Отчет по лабораторной работе №6

Информационная безопасность

Нгуен Фыок Дат

Содержание

Список таблиц

Список иллюстраций

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 Задание

- Поиск информации про веб-сервер
- Работа с Html-файлами
- Просмотр лог-файлов

3 Теоретическое введение

SELinux представляет собой систему маркировки, каждый процесс файл, каталог, пользователь, устройство, порт и так далее имеет метку. SELinux определяет правила доступа процесса к объектам с определенными метками. Это называется политикой.

Владелец файла не имеет полной свободы действий над атрибутами безопасности. Стандартные атрибуты контроля доступа, такие как группа и владелец ничего не значат для SELinux. Полностью все управляется метками. Значения атрибутов могут быть установлены и без прав root, но на это нужно иметь специальные полномочия SELinux.

SELinux может работать в трех режимах — отключен, система полностью отключена и не работает, режим ограничений Enforcing — программа активирована и блокирует все не соответствующие политикам действия и третий режим Permissive — только фиксировать нарушения.

Политики SELinux бывают тоже нескольких типов. Политика targeted относится к типу Type Enforcment (TE) политик, в которых управление доступом к файлам осуществляется на основе ролей. Сюда же относится политика strict. Есть ещё политики Multi-Level Security (MLS), в которых добавлены дополнительные категории.

Более подробно о см. в [1,2].

4 Выполнение лабораторной работы

В качестве первого шага лабораторной работы мы проверили режим работы SELinux с помощью команд getenforce и sestatus (рис. 4.1).

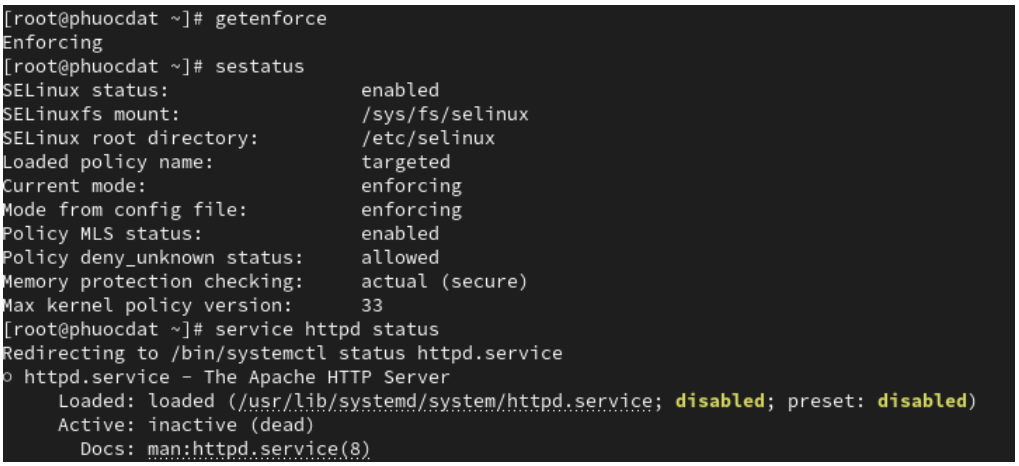

Рис. 4.1: Проверка режима работы

Далее мы проверили, работает ли веб-сервер (рис. 4.2), и запустили его, так как он не работал (рис. 4.3).

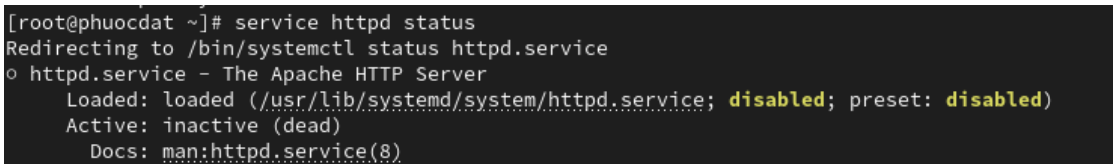

Рис. 4.2: Проверка работы веб-сервера

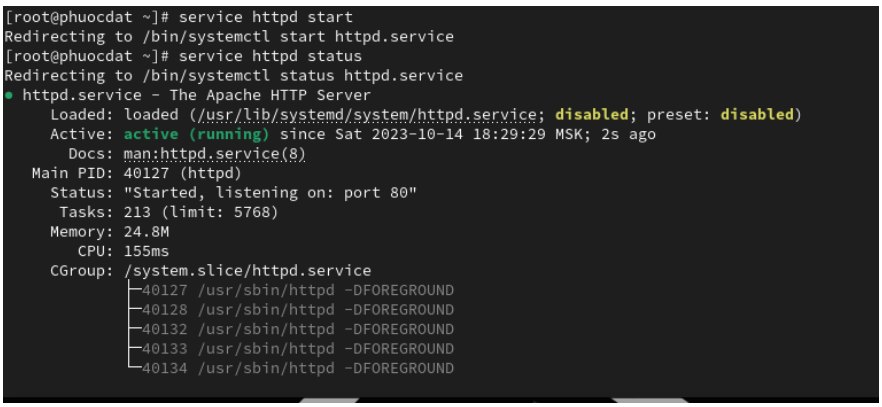

Рис. 4.3: Запуск веб-сервера

Определили контекст безопасности процесса веб-сервера (рис. 4.4). Главной информацией для нас стал тип процесса — httpd_t.

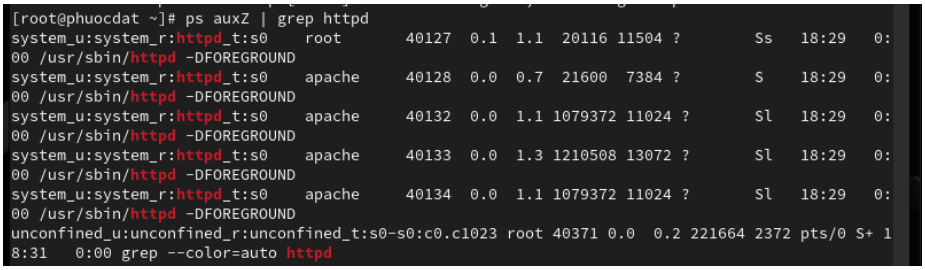

Рис. 4.4: Определение контекста безопасности

Посмотрели текущее положение переключателей SELinux, большинство из них находятся в выключенном состоянии (рис. 4.5).

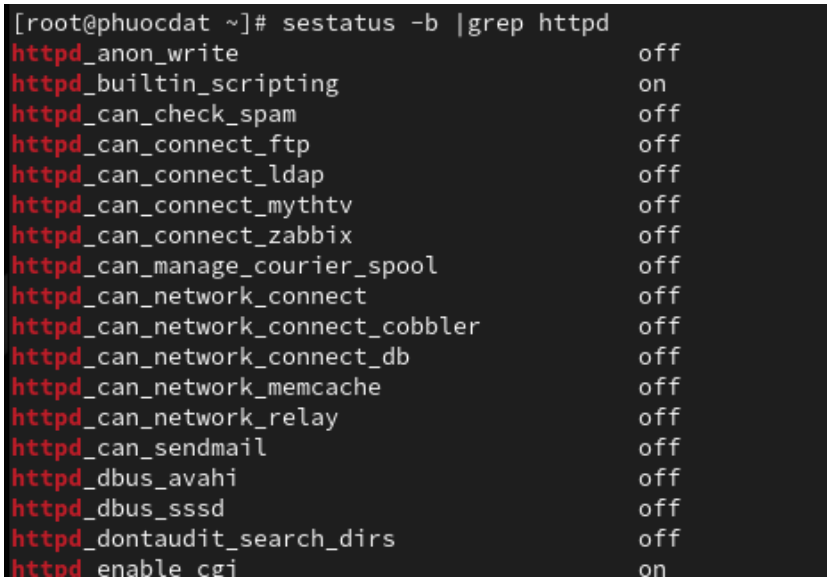

Рис. 4.5: Просмотр переключателей

Посмотрели статистику по политике с помощью seinfo (рис. 4.6). Определили, что множество пользователей имеет размер 8, множество ролей — 14, а множество типов — 5100.

```
[root@phuocdat ~]# 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:             65009    Neverallow:            0
  Auditallow:          170    Dontaudit:          8572
  Type_trans:       265337    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
```

Рис. 4.6: Статистика по политике

Определили тип файлов и поддиректорий директории /var/www (рис. 4.7). Поддиректория cgi-bin имеет тип httpd_sys_script_exec_t, а html — httpd_sys_content_t. Только пользователь-владелец имеет право создавать файлы в папке html.

```
[root@phuocdat ~]# ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 May 16 23:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0     6 May 16 23:21 html
[root@phuocdat ~]# ls -lZ /var/www/html
total 0
```

Рис. 4.7: Определение типов поддиректорий

Создали файл test.html в папке html от лица суперпользователя (рис. 4.8).

```
[root@phuocdat ~]# su
[root@phuocdat ~]# touch /var/www/html/test.html
[root@phuocdat ~]# echo '<html>' > /var/www/html/test.html
[root@phuocdat ~]# echo '<body>test</body>' >> /var/www/html/test.html
[root@phuocdat ~]# echo '</html>' >> /var/www/html/test.html
```

Рис. 4.8: Создание файла test.html

Проверили его контекст (рис. 4.9). Вновь созданным файлам в папке html по умолчанию присваивается тип httpd_sys_content_t.

```
[root@phuocdat ~]# ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 14 18:43 /var/www/ht
ml/test.html
```

Рис. 4.9: Проверка контекста test.html

Обратились к файлу через веб-сервер и увидели его содержимое (рис. 4.10).

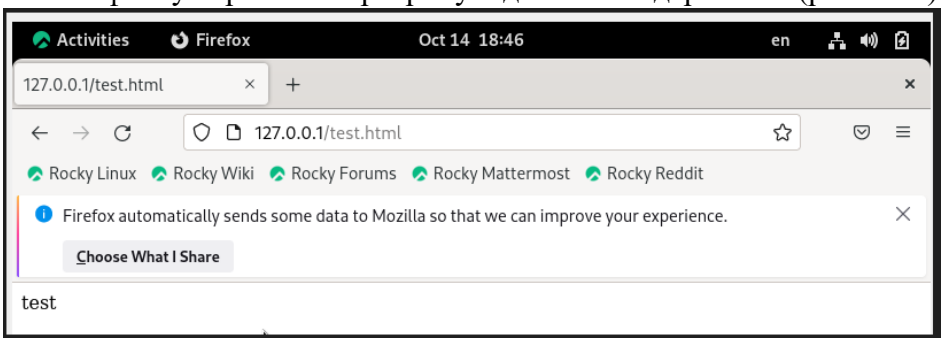

Рис. 4.10: Просмотр файла в браузере

Снова проверили контекст файла и поменяли его на другой (рис. 4.11). Новый контекст файла не позволяет процессу httpd получить доступ к файлу при обращении к нему через браузер.

```
[root@phuocdat ~]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@phuocdat ~]# chcon -t samba_share_t /var/www/html/test.html
[root@phuocdat ~]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Рис. 4.11: Смена контекста

Попробовали открыть файл в браузере (рис. 4.12). Возникла ошибка из-за нового контекста.

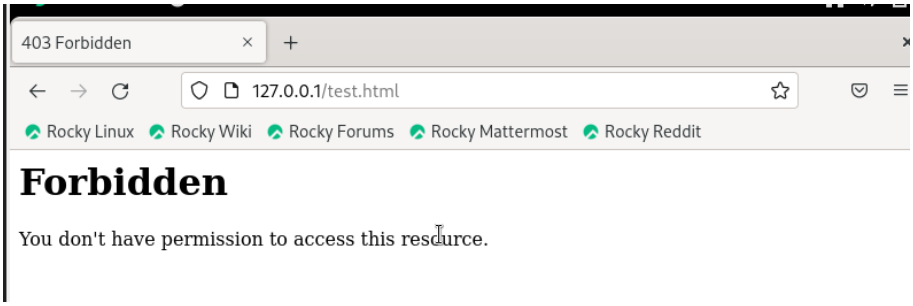

Рис. 4.12: Просмотр файла в браузере

Посмотрели лог-файлы (рис. 4.13). Увидели запись о неудачной попытке браузера получить доступ к файлу (ошибка 403).

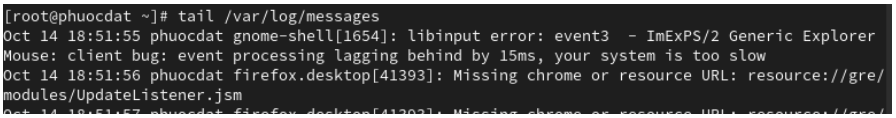

Рис. 4.13: Чтение лог-файлов

Изменили порт в конфигурационном файле httpd.conf с 80 на 81 (рис. 4.14).

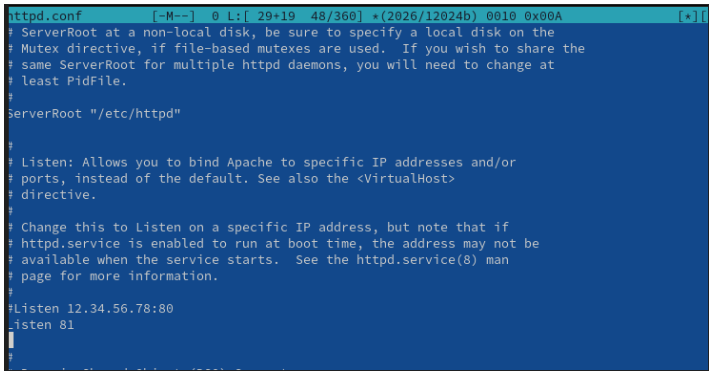

Рис. 4.14: Смена порта

Перезапустили веб-сервер, получили сообщение о том, что он запущен на прослушивание 81 порта (рис. 4.15).

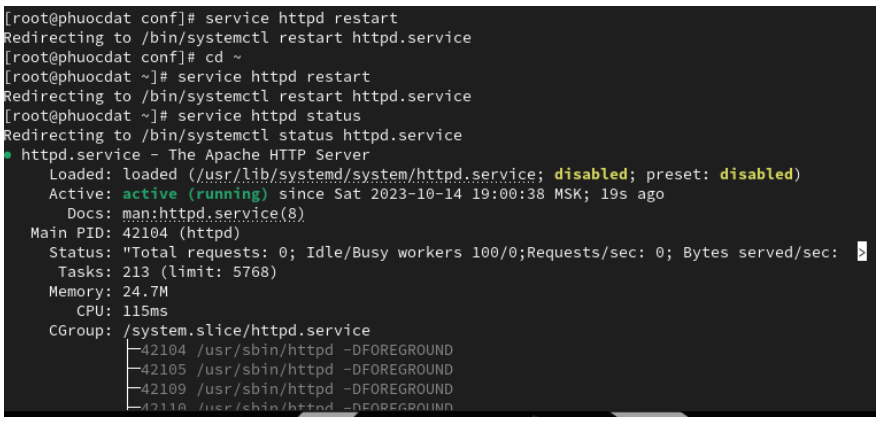

Рис. 4.15: Перезапуск веб-сервера

Проверили лог-файлы и нашли информацию о переключении веб-сервера на прослушивание нового порта (рис. 4.16).

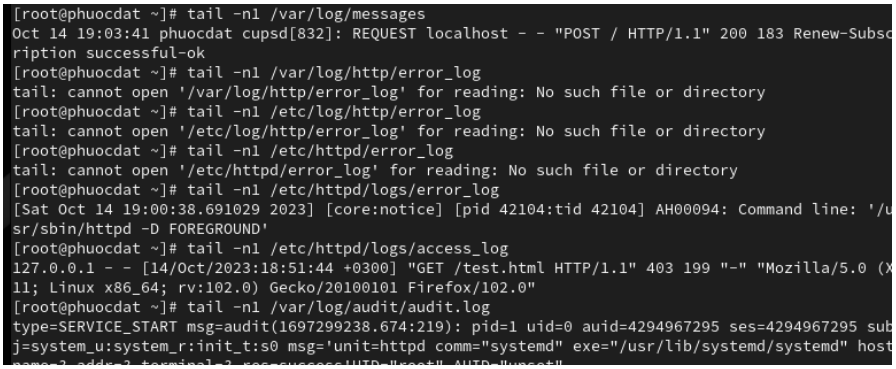

Рис. 4.16: Проверка лог-файлов

Посмотрели список портов веб-сервера, нашли там указанный нами порт (рис. 4.17).

Рис. 4.17: Просмотр списка портов

Вернули файлу test.html старый контекст (рис. 4.18).

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

Рис. 4.18: Смена контекста

Получили доступ к файлу через веб-сервер в браузере (рис. 4.19).

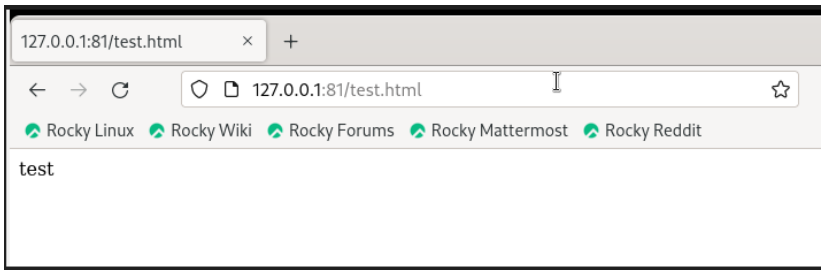

Рис. 4.19: Просмотр файла

Вернули порт 80 в конфигурационном файле (рис. 4.20).

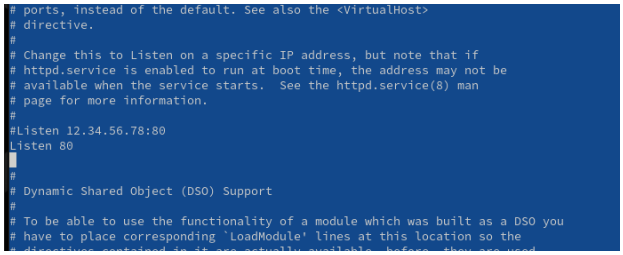

Рис. 4.20: Изменение порта

Попытались удалить 81 порт, но столкнулись с ошибкой, что он определен на уровне политики и не может быть удален (рис. 4.21). После этого удалили файл test.html.

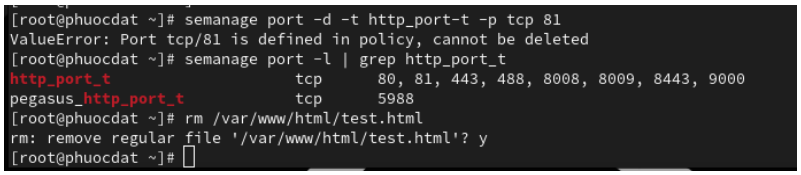

Рис. 4.21: Попытка удаления порта

5 Выводы

В результате лабораторной работы я получила базовые навыки администрирования ОС Linux, познакомилась с технологией SELinux и проверила ее работу на практике совместно с веб-сервером Apache.

Список литературы

1. Мандатное разграничение прав в Linux [Электронный ресурс]. URL: https://esystem.rudn.ru/pluginfile.php/2090282/mod_resource/content/2/006-lab_selinux.pdf.
2. Настройка SELinux [Электронный ресурс]. URL: https://losst.pro/nastrojka-selinux.