Отчёт по лабораторной работе №6

Мандатное разграничение прав в Linux

Голощапова Ирина Борисовна

Содержание

Список иллюстраций

Список таблиц

1 Цели и задачи лабораторной работы

1.1 Цели и задачи 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

2.1 Подготовка к выполнению работы

1. Установила от имени администратора httpd (рис. 2.1)

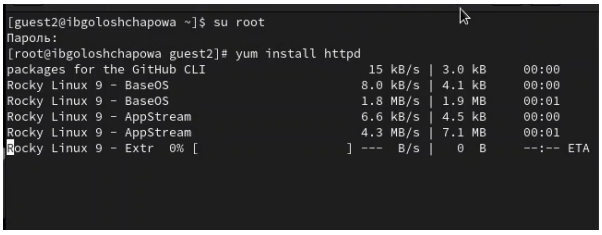

Рис. 2.1: Установка httpd

2. В конфигурационном файле /etc/httpd/httpd.conf задала параметр ServerName (рис. 2.2):

Рис. 2.2: параметр ServerName

3. Также проследила, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp (рис. 2.3):

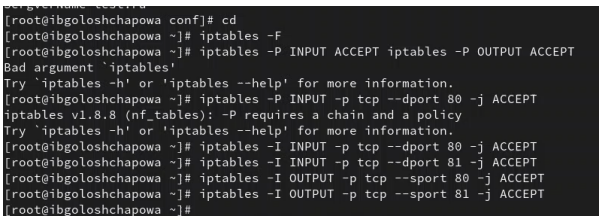

Рис. 2.3: Отключение пакетного фильтра

2.2 Выполнение основной части работы

1. Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted (рис. 2.4):

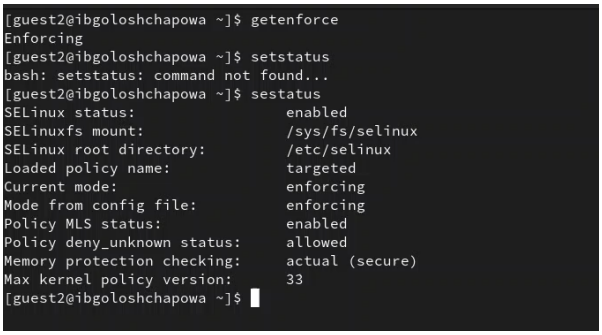

Рис. 2.4: Вход в систему

2. Обратилась с помощью браузера к веб-серверу, запущенному на компьютере, и убедилась, что последний работает (рис. 2.5):

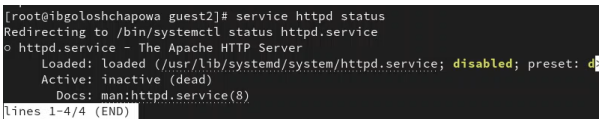

Рис. 2.5: Обращение к веб-браузеру

3. Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности (рис. 2.6):

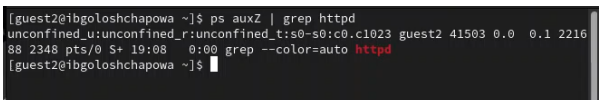

Рис. 2.6: Apach

4. Посмотрела текущее состояние переключателей SELinux для Apache (рис. 2.7):

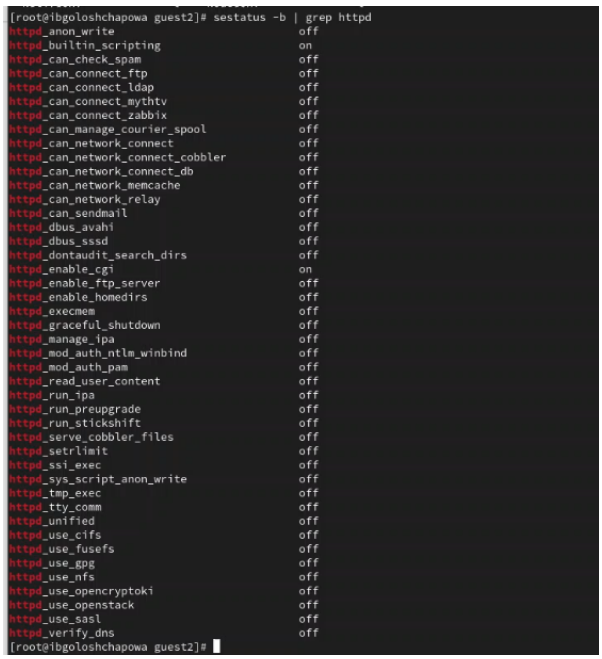

Рис. 2.7: переключатели SELinux

5. Посмотрела статистику по политике с помощью команды seinfo, также определила множество пользователей, ролей, типов (рис. 2.8):

```
httpd_verify_dns                            off
[root@ibgoloshchapowa guest2]# 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:              5100    Attributes:          258
  Users:                 8    Roles:                14
  Booleans:            353    Cond. Expr.:         384
  Allow:             65000    Neverallow:            0
  Auditallow:          170    Dontaudit:          8572
  Type_trans:       265341    Type_change:          87
  Type_member:          35    Range_trans:        6164
  Role allow:           38    Role_trans:          420
  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:             660
  Netifcon:              0    Nodecon:               0
[root@ibgoloshchapowa guest2]#
```

Рис. 2.8: seinfo

6. Определила тип файлов и поддиректорий, находящихся в директории /var/www (рис. 2.9):

```
[root@ibgoloshchapowa guest2]# ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 мая 16 23:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0     6 мая 16 23:21 html
[root@ibgoloshchapowa guest2]#
```

Рис. 2.9: тип файлов и поддиректорий

7. 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 (рис. 2.10):

```
Открыть                    test.html
                           /var/www/html
1 <html>
2 <body>test</body>
3 </html>
```

Рис. 2.10: html-файл

8. Обратилаась к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html (рис. 2.11):

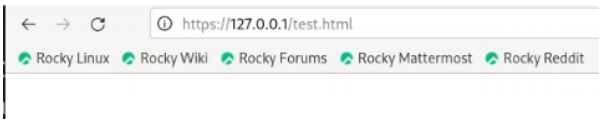

Рис. 2.11: Обращение к файлу через веб-сервер

9. Изучила справку man httpd_selinux и выяснила, какие контексты файлов определены для httpd. Сопоставила их с типом файла test.html. Проверила контекст файла командой ls -Z (рис. 2.12):

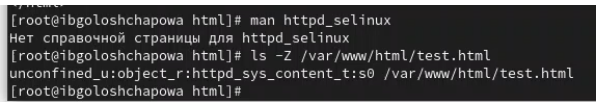

Рис. 2.12: контексты файлов для httpd

10. Изменила контекст файла /var/www/html/test.html с httpd_sys_content_t на другой, к которому процесс httpd не должен иметь доступа, например, на samba_share_t (рис. 2.13):

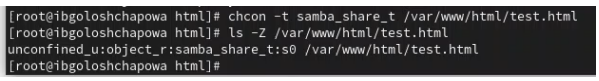

Рис. 2.13: Изменение контекст файла

11. Проанализировала ситуацию. Просмотрела log-файлы веб-сервера Apache. Также просмотрела системный лог-файл (рис. 2.14) (рис. 2.15):

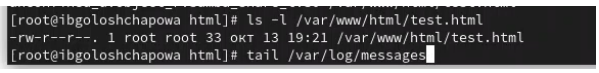

Рис. 2.14: log-файлы веб-сервера Apache

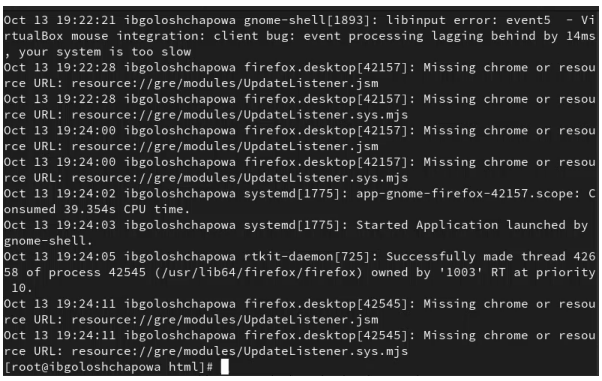

Рис. 2.15: log-файлы веб-сервера Apache_2

12. 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 найшла строчку Listen 80 и заменила её на Listen 81 (рис. 2.16):

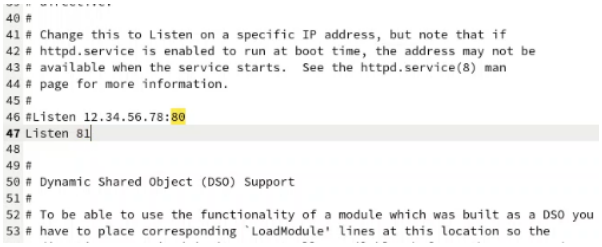

Рис. 2.16: Замена на Listen 81

13. Выполнила команду и после этого проверила список портов(рис. 2.17):

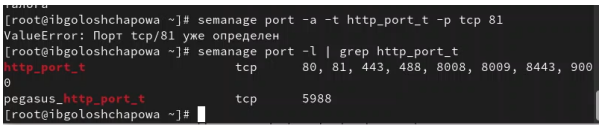

Рис. 2.17: Список портов

14. Вернула контекст httpd_sys_content__t к файлу /var/www/html/ test.html и после этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html (рис. 2.18):

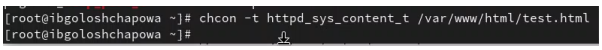

Рис. 2.18: Возврат контекст

15. Исправила обратно конфигурационный файл apache, вернув Listen 80. Удалила привязку http_port_t к 81 порту и проверьила, что порт 81 удалён. Затем удалила файл /var/www/html/test.html (рис. 2.19):

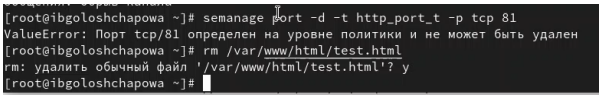

Рис. 2.19: Удаление привязки и файла

3 Выводы

В ходе лабораторной работы мне удалось:

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

4 Библиография

1. Git - система контроля версий

2. Rocky Linux