Отчет по лабораторной работе №6

Дисциплина: Основы информационной безопасности

Иванов Сергей Владимирович

Содержание

# 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 Выполнение лабораторной работы

Убедимся, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus (рис. 1)

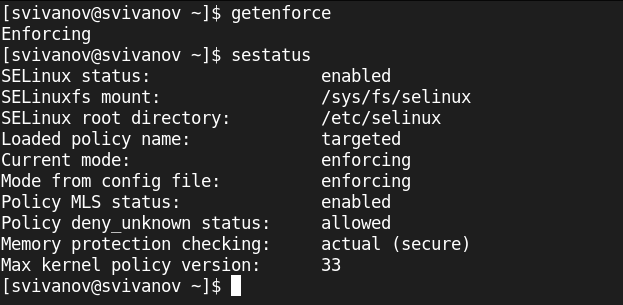


Рис. 1: Проверка режима работы SELinux

Запускаю сервер apache, проверяю статус службы, убеждаюсь что она запущена. (рис. 2).

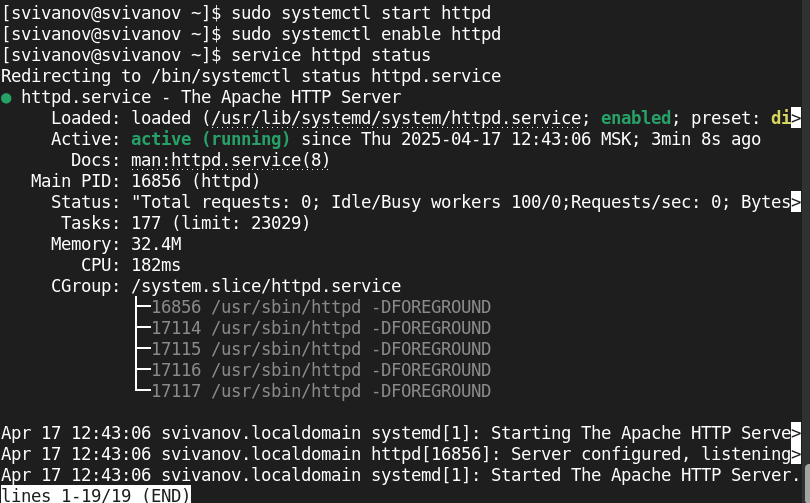


Рис. 2: Проверка службы apache

Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности командой ps -eZ | grep httpd. Его контекст безопасности httpd\_t (рис. 3).

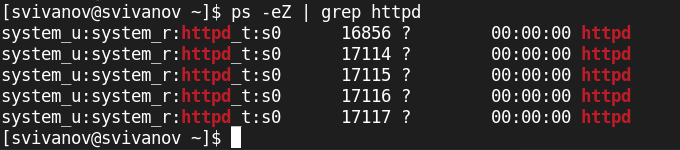


Рис. 3: Проверка работы Apache

Посмотрим текущее состояние переключателей SELinux для Apache с помощью команды sestatus -bigrep httpd (рис. 4).

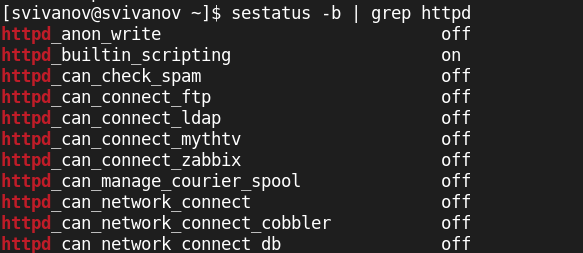


Рис. 4: Состояние переключателей

Посмотрим статистику по политике с помощью команды seinfo, множество пользователей - 8, ролей - 15, типов - 5169 (рис. 5).

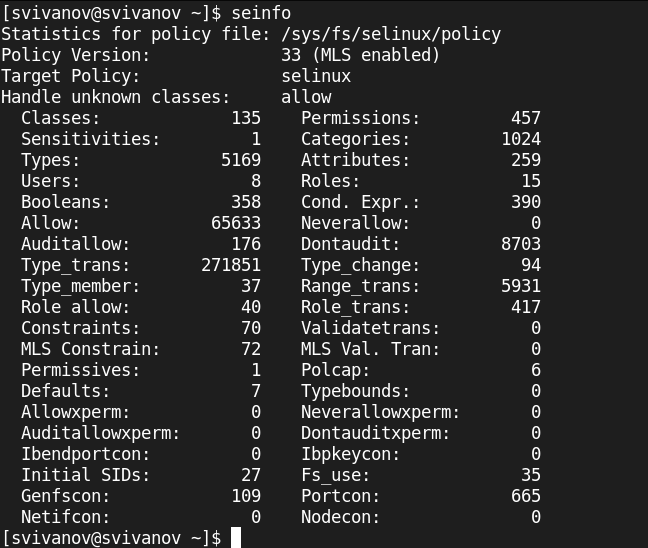


Рис. 5: Статистика по политике

Определим тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды ls -lZ /var/www. Файлов 0, 2 поддиректории, владелец - root (рис. 6).

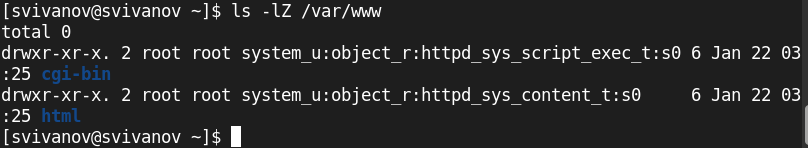


Рис. 6: Типы поддиректорий

Определим тип файлов, находящихся в директории /var/www/html: ls -lZ /var/www/html. Файлов нет (рис. 7).

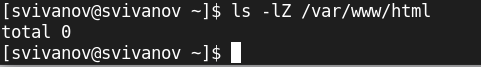


Рис. 7: Тип файлов

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

<html>  
 <body>test</body>  
</html>.

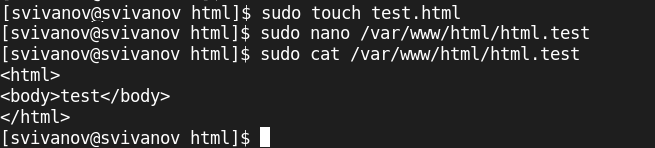


Рис. 8: Создание файла

Проверим контекст созданного файла. Контекст - httpd\_sys\_content\_d (рис. 9).

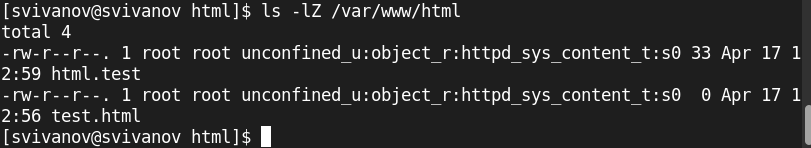


Рис. 9: Проверка контекста

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

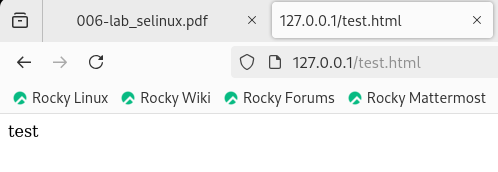


Рис. 10: Отображение файла

Изучим справку man httpd\_selinux. Так как по умолчанию пользователи CentOS являются свободными от типа (unconfined в переводе с англ. свободный), созданному нами файлу test.html был сопоставлен SELinux, пользователь unconfined\_u. Это первая часть контекста. Далее политика ролевого разделения доступа RBAC используется процессами, но не файлами, поэтому роли не имеют никакого значения для файлов. Роль object\_r используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах. (В директории /ргос файлы, относящиеся к процессам, могут иметь роль system\_r. Если активна политика MLS, то могут использоваться и другие роли, например, secadm\_r. Данный случай мы рассматривать не будем, как и предназначение :s0). Тип httpd\_sys\_content\_t позволяет процессу httpd получить доступ к файлу. Благодаря наличию последнего типа мы получили доступ к файлу при обращении к нему через браузер. (рис. 11).

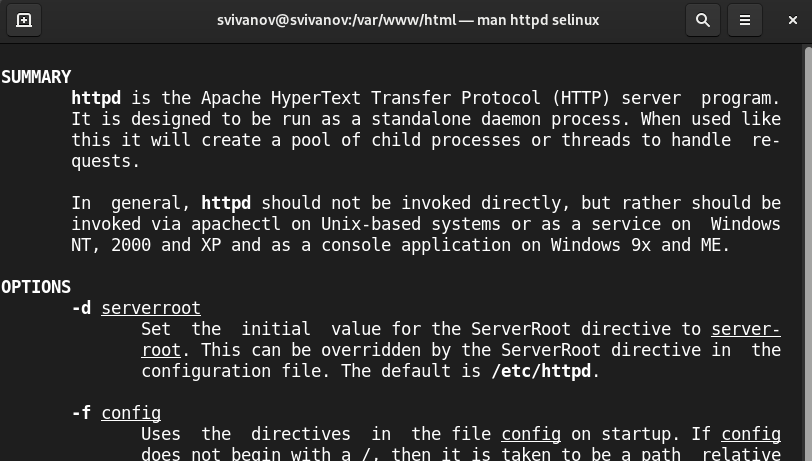


Рис. 11: Изучение справки

Изменим контекст файла /var/www/html/test.html с httpd\_sys\_content\_t, например, на samba\_share\_t: chcon -t samba\_share\_t /var/www/html/test.html ls -Z /var/www/html/test.html (рис. 12).

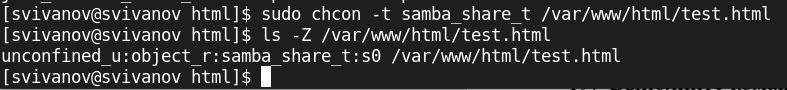


Рис. 12: Изменение контекста файла

Попробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Мы получили сообщение об ошибке: (рис. 13).

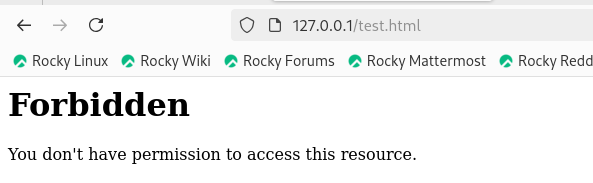


Рис. 13: Отображение файла(ошибка)

Файл не был отображен потому что мы установили контекст, к котрому процесс httpd не имеет доступа. Просмотрим log-файлы веб-сервера Apache. Также просмотрим системный лог-файл: tail /var/log/messages. (рис. 14)

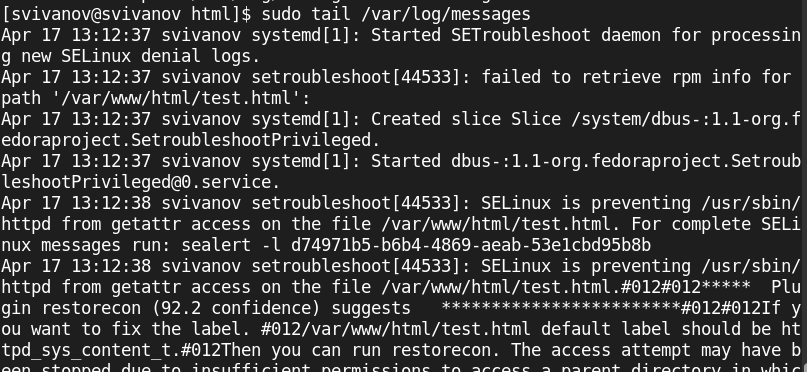


Рис. 14: Просмотр log-файлов

Попробуем запустить 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. 15)

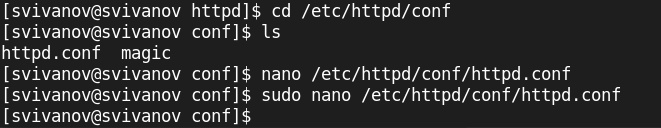


Рис. 15: Изменение номера порта

Изменение порта (рис. 16)

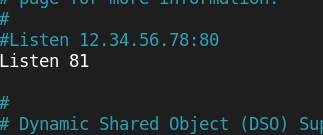


Рис. 16: Изменение порта

Выполним перезапуск веб-сервера Apache. Произошёл сбой, потому что порт 81 не добавлен в список прослушиваемых портов (рис. 17)

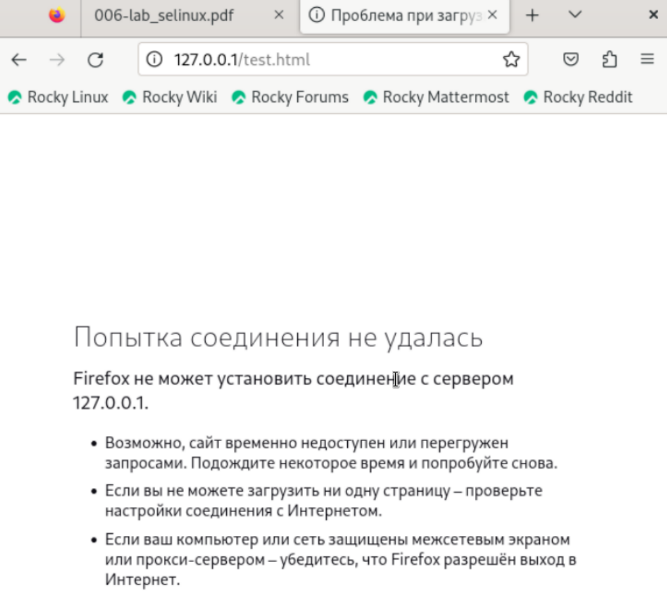


Рис. 17: Попытка прослушивания 81 порта

Проанализируем лог-файлы: tail -nl /var/log/messages (рис. 18)

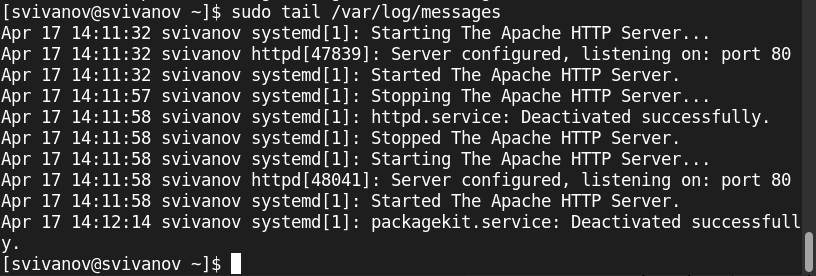


Рис. 18: log-файл

Просмотрим файлы /var/log/http/error\_log, /var/log/http/access\_log и /var/log/audit/audit.log. Записи появились в файле error\_log (рис. 19)

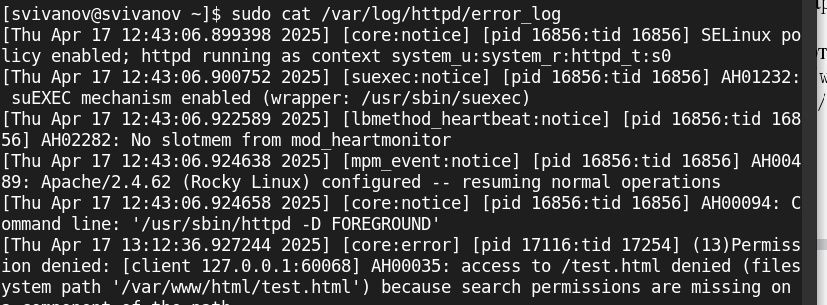


Рис. 19: log-файлы

Выполним команду semanage port -a -t http\_port\_t -р tcp 81 После этого проверим список портов командой semanage port -l | grep http\_port\_t (рис. 20)

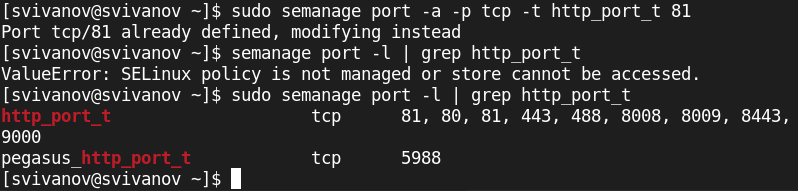


Рис. 20: Добавление и проверка портов

Попробуем запустить веб-сервер Apache ещё раз. Вернем контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html: chcon -t httpd\_sys\_content\_t /var/www/html/test.html. (рис. 21)

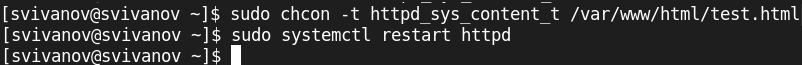


Рис. 21: Перезапуск сервера

Сервер запустился, т.к порт 81 теперь прослушивается. (рис. 22)

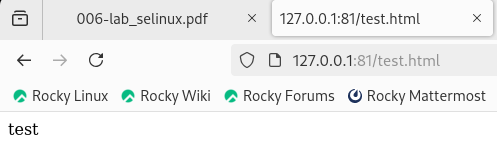


Рис. 22: Проверка сервера

Исправим обратно конфигурационный файл apache, вернув Listen 80. Удалим привязку http\_port\_t к 81 порту: semanage port -d -t http\_port\_t -p tcp 81 и проверим, что порт 81 удалён. (рис. 23)

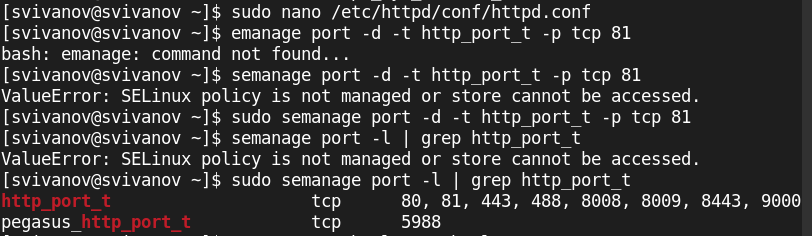


Рис. 23: Удаление порта

Удалим файл /var/www/html/test.html: rm /var/www/html/test.html. (рис. 24)

Удаление файла

Рис. 24: Удаление файла

# 3 Вывод

В ходе работы были развиты навыки администрирования ОС Linux. Получено первое практическое знакомство с технологией SELinux. Проверена работу SELinx на практике совместно с веб-сервером Apache.