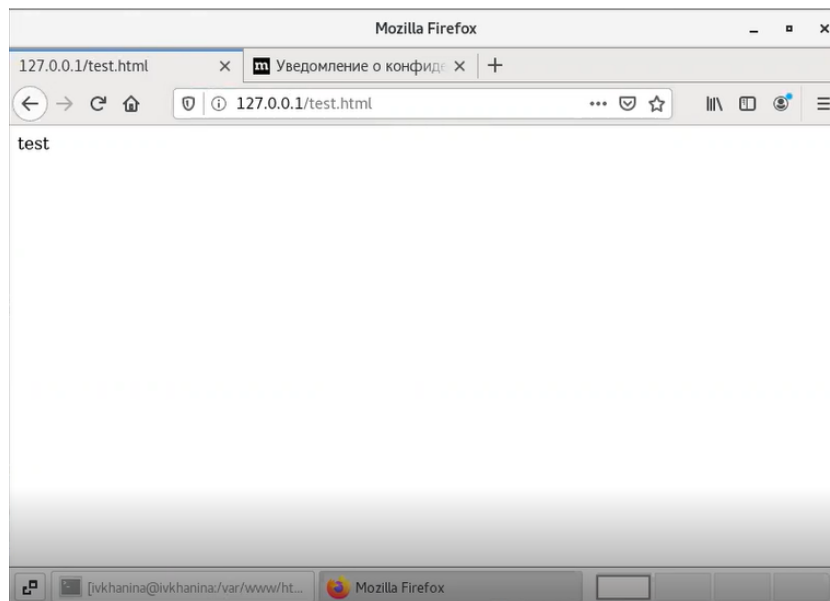


Figure 4.10: Рис 10. Обращение к файлу через сервер

8. Я изучила справку `man httpd_selinux` и выяснила, какие контексты файлов определены для `httpd`. Затем я сопоставила их с типом файла `test.html`. Проверила контекст файла с помощью команды `ls -Z /var/www/html/test.html`. Я изменила контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на `samba_share_t` командами `chcon -t samba_share_t /var/www/html/test.html` и `ls -Z /var/www/html/test.html`. Проверила, контекст поменялся. (рис. 11)

```
[root@ivkhanina html]# man httpd_selinux
Нет справочной страницы для httpd_selinux
[root@ivkhanina html]# ls -Z /var/html/test.html
ls: невозможно получить доступ к /var/html/test.html: Нет такого файла или каталога
[root@ivkhanina html]# ls -Z /var/www/html/test.html
-rw-r--r--. root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.h
tml
[root@ivkhanina html]# chcon -t samba_share_t /var/www/html/test.html
[root@ivkhanina html]# ls -Z /var/www/html/test.html
-rw-r--r--. root root unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Figure 4.11: Рис 11. Изменение контекста файла `/var/www/html/test.html`

9. Попробовала ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html` и получила сообщение об ошибке. Из-за смены контекста доступ к файлу запрещен. (рис. 12)

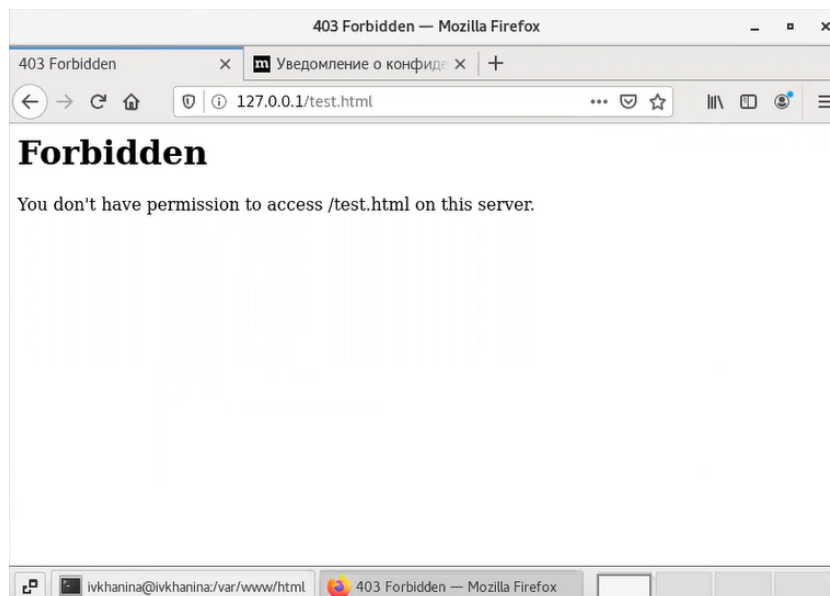


Figure 4.12: Рис 12. Доступ к файлу запрещен

10. Я просмотрела log-файлы веб-сервера Apache, а также системный лог-файл: `tail /var/log/messages`. (рис. 13)

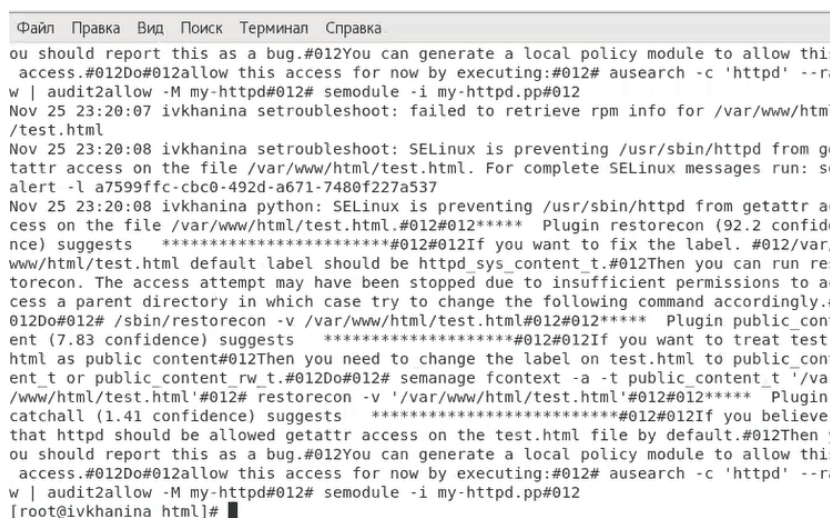


Figure 4.13: Рис 13. Команда `tail /var/log/messages`

11. Попробовала запустить веб-сервер Apache на прослушивание TCP-порта 81. Для этого в файле `/etc/httpd/httpd.conf` нашла строчку `Listen 80` и заменила её на `Listen 81`. (рис. 14)

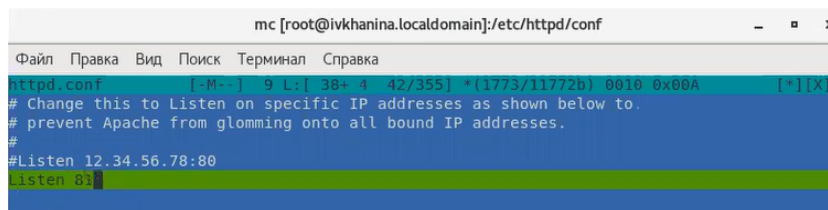


Figure 4.14: Рис 14. Изменение Listen 80 на Listen 81

12. Перезапустила веб-сервер Apache. Проанализировала лог-файлы: `tail -nl /var/log/messages`. Просмотрела файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и выяснила, в каких файлах появились записи. (рис. 15)

```

NCH, shutting down gracefully
[Thu Nov 25 23:30:17.957764 2021] [core:notice] [pid 4579] SELinux policy enabled; http
d running as context system_u:system_r:httpd_t:s0
[Thu Nov 25 23:30:17.960329 2021] [suexec:notice] [pid 4579] AH01232: suEXEC mechanism
enabled (wrapper: /usr/sbin/suexec)
[Thu Nov 25 23:30:17.977191 2021] [lbmethod_heartbeat:notice] [pid 4579] AH02282: No sl
otmem from mod_heartbeat
[Thu Nov 25 23:30:17.981004 2021] [mpm_prefork:notice] [pid 4579] AH00163: Apache/2.4.6
(CentOS) configured -- resuming normal operations
[Thu Nov 25 23:30:17.981044 2021] [core:notice] [pid 4579] AH00094: Command line: '/usr
/sbin/httpd -D FOREGROUND'
[root@ivkhanina ~]# cat /var/log/httpd/access_log
127.0.0.1 - - [25/Nov/2021:23:09:27 +0300] "GET /test.html HTTP/1.1" 200 33 "-" "Mozill
a/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
127.0.0.1 - - [25/Nov/2021:23:09:28 +0300] "GET /favicon.ico HTTP/1.1" 404 209 "-" "Moz
illa/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
127.0.0.1 - - [25/Nov/2021:23:13:00 +0300] "GET /test.html HTTP/1.1" 200 33 "-" "Mozill
a/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
127.0.0.1 - - [25/Nov/2021:23:13:00 +0300] "GET /favicon.ico HTTP/1.1" 404 209 "-" "Moz
illa/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
127.0.0.1 - - [25/Nov/2021:23:20:00 +0300] "GET /test.html HTTP/1.1" 403 211 "-" "Mozil
la/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
127.0.0.1 - - [25/Nov/2021:23:20:00 +0300] "GET /favicon.ico HTTP/1.1" 404 209 "-" "Moz
illa/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
[root@ivkhanina ~]# cat /var/log/audit/audit.log

```

Figure 4.15: Рис 15. Просмотр log файлов

13. Выполнила команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверила список портов командой `semanage port -l | grep http_port_t`. Убедилась, что порт 81 появился в списке. Запустила веб-сервер Apache ещё раз. (рис. 16)

```

[root@ivkhanina ~]# semanage port -a -t http_port_t -p tcp 81
ValueError: Порт tcp/81 уже определен
[root@ivkhanina ~]# semanage port -l | grep http_port_t
http_port_t      tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t tcp      5988
[root@ivkhanina ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ivkhanina ~]#

```

Figure 4.16: Рис 16. Команда `semanage port -a -t http_port_t -p tcp 81`

14. Вернула контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html` командами `chcon -t httpd_sys_content_t /var/www/html/test.html`. (рис. 17)

```
[root@ivkhanina ~]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@ivkhanina ~]#
```

Figure 4.17: Рис 17. Изменение контекста файла `/var/www/html/test.html`

15. После этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`. Я увидела содержимое файла — слово «test». (рис. 18)

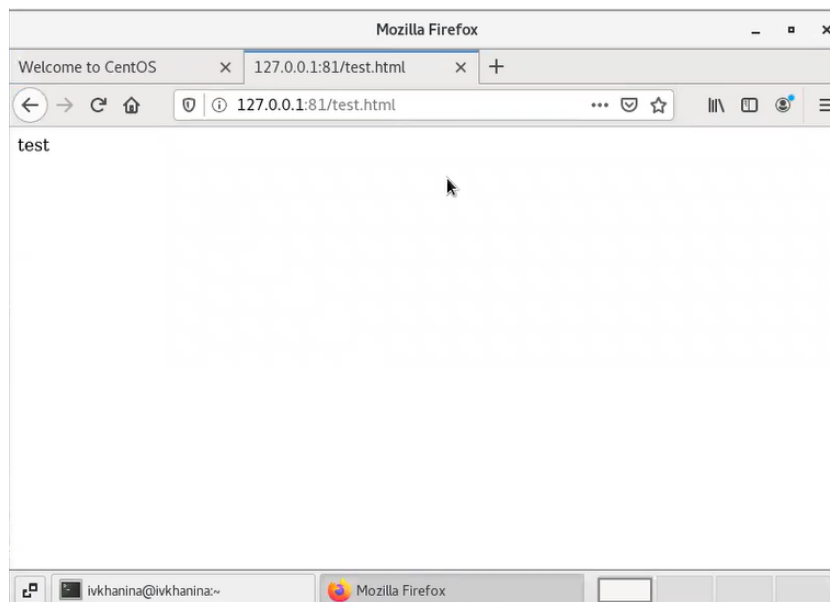


Figure 4.18: Рис 18. Обращение к файлу через сервер

16. Исправила обратно конфигурационный файл `apache`, вернув `Listen 80`. (рис. 19)

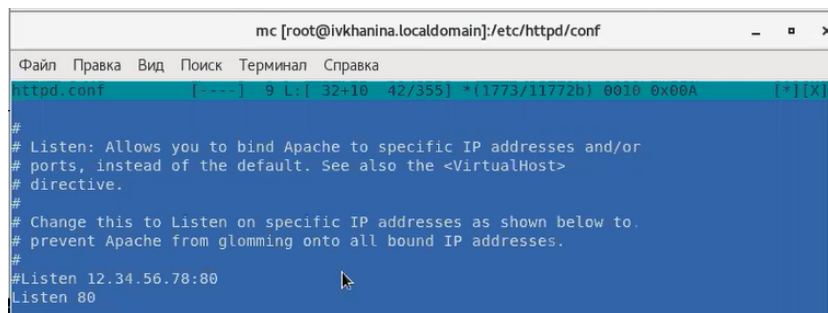


Figure 4.19: Рис 19. Изменение конфигурационного файла /etc/httpd/httpd.conf

24. Удалила привязку `http_port_t` к 81 порту с помощью команды `semanage port -d -t http_port_t -p tcp 81` и проверила, что порт 81 удалён. Удалила файл `/var/www/html/test.html` командой `rm /var/www/html/test.html`. (рис. 20)

```
[root@ivkhanina ~]# semanage port -d -t http_port_t -p tcp 81
ValueError: Порт tcp/81 определен на уровне политики и не может быть удален
[root@ivkhanina ~]# semanage port -l | grep http_port_t
http_port_t          tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t  tcp      5988
[root@ivkhanina ~]# rm /var/www/html/test.html
rm: удалить обычный файл «/var/www/html/test.html»? y
[root@ivkhanina ~]#
```

Figure 4.20: Рис 20. Удаление файла /var/www/html/test.html

5 Выводы

В результате выполнения лабораторной работы я развила навыки администрирования ОС Linux, получила первое практическое знакомство с технологией SELinux, а также проверить работу SELinux на практике совместно с веб-сервером Apache.

Список литературы

1. SELinux
2. SELinux – описание и особенности работы с системой
3. ВВЕДЕНИЕ В SELINUX В CENTOS 7: БАЗОВЫЕ ПОНЯТИЯ