Лабораторная работа №6.

Мандатное разграничение прав в Linux

Стариков Данила Андреевич

Содержание

1 Цель работы

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

2 Выполнение лабораторной работы

1. Вошли в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (Рис. 2.1).

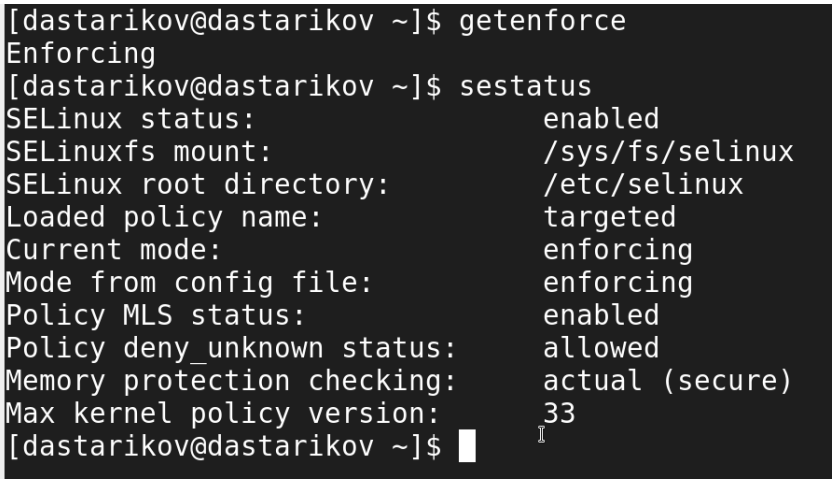

Рис. 2.1: Проверка режима работы SELinux.

2. Обратились с помощью браузера к веб-серверу, запущенному на компьютере, и убедились, что последний работает (Рис. 2.2):

```
service httpd status
```

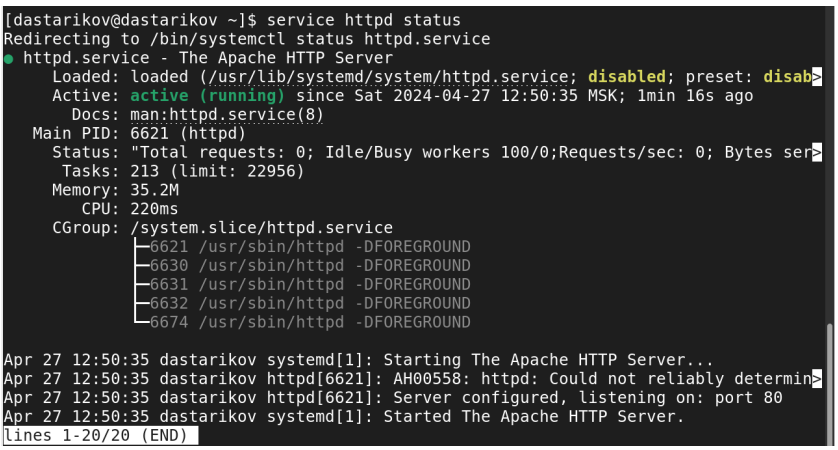

Рис. 2.2: Проверка режима работы httpd.

3. Нашли веб-сервер Apache в списке процессов, определили его контекст безопасности (Рис. 2.3)

```
ps auxZ | grep httpd
```

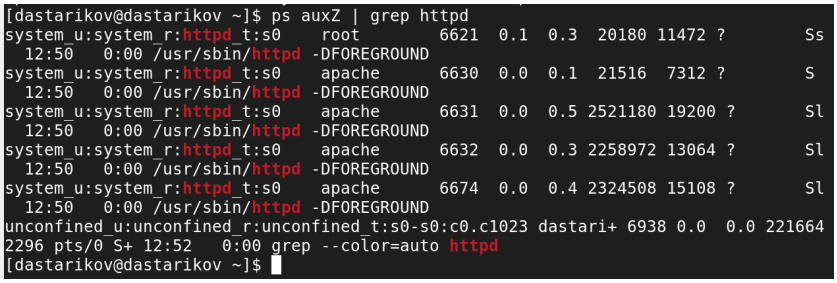

Рис. 2.3: Список всех связанных с httpd процессов.

4. Посмотрели текущее состояние переключателей SELinux для Apache с помощью команды (Рис. 2.4)

```
sestatus -b httpd
```

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

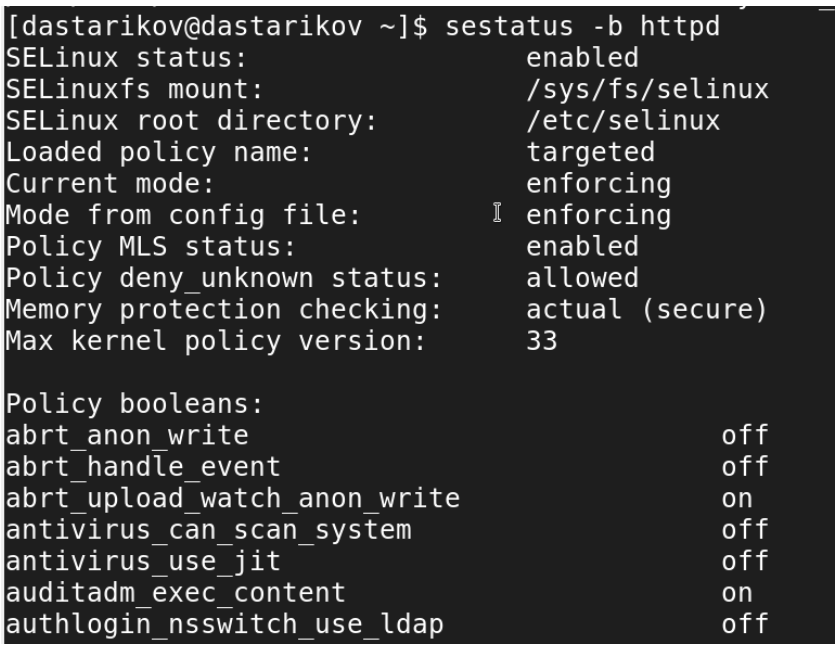

Рис. 2.4: Фрагмент справки текущих состояний httpd.

5. Посмотрели статистику по политике с помощью команды `seinfo`, также определили множество пользователей, ролей, типов (Рис. 2.5). Пользователей – 8, ролей – 15, типов – 5135.

```
[dastarikov@dastarikov ~]$ 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:              5135    Attributes:          259
  Users:                 8    Roles:                15
  Booleans:            357    Cond. Expr.:         390
  Allow:             65409    Neverallow:            0
  Auditallow:          172    Dontaudit:          8647
  Type_trans:       267813    Type_change:          94
  Type_member:          37    Range_trans:        6164
  Role allow:           39    Role_trans:          419
  Constraints:          70    Validatetrans:         0
  MLS Constrain:        72    MLS Val. Tran:         0
  Permissives:           2    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
[dastarikov@dastarikov ~]$
```

Рис. 2.5: Статистика по политике httpd.

6. Определили тип файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды (Рис. 2.6)

```
ls -lZ /var/www
```

```
[dastarikov@dastarikov ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Oct 28 12:35
cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0     6 Oct 28 12:35
html
[dastarikov@dastarikov ~]$
```

Рис. 2.6: Информация о содержимом каталога /var/www .

7. Определили тип файлов, находящихся в директории `/var/www/html` (Рис. 2.7):

```
ls -lZ /var/www/html
```

```
[dastarikov@dastarikov ~]$ ls -lZ /var/www/html
total 0
```

Рис. 2.7: Информация о содержимом каталога /var/www/html .

8. Определили круг пользователей, которым разрешено создание файлов в директории `/var/www/html`. Только пользователи `root` имеют право создавать файлы в директории `/var/www/html`.

9. Создали от имени суперпользователя html-файл `/var/www/html/test.html` следующего содержания (Рис. 2.8):

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

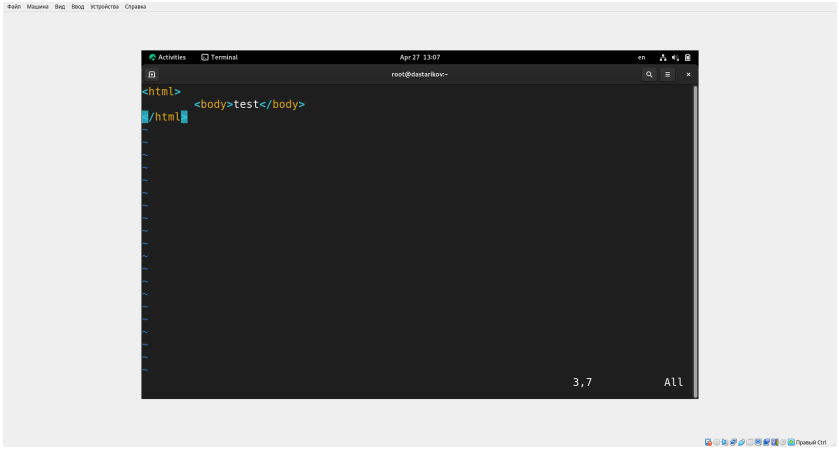

Рис. 2.8: Создание файла test.html.

10. Проверили контекст созданного вами файла. По умолчанию файлам присваивается контекст `unconfined_u:object_u:httpd_sys_content_t:s0` (Рис. 2.9).

```
[root@dastarikov ~]# ls -lZ /var/www/html
total 4
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 34 Apr 27 13:08
 test.html
```

Рис. 2.9: Проверка контекста test.html.

11. Обратились к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.htm` Убедились, что файл был успешно отображён.

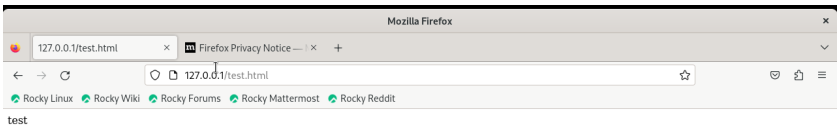

Рис. 2.10: Открытие файла через веб-браузер.

12. Изучили справку man httpd_selinux и выяснили, какие контексты файлов определены для `httpd`. Сопоставили их с типом файла `test.html` (Рис. 2.9).

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

13. Изменили контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на `samba_share_t`(Рис. 2.11):

```
chcon -t samba_share_t /var/www/html/test.html
ls -Z /var/www/html/test.html
```

Рис. 2.11: Изменение контекста файла test.html.

14. Попробовали ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Получили сообщение об ошибке (Рис. 2.12):

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

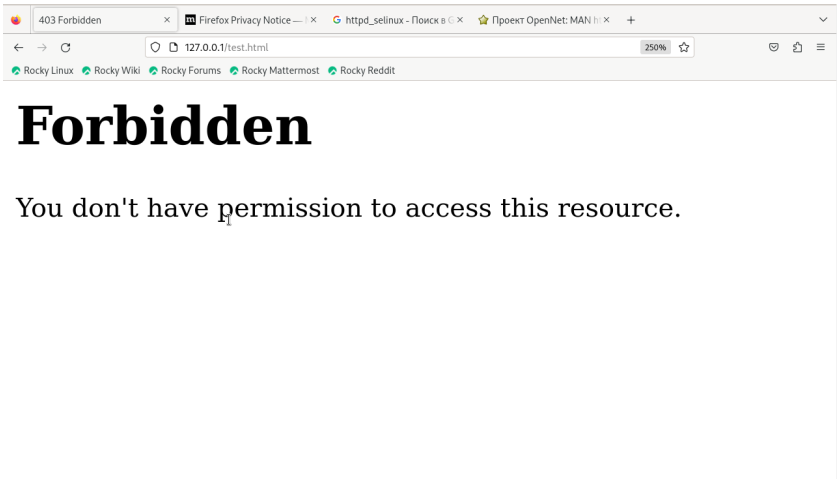

Рис. 2.12: Попытка открытия файла через веб-браузер.

15. Проанализировали 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 Рис. 2.13):

```
tail /var/log/messages
```

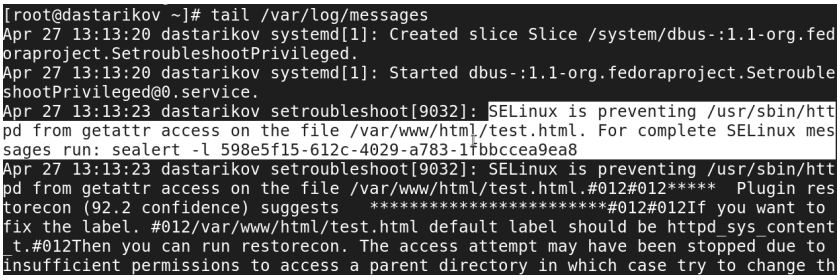

Рис. 2.13: Просмотр логов веб-сервера.

SELinux отказывает в доступе к файлу, так как нет соответствующего контекста безопасности.

16. Попробовали запустить веб-сервер Apache на прослушивание TCP-порта 81. Для этого в файле `/etc/httpd/httpd.conf` найдите строчку `Listen 80` и замените её на `Listen 81`.

17. Выполнили перезапуск веб-сервера Apache. Произошёл сбой (Рис. 2.14).

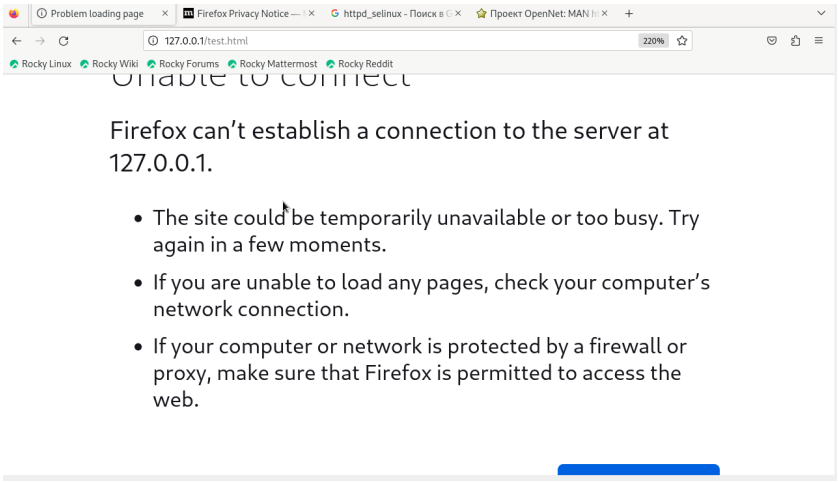

Рис. 2.14: Ошибка при попытке открытия файла через веб-браузер.

18. Проанализировали лог-файлы (Рис. 2.15 - 2.17):

```
tail -nl /var/log/messages
```

Просмотрели файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и выяснили, в каких файлах появились записи.

```
[root@dastarikov ~]# tail /var/log/messages
Apr 27 13:16:48 dastarikov httpd[9256]: AH00558: httpd: Could not reliably determine
 the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' dir
ective globally to suppress this message
Apr 27 13:16:48 dastarikov httpd[9256]: Server configured, listening on: port 81
Apr 27 13:16:48 dastarikov systemd[1]: Started The Apache HTTP Server.
Apr 27 13:17:30 dastarikov systemd[1]: Stopping The Apache HTTP Server...
Apr 27 13:17:31 dastarikov systemd[1]: httpd.service: Deactivated successfully.
Apr 27 13:17:31 dastarikov systemd[1]: Stopped The Apache HTTP Server.
Apr 27 13:17:33 dastarikov systemd[1]: Starting The Apache HTTP Server...
Apr 27 13:17:33 dastarikov httpd[9496]: AH00558: httpd: Could not reliably determine
 the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' dir
ective globally to suppress this message
Apr 27 13:17:33 dastarikov httpd[9496]: Server configured, listening on: port 81
Apr 27 13:17:33 dastarikov systemd[1]: Started The Apache HTTP Server.
```

Рис. 2.15: Просмотр /var/log/messages.

```
[root@dastarikov ~]# tail /var/log/httpd/error_log
[Sat Apr 27 13:16:48.502051 2024] [lbmethod_heartbeat:notice] [pid 9256:tid 9256] AH
02282: No slotmem from mod_heartmonitor
[Sat Apr 27 13:16:48.513891 2024] [mpm_event:notice] [pid 9256:tid 9256] AH00489: Ap
ache/2.4.57 (Rocky Linux) configured -- resuming normal operations
[Sat Apr 27 13:16:48.513956 2024] [core:notice] [pid 9256:tid 9256] AH00094: Command
 line: '/usr/sbin/httpd -D FOREGROUND'
[Sat Apr 27 13:17:30.117846 2024] [mpm_event:notice] [pid 9256:tid 9256] AH00492: ca
ught SIGWINCH, shutting down gracefully
[Sat Apr 27 13:17:33.536517 2024] [core:notice] [pid 9496:tid 9496] SELinux policy e
nabled; httpd running as context system_u:system_r:httpd_t:s0
[Sat Apr 27 13:17:33.539485 2024] [suexec:notice] [pid 9496:tid 9496] AH01232: suEXE
C mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain nam
e, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
[Sat Apr 27 13:17:33.589789 2024] [lbmethod_heartbeat:notice] [pid 9496:tid 9496] AH
02282: No slotmem from mod_heartmonitor
[Sat Apr 27 13:17:33.602049 2024] [mpm_event:notice] [pid 9496:tid 9496] AH00489: Ap
ache/2.4.57 (Rocky Linux) configured -- resuming normal operations
[Sat Apr 27 13:17:33.602138 2024] [core:notice] [pid 9496:tid 9496] AH00094: Command
 line: '/usr/sbin/httpd -D FOREGROUND'
```

Рис. 2.16: Просмотр /var/log/httpd/error_log.

```
[root@dastarikov ~]# tail /var/log/audit/audit.log
type=SERVICE_STOP msg=audit(1714212800.340:509): pid=1 uid=0 auid=4294967295 ses=429
4967295 subj=system_u:system_r:init_t:s0 msg='unit=systemd-hostnamed comm="systemd"
exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'UID="root" A
UID="unset"
type=BPF msg=audit(1714212800.391:510): prog-id=123 op=UNLOAD
type=BPF msg=audit(1714212800.391:511): prog-id=122 op=UNLOAD
type=SERVICE_START msg=audit(1714212800.642:512): pid=1 uid=0 auid=4294967295 ses=42
94967295 subj=system_u:system_r:init_t:s0 msg='unit=dbus-:1.1-org.fedoraproject.Setr
oubleshootPrivileged@0 comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr
=? terminal=? res=success'UID="root" AUID="unset"
type=SERVICE_STOP msg=audit(1714212813.386:513): pid=1 uid=0 auid=4294967295 ses=429
4967295 subj=system_u:system_r:init_t:s0 msg='unit=dbus-:1.1-org.fedoraproject.Setro
ubleshootPrivileged@0 comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=
? terminal=? res=success'UID="root" AUID="unset"
type=SERVICE_STOP msg=audit(1714212813.463:514): pid=1 uid=0 auid=4294967295 ses=429
4967295 subj=system_u:system_r:init_t:s0 msg='unit=setroubleshootd comm="systemd" ex
e="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'UID="root" AUI
D="unset"
type=SERVICE_STOP msg=audit(1714213008.324:515): pid=1 uid=0 auid=4294967295 ses=429
4967295 subj=system_u:system_r:init_t:s0 msg='unit=httpd comm="systemd" exe="/usr/li
b/systemd/systemd" hostname=? addr=? terminal=? res=success'UID="root" AUID="unset"
type=SERVICE_START msg=audit(1714213008.505:516): pid=1 uid=0 auid=4294967295 ses=42
94967295 subj=system_u:system_r:init_t:s0 msg='unit=httpd comm="systemd" exe="/usr/l
```

Рис. 2.17: Просмотр /var/log/audit/audit.log.

19. Выполнили команду (Рис. 2.18)

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

После этого проверили список портов командой

```
semanage port -l | grep http_port_t
```

Убедились, что порт 81 появился в списке.

```
[root@dastarikov ~]# 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
```

Рис. 2.18: Настройка прослушивания порта 81.

20. Вернули контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html` (Рис. 2.19):

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

Рис. 2.19: Возвращение нужного контекста файлу 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» (Рис. 2.20).

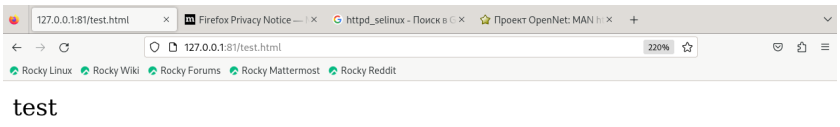

Рис. 2.20: Открытие файла через веб-браузер на порте 81.

21. Исправили обратно конфигурационный файл apache, вернув `Listen 80`.

22. Удалили файл `/var/www/html/test.html`:

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

Более подробно о SELinux можно прочитать в книге [1].

3 Выводы

В рамках лабораторной работы развили навыки администрирования ОС Linux, получили первое практическое знакомство с технологией SELinux и проверили работу SELinux на практике совместно с веб-сервером Apache.

Список литературы

1. Vermeulen S. SELinux System Administration. Second Edition. Packt Publishing, 2016. 300 с.