Отчёт по лабораторной работе №6

Информационная безопасность

Арбатова Варвара Петровна

Содержание

# Цель работы

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

# Теоретическое введение

SELinux (Security-Enhanced Linux) обеспечивает усиление защиты путем внесения изменений как на уровне ядра, так и на уровне пространства пользователя, что превращает ее в действительно «непробиваемую» операционную систему. Впервые эта система появилась в четвертой версии CentOS, а в 5 и 6 версии реализация была существенно дополнена и улучшена. SELinux имеет три основных режим работы:

Enforcing: режим по умолчанию. При выборе этого режима все действия, которые каким-то образом нарушают текущую политику безопасности, будут блокироваться, а попытка нарушения будет зафиксирована в журнале.

Permissive: в случае использования этого режима, информация о всех действиях, которые нарушают текущую политику безопасности, будут зафиксированы в журнале, но сами действия не будут заблокированы.

Disabled: полное отключение системы принудительного контроля доступа.

Политика SELinux определяет доступ пользователей к ролям, доступ ролей к доменам и доступ доменов к типам. Контекст безопасности — все атрибуты SELinux — роли, типы и домены. Более подробно см. в [@f].

Apache — это свободное программное обеспечение, с помощью которого можно создать веб-сервер. Данный продукт возник как доработанная версия другого HTTP-клиента от национального центра суперкомпьютерных приложений (NCSA). Для чего нужен Apache сервер:

чтобы открывать динамические PHP-страницы,

для распределения поступающей на сервер нагрузки,

для обеспечения отказоустойчивости сервера,

чтобы потренироваться в настройке сервера и запуске PHP-скриптов.

Apache является кроссплатформенным ПО и поддерживает такие операционные системы, как Linux, BSD, MacOS, Microsoft, BeOS и другие.

Более подробно см. в [@s].

# Выполнение лабораторной работы

Перехожу в корневую директорию и устанавливаю httpd

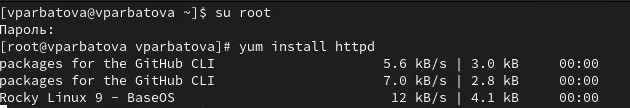


Figure 1: Установка

Перехожу в директорию /etc/httpd, чтобы настроить веб-сервер. Добавляю строку ServerName test.ru в файл httpd.conf, указывая имя сервера. Очищаю правила iptables, чтобы сбросить настройки фаервола. Устанавливаю политику ACCEPT для входящих подключений, разрешая входящий трафик. Устанавливаю политику ACCEPT для исходящих подключений, разрешая исходящий трафик.

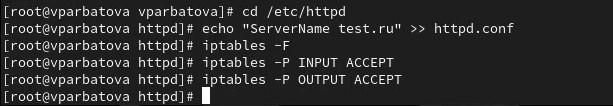


Figure 2: Выполняю настройки

Убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus

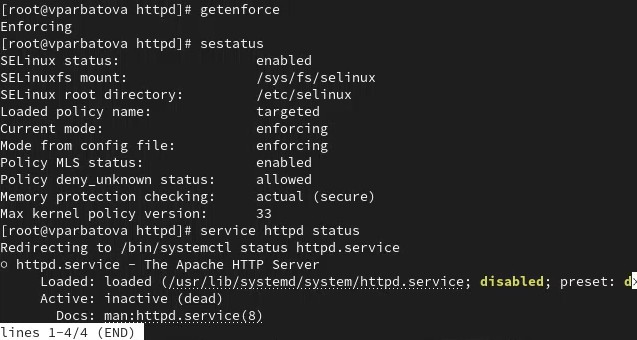


Figure 3: Убеждаюсь в работе

Запускаю сервер apache, далее обращаюсь с помощью браузера к веб-серверу, запущенному на компьютере, он работает, что видно из вывода команды service httpd status (рисунки 4-5)

Figure 4: Запуск сервера

Figure 4: Запуск сервера

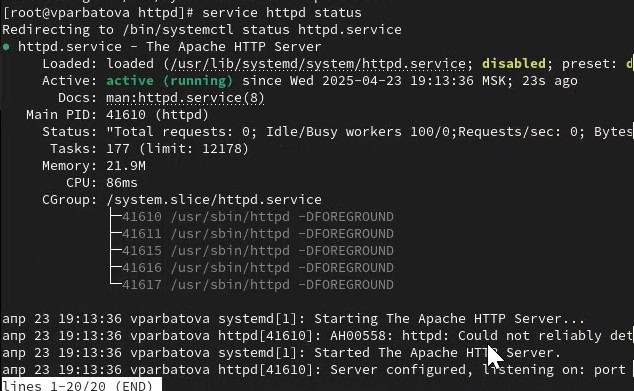


Figure 5: Статус работы

С помощью команды ps auxZ | grep httpd нашла веб-сервер Apache в списке процессов. Его контекст безопасности - httpd\_t

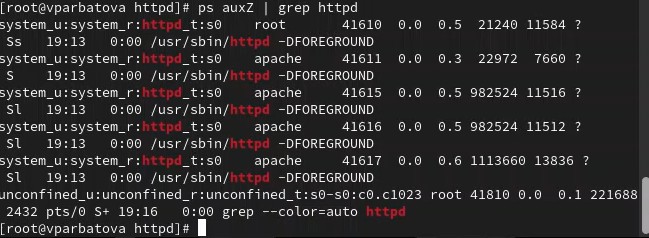


Figure 6: Ищу контекст безопасности

Просмотрела текущее состояние переключателей SELinux для Apache с помощью команды sestatus -bigrep httpd

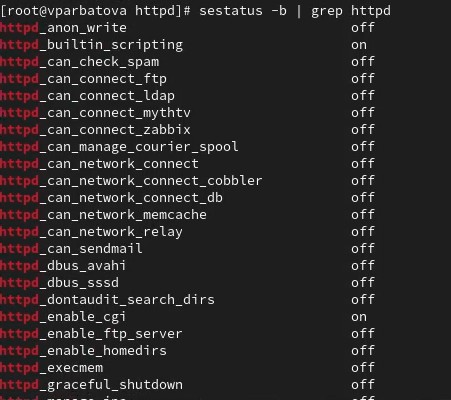


Figure 7: Просмотр текущего состояния

Просмотрела статистику по политике с помощью команды seinfo. Множество пользователей - 8, ролей - 39, типов - 5135

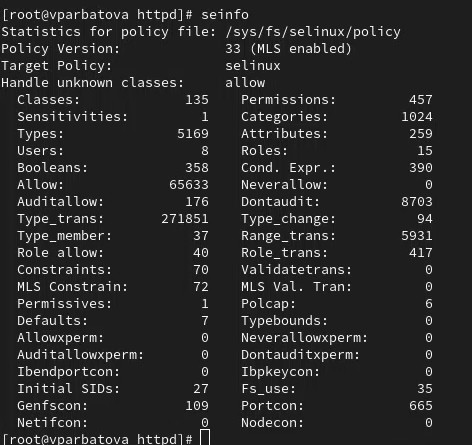


Figure 8: Просмотр статистики по политике

Типы поддиректорий, находящихся в директории /var/www, с помощью команды ls -lZ /var/www следующие: владелец - root, права на изменения только у владельца. Файлов в директории нет

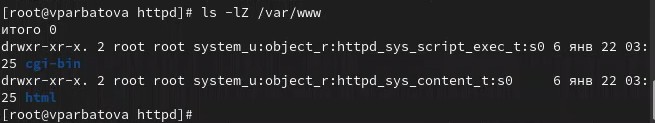


Figure 9: Смотрю права

В директории /var/www/html нет файлов.



Figure 10: Файлы директории

Создаю файл

Figure 11: Создание файла

Figure 11: Создание файла

Создать файл может только суперпользователь, поэтому от его имени создаем файл touch.html cо следующим содержанием

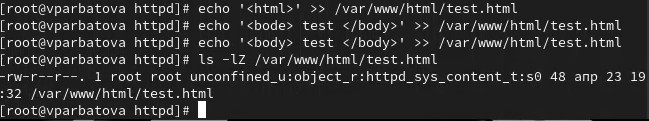


Figure 12: Заполняю файл

Перехожу на сайт и смотрю. Всё удачно

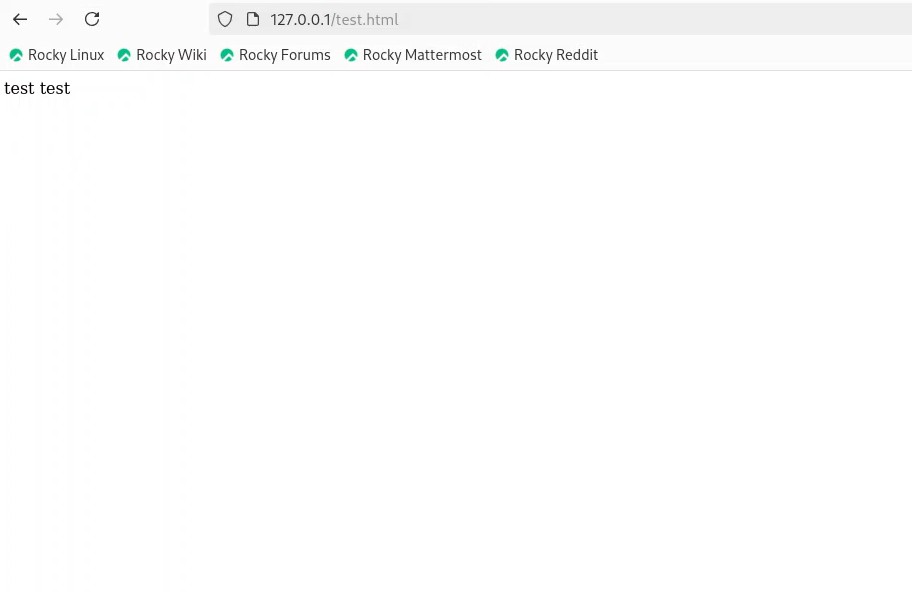


Figure 13: Сайт

Изменяю 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 Контекст действительно поменялся

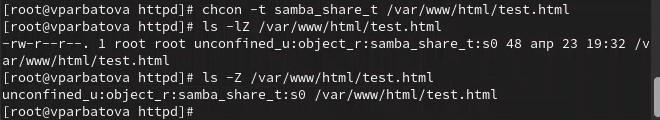


Figure 14: Изменение контекста

Доступ запрещен

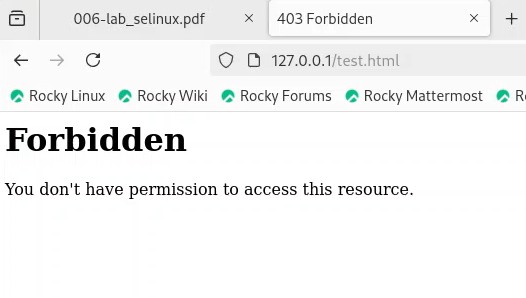


Figure 15: Сайт

файл не был отображён, хотя права доступа позволяют читать этот файл любому пользователю, потому что установлен контекст, к которому процесс httpd не должен иметь доступа.

Просматриваю log-файлы веб-сервера Apache и системный лог-файл: tail /var/log/messages. Если в системе окажутся запущенными процессы setroubleshootd и audtd, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log

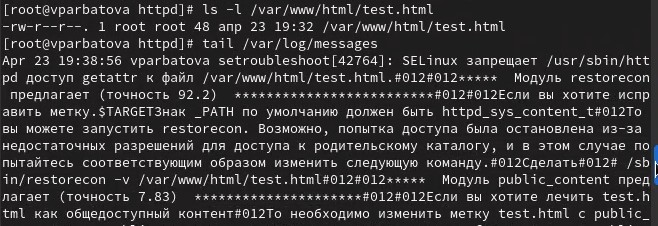


Figure 16: Проверка

# Выводы

В ходе выполнения данной лабораторной работы были развиты навыки администрирования ОС Linux, получено первое практическое знакомство с технологией SELinux и проверена работа SELinux на практике совместно с веб-сервером Apache.

# Список литературы