Отчёт по лабораторной работе

Мандатное разграничение прав в Linux

Назарьева Алена Игоревна НФИбд-03-18

Содержание

List of Figures

List of Tables

1 Цель работы

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

2 Выполнение лабораторной работы

1. Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus.
2. Обратилась с помощью браузера к веб-серверу, запущенному на моем компьютере, и убедилась, что последний работает: service httpd status или /etc/rc.d/init.d/httpd status (рис. -fig. 2.1)

Figure 2.1: пункты 1-2

3. Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности. Как мы видим, этот процесс запущен на домене httpd_t. Например, можно использовать команду ps auxZ | grep httpd или ps -eZ | grep httpd
4. Посмотрела текущее состояние переключателей SELinux для Apache с помощью команды sestatus -bigrep httpd (рис. -fig. 2.2)

```
[ainazarieva@ainazarieva ~]$ ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      3403 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      3494 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      3495 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      3496 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      3497 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      3498 ?        00:00:00 httpd
[ainazarieva@ainazarieva ~]$ sestatus -bigrep httpd
```

Figure 2.2: пункты 3-4

5. Посмотрела статистику по политике с помощью команды seinfo, также определила множество пользователей:8, ролей:14, типов:4793. (рис. -fig. 2.3)

```
[root@ainazarieva ainazarieva]# 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
```

Figure 2.3: пункт 5

6. Определила тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды: ls -lZ /var/www: httpd_sys_script_exec_t httpd_sys_content_t
7. Определила тип файлов, находящихся в директории /var/www/html: ls -lZ /var/www/html (рис. -fig. 2.4)

```
[root@ainazarieva ainazarieva]# 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
[root@ainazarieva ainazarieva]# ls -lZ /var/www/html
```

Figure 2.4: пункты 6-7

8. Определила круг пользователей, которым разрешено создание файлов в директории /var/www/html. rwxr-xr-x
9. Создала 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
10. Проверила контекст созданного файла. Контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html: unconfined_u:object_r:httpd_sys_content_t:s0 (рис. -fig. **??**)

11. Обратилась к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Убедилась, что файл был успешно отображён (рис. -fig. 2.5)

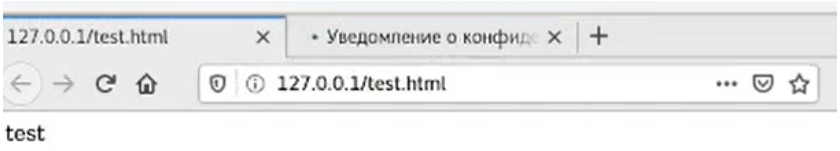

Figure 2.5: пункты 11

12. Изучила справку man httpd_selinux и выяснила, какие контексты файлов определены для httpd:httpd_sys_content_t;httpd_sys_script_exec_t;httpd_sys_script_ro_t; httpd_sys_script_rw_t . Они совпадают с типом

файла test.html. Проверила контекст файла можно командой ls -Z. ls -Z /var/www/html/test.html

13. Изменила 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 проверила, что контекст поменялся. (рис. -fig. 2.6)

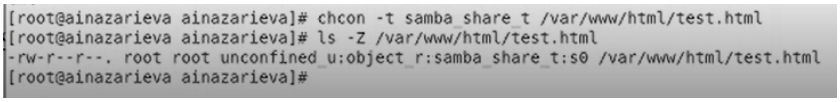

Figure 2.6: пункты 12-13

14. Попробовала ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Получила сообщение об ошибке: Forbidden You don't have permission to access /test.html on this server. (рис. -fig. 2.7)

Figure 2.7: пункт 14

15. Проанализировала ситуацию. Доступ разрешен только между элементами с одинаковым типом, именно поэтому веб-сервер Apache может без проблем читать файл /var/www/html/test.html, который имеет тип httpd_sys_content_t. В то же самое время, так как Apache запущен на домене httpd_t и не имеет заполненных полей userid:username, он не может получить

доступ к файлу home/username/test.html с другим типом, хотя этот файл доступен для чтения процессам, для которых не определена целевая политика ls -l /var/www/html/test.html Просмотрела log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: tail /var/log/messages В системе оказались запущенные процессы setroubleshootd и audtd, мы также смогли увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log. (рис. -fig. 2.8)

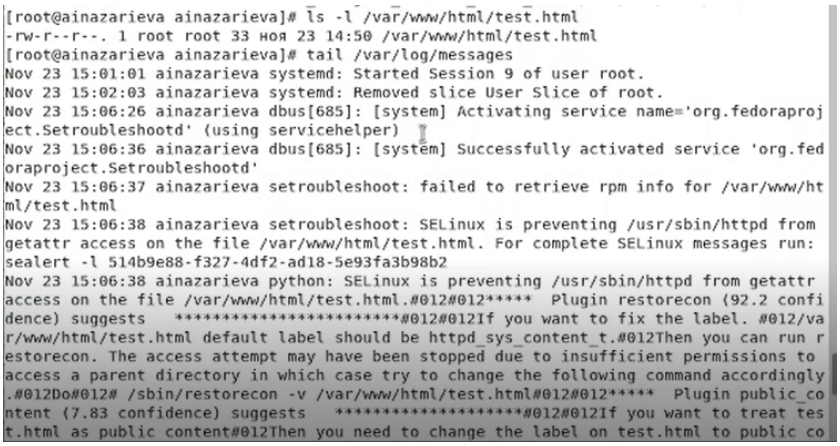

Figure 2.8: пункт 15

16. Попробовала 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 нашла строчку Listen 80 и заменила её на Listen 81. (рис. -fig. 2.9)

Figure 2.9: пункт 16

17. Выполнила перезапуск веб-сервера Apache. Сбой не произошел (рис. -fig. 2.10)

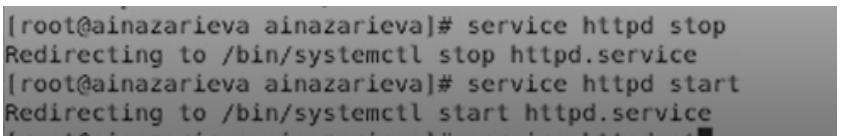

Figure 2.10: пункт 17

18. Проанализировала лог-файлы: tail -nl /var/log/messages Просмотрите файлы /var/log/http/error_log, /var/log/http/access_log и /var/log/audit/audit.log и выяснила, что записи появились только в /var/log/audit/audit.log (рис. -fig. 2.11)

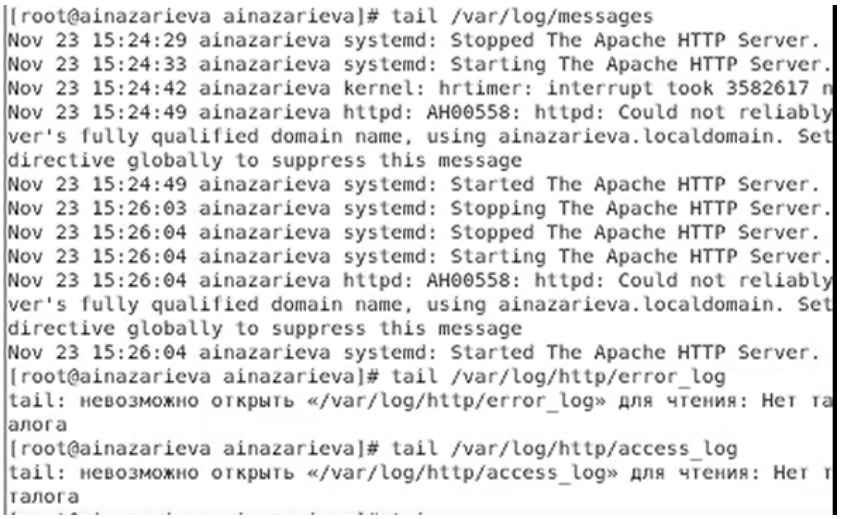

Figure 2.11: пункт 18-1

(рис. -fig. 2.12)

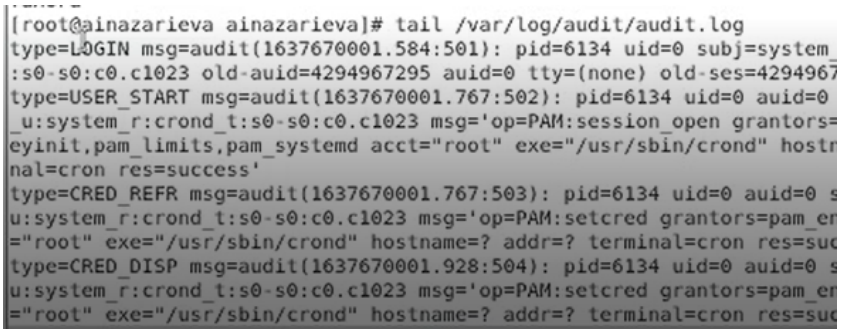

Figure 2.12: пункт 18-2

19. 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.

20. Попробовала запустить веб-сервер Apache ещё раз.

21. Вернула контекст httpd_sys_content__t к файлу /var/www/html/ test.html: chcon -t httpd_sys_content_t /var/www/html/test.html (рис. -fig. 2.13)

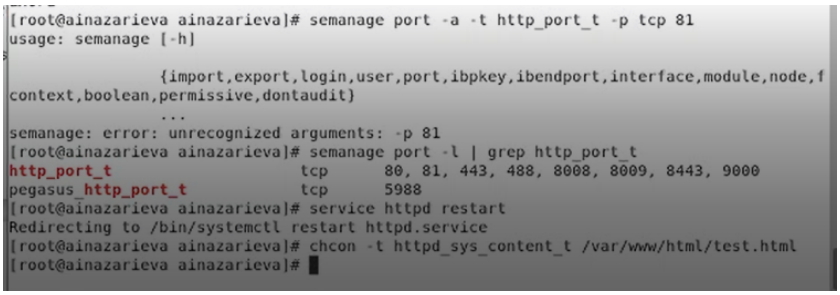

Figure 2.13: пункт 19-21-1

После этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. Увидела содержимое файла — слово «test». (рис. -fig. 2.14)

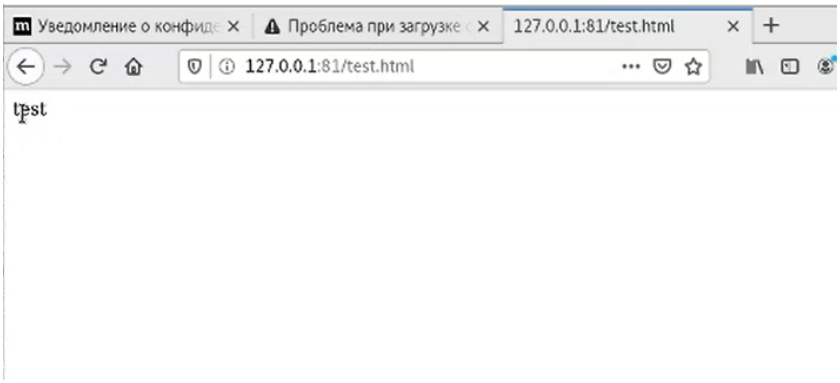

Figure 2.14: пункт 19-21-2

22. Исправила обратно конфигурационный файл apache, вернув Listen 80.

23. Попробовала удалить привязку http_port_t к 81 порту: semanage port -d -t http_port_t -p tcp 81 порт 81 нельзя удалить, т.к. он определен на уровне политики
24. Удалила файл /var/www/html/test.html: rm /var/www/html/test.html (рис. - fig. 2.15)

Figure 2.15: пункты 22-24

3 Выводы

В результате выполнения работы я Развила навыки администрирования ОС Linux, Получида первое практическое знакомство с технологией SELinux1, Проверила работу SELinx на практике совместно с веб-сервером Apache.