Основы информационной безопасности

Лабораторная работа № 6. Мандатное разграничение прав в Linux

Подлесный Иван Сергеевич

Содержание

# Цель работы

Целью данной работы является приобретение практических навыков администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux. Проверить работу SELinx на практике совместно с веб-сервером Apache.

# Выполнение лабораторной работы

В конфигурационном файле /etc/httpd/httpd.conf зададим параметр ServerName. Отключим фильтр командами(рис. fig. 1)

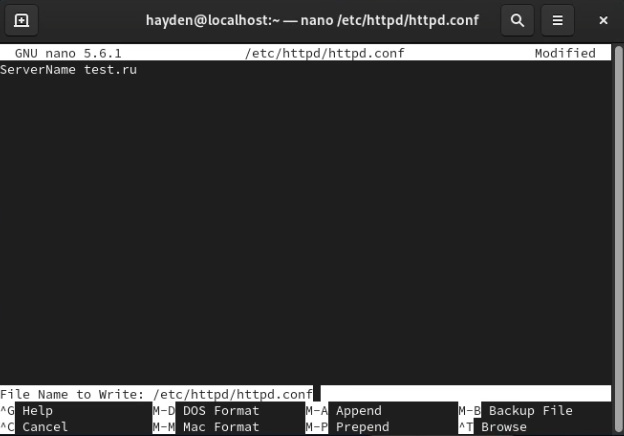


Рис. 1: Подготовка лабораторного стенда

Войдем в систему с полученными учётными данными и убедимся, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus(рис. fig. 2).

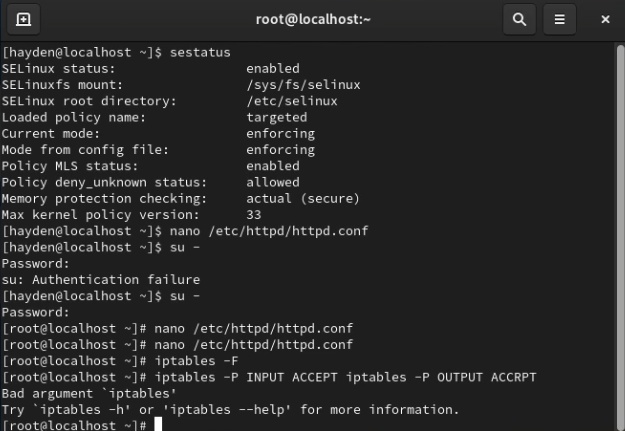


Рис. 2: Проверка статуса SELinux

Проверим, что веб-сервер работает (рис. fig. 3).

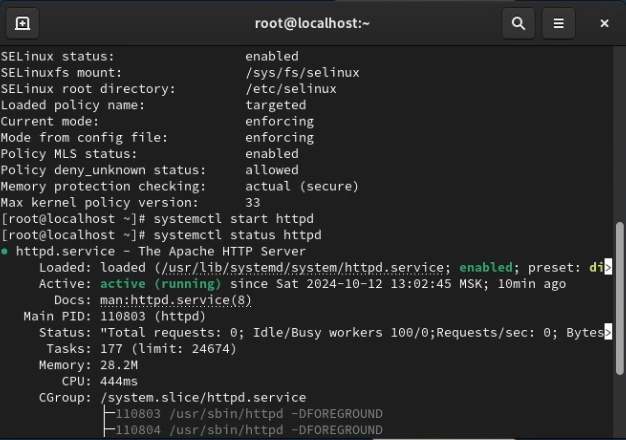


Рис. 3: Проверка статуса веб-сервера

Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности(рис. fig. 4)

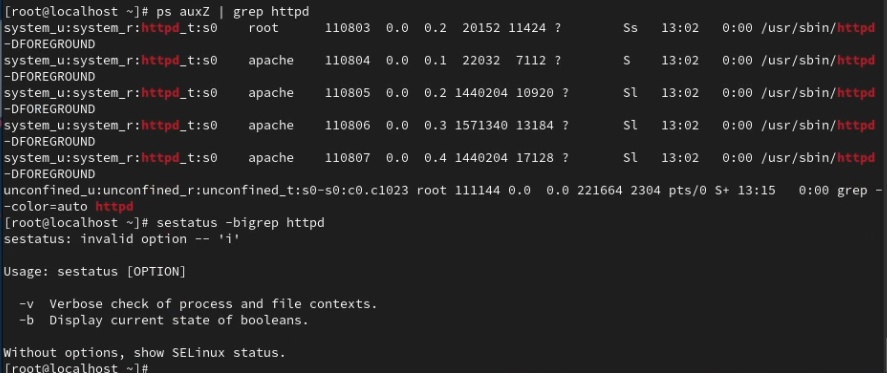


Рис. 4: Просмотр контекста безопасности веб-сервера

Посмотрим текущее состояние переключателей SELinux для Apache(рис. fig. 5)

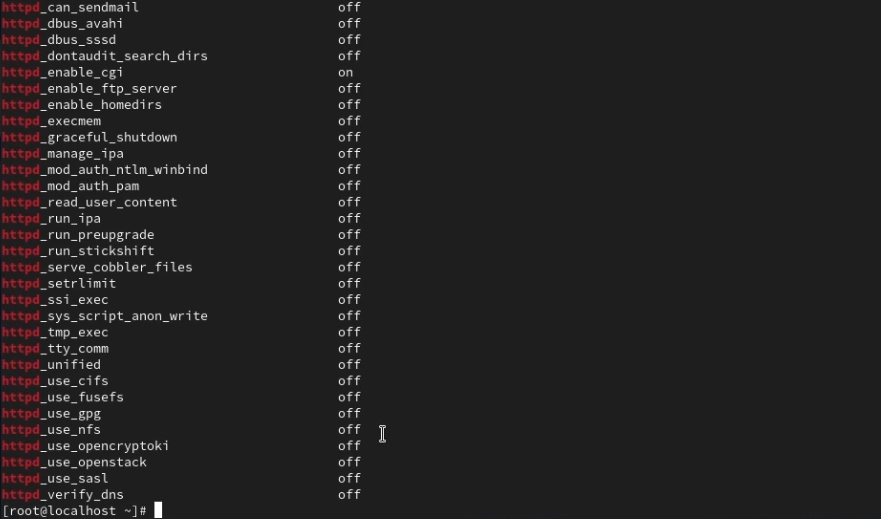


Рис. 5: Состояние переключателей SELinux для Apache

Посмотрим статистику по политике с помощью команды seinfo(рис. fig. 6):



Рис. 6: Статистика по политике

Определив тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды ls -lZ /var/www, увидим, что есть директория, содержащая cgi-скрипты, и директория /var/www/html, содержащая все скрипты httpd(в данный момент пустая)(рис. fig. 7):

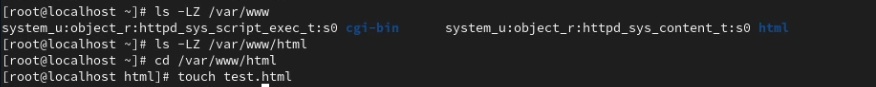


Рис. 7: Просмотр типов директорий в /var/www

Можно увидеть, что создание файлов в директории /var/www/html разрешено только владельцу – root.

Создадим от имени суперпользователя html-файл /var/www/html/test.html следующего содержания(рис. fig. 8):

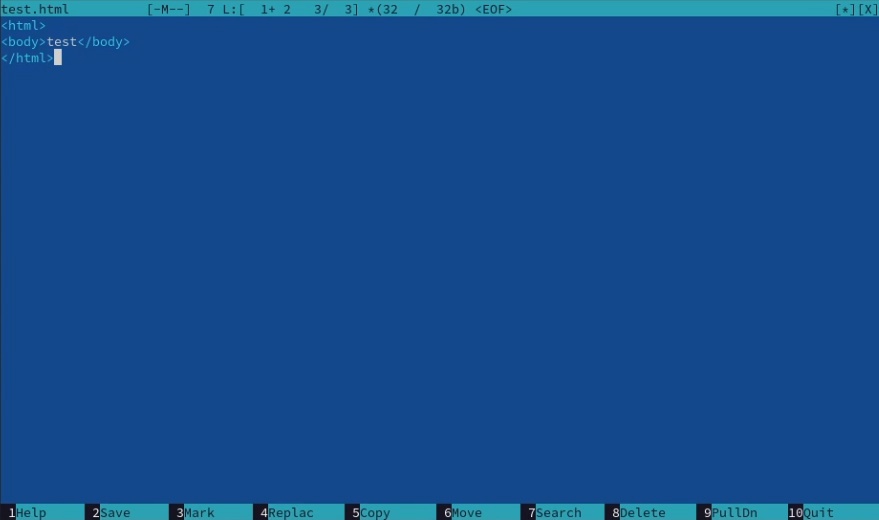


Рис. 8: Содержимое html-файла /var/www/html/test.html

Посмотрим контекст безопасности, заданный по умолчанию для html документа(fig. 9):

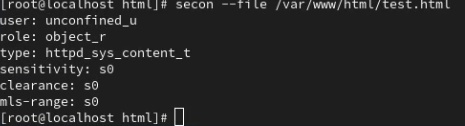


Рис. 9: Установка пароля для пользователя с правами администратора

Увидим, что файлам по умолчанию сопоставляется свободный пользователь SELinux unconfined\_u, указана роль object\_r используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах и тип httpd\_sys\_content\_t, который позволяет процессу httpd получить доступ к файлу

Обратимся к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html, убедимся, что файл был успешно отображён.(рис. fig. 10):

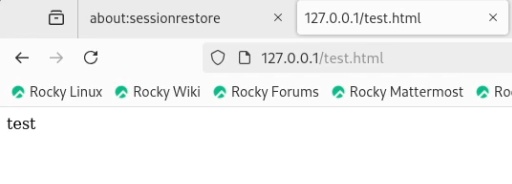


Рис. 10: Открытие html-страницы через браузер

Изучим справку man httpd\_selinux (через интернет, ибо команда не работает), выясним, какие контексты файлов определены для httpd. Сопоставив их с типом файла test.html увидим, что его контекст httpd\_sys\_content\_t для содержимого, которое должно быть доступно для всех скриптов httpd и для самого демона. Изменим контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на тот, к которому процесс httpd не должен иметь доступа – samba\_share\_t(рис. fig. 11):

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

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

Теперь снова попробуем получить доступ к файлу через браузер и получим отказ(рис. fig. 12):

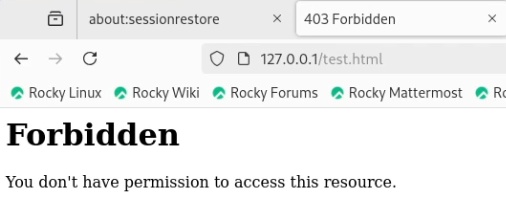


Рис. 12: Отказ в доступе к html-странице через браузер

Посмотрим log-файлы веб-сервера Apache и системный лог-файл и увидим, что отказ происходит, так как доступ запрещен SELinux именно к веб-серверу(на просто просмтр текстовых файлов это не влияет)(рис. fig. 13):

Рис. 13: Просмотр лог-файлов

Рис. 13: Просмотр лог-файлов

Запустим веб-сервер Apache на прослушивание ТСР-порта 81. Для этого в файле /etc/httpd/httpd.conf найдем строчку Listen 80 и заменим её на Listen 81(рис. fig. 14):

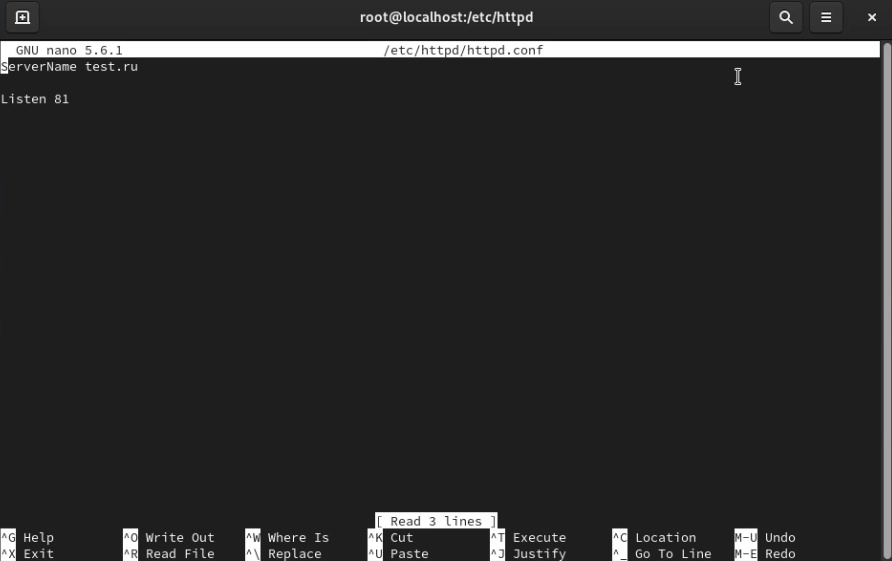


Рис. 14: Замена прослушиваемого порта

Выполним перезапуск веб-сервера Apache и не увидим изменений по не понятным мне причинам, несмотря на то, что 81 порт не является официальным портом для доступа по TCP(рис. fig. 15):

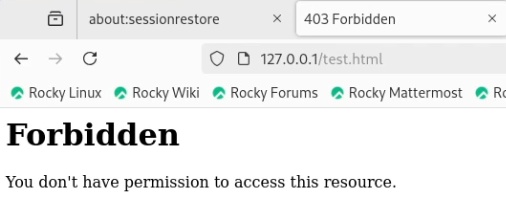


Рис. 15: Открытие html-страницы через браузер при прослушивании 81 порта

Просмотрев лог-файлы увидим, что порт для прослушивания был сменен(рис. fig. 16):

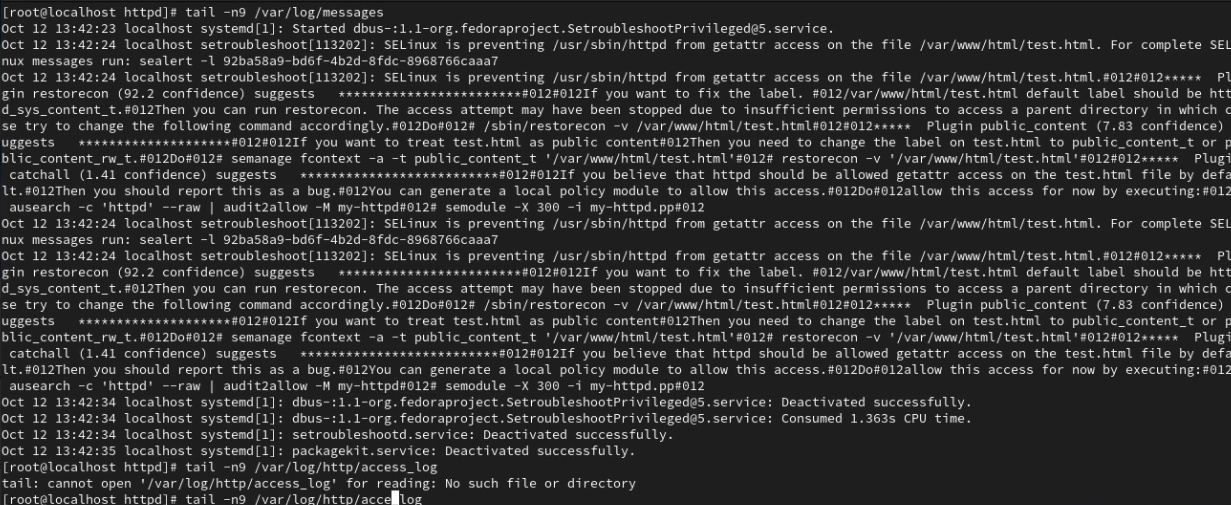


Рис. 16: Просмотр лог-файлов

Также этот порт мог быть отключен, тогда мы бы совсем не видели страницу, добавлять порты и просматривать актуальные можно с помощью команды seamanage(рис. fig. 17):

Рис. 17: Просмотр портов с помощью seamnage

Рис. 17: Просмотр портов с помощью seamnage

# Выводы

В результате выполнения работы были получены базовые навыки работы с технологией SELinux. Проверена работа SELinx на практике совместно с веб-сервером Apache.