Отчёт по лабораторной работе 6

Соболев Максим Сергеевич

Содержание

# 1 Мандатное разграничение прав в Linux

# 2 Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux

Проверить работу SELinx на практике совместно с веб-сервером Apache

# 3 Задание

Исследовать технологию SELinux Исследовать работу SELinx на практике совместно с веб-сервером Apache

# 4 Теоретическое введение

SELinux — реализация системы принудительного контроля доступа, которая может работать параллельно с классической избирательной системой контроля доступа.

Apache HTTP-сервер — свободный веб-сервер. Apache является кроссплатформенным ПО, поддерживает операционные системы Linux, BSD, Mac OS, Microsoft Windows, Novell NetWare, BeOS. Основными достоинствами Apache считаются надёжность и гибкость конфигурации.

# 5 Выполнение лабораторной работы

## 5.1 Шаг 1

Входим в систему с полученными учётными данными и убеждаемся, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus

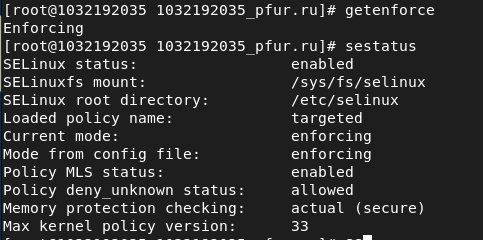


Рис. 1: 1

## 5.2 Шаг 2

Проверяем, что apache работает: systemctl status httpd

Рис. 2: 2

Рис. 2: 2

## 5.3 Шаг 3

Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности.

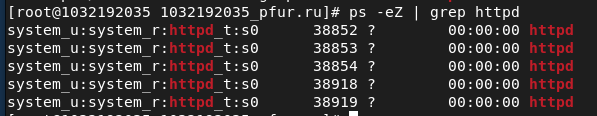


Рис. 3: 3

## 5.4 Шаг 4

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

Обратим внимание, что многие из них находятся в положении «off».

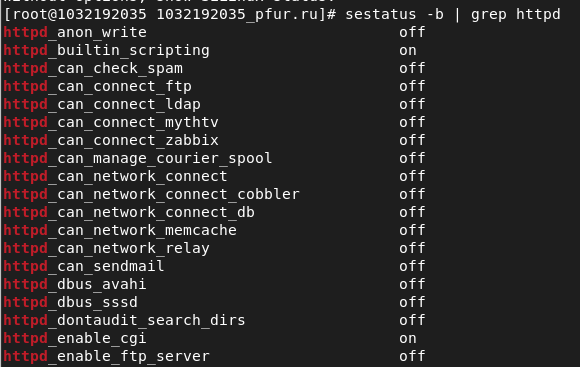


Рис. 4: 4

## 5.5 Шаг 5

Посмотрим статистику по политике с помощью команды seinfo, также определите множество пользователей, ролей, типов.



Рис. 5: 5

## 5.6 Шаг 6

Определим тип файлов и поддиректорий, находящихся в директории /var/www/, с помощью команды ls -lZ /var/www/

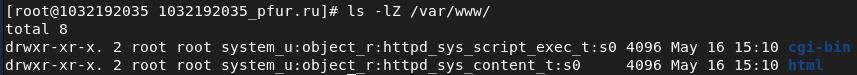


Рис. 6: 6

## 5.7 Шаг 7

Определим тип файлов, находящихся в директории /var/www/html/: ls -lZ /var/www/html/

Рис. 7: 7

Рис. 7: 7

Файлов нет

## 5.8 Шаг 8

Определим круг пользователей, которым разрешено создание файлов в директории /var/www/html/.

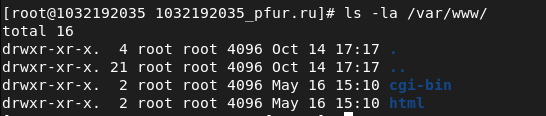


Рис. 8: 8

Создание файлов разрешено только пользователю root

## 5.9 Шаг 9

Создадим от имени суперпользователя html-файл /var/www/html/test.html следующего содержания:

test

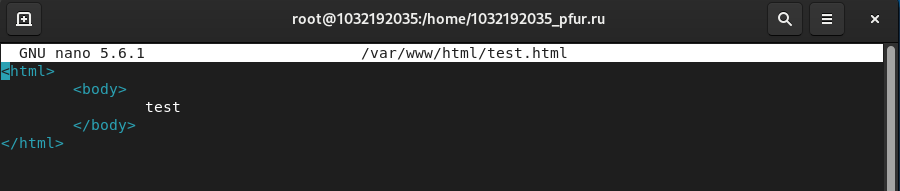


Рис. 9: 9

## 5.10 Шаг 10

Проверим контекст созданного нами файла.

Рис. 10: 10

Рис. 10: 10

## 5.11 Шаг 11

Обращаемся к файлу через веб-сервер, введя в браузере адрес http://localhost/test.html. Убеждаемся, что файл был успешно отображён.

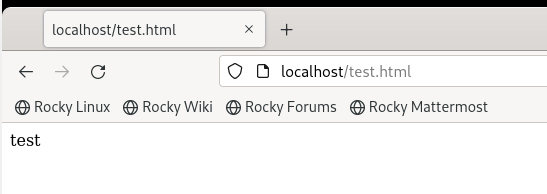


Рис. 11: 11

## 5.12 Шаг 12

Изучим справку man httpd\_selinux и выясним, какие контексты файлов определены для httpd.

Изучили.

## 5.13 Шаг 13

Изменяем контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на samba\_share\_t, к которому процесс httpd не должен иметь доступа: chcon -t samba\_share\_t /var/www/html/test.html ls -Z /var/www/html/test.html

После этого проверяем, что контекст поменялся.

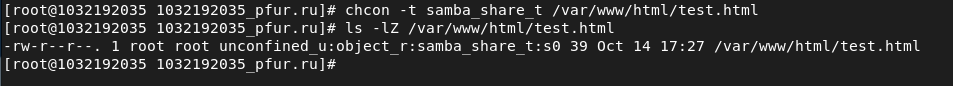


Рис. 12: 13

## 5.14 Шаг 14

Попробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://localhost/test.html. Получаем сообщение об ошибке: Forbidden You don’t have permission to access /test.html on this server.

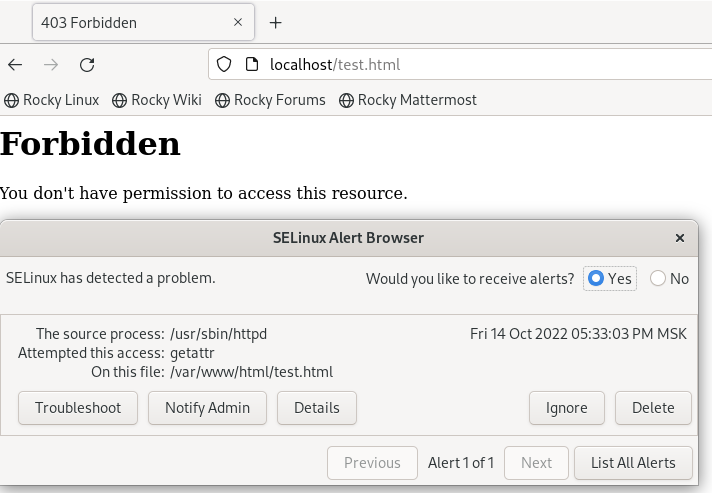


Рис. 13: 14

## 5.15 Шаг 15

Файл не отображён, поскольку процесс httpd не имеет доступа к файлам с заданным процессом. SELinux не выдаёт мандат на чтение, таким образом запрещая чтение файла

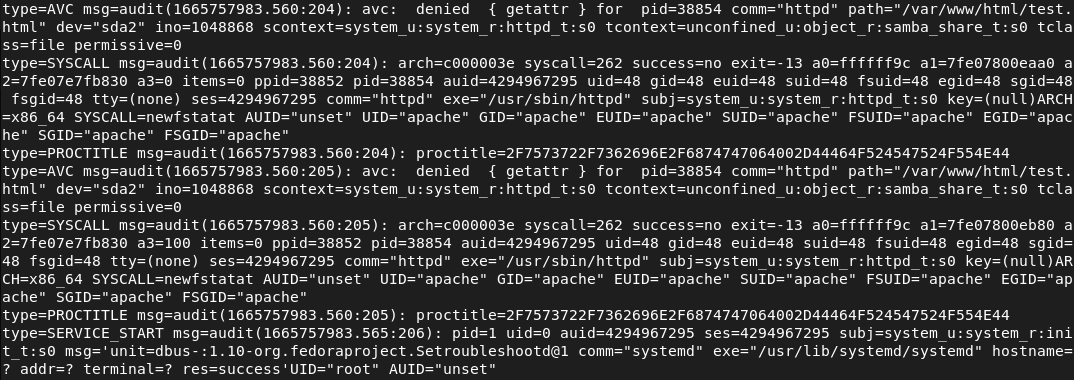


Рис. 14: 15

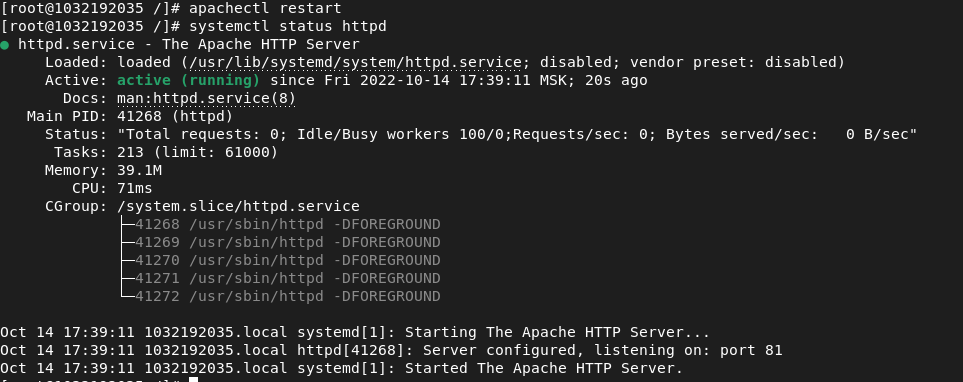
## 5.16 Шаг 16

Пробуем запустить веб-сервер Apache на прослушивание ТСР-порта 81.

Рис. 15: 16

Рис. 15: 16

## 5.17 Шаг 17

Выполните перезапуск веб-сервера Apache. Произошёл сбой? Почему? Нет, не произошёл. Новые версии политик selinux позволяют httpd работать на разных портах, в т.ч. 81. 

## 5.18 Шаг 18

Поменяем в конфиге порт на тот, который действительно не находится в списке разрешённых (8874), попробуем перезапустить httpd, получим ошибку, изучим логи.

## 5.19 Шаг 19

Выполним команду semanage port -a -t http\_port\_t -р tcp 8874 После этого проверим список портов командой semanage port -l | grep http\_port\_t Убеждаемся, что порт 8874 появился в списке.

Рис. 16: 19

Рис. 16: 19

## 5.20 Шаг 20

Да, поняли. Политика selinux не позволяла процессу прослушивать порт

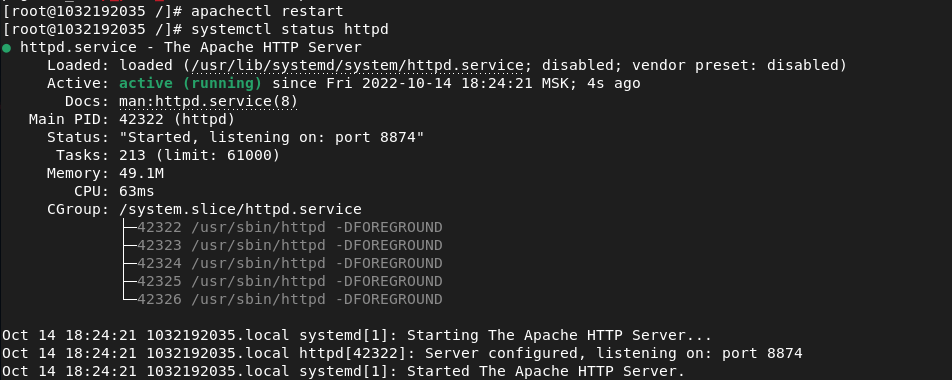


Рис. 17: 20

## 5.21 Шаг 21

Вернем контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html: chcon -t httpd\_sys\_content\_t /var/www/html/test.html После этого попробуем получить доступ к файлу через веб-сервер, введя в браузере адрес http://localhost:8874/test.html. Мы должны увидеть содержимое файла — слово «test».

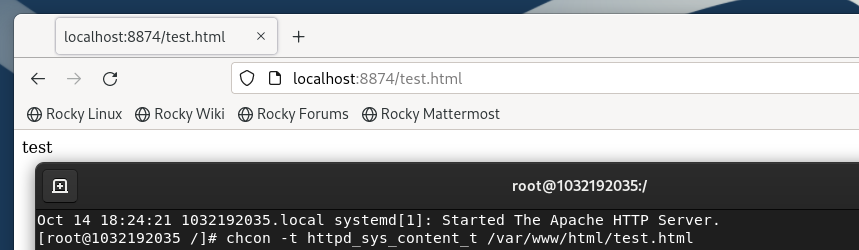


Рис. 18: 21

## 5.22 Шаг 22

Исправим обратно конфигурационный файл apache, вернув Listen 80

Рис. 19: 22

Рис. 19: 22

## 5.23 Шаг 23

Удалим привязку http\_port\_t к 8874 порту: semanage port -d -t http\_port\_t -p tcp 8874 и проверим, что порт 8874 удалён

Рис. 20: 23

Рис. 20: 23

## 5.24 Шаг 24

Удалим файл /var/www/html/test.html: rm /var/www/html/test.html

Рис. 21: 24

Рис. 21: 24

# 6 Выводы

Мы развили навыки администрирования ОС Linux. Получили первое практическое знакомство с технологией SELinux

Проверили работу SELinx на практике совместно с веб-сервером Apache

# Список литературы

1. xattr(7) — Linux manual page // Linux man-pages project URL: https://man7.org/linux/man-pages/man7/xattr.7.html (дата обращения: 30.09.2022).