Отчёт по лабораторной работе №6

Мандатное разграничение прав в Linux

Ле Тиен Винь

Содержание

[I.Цель работы 1](#_Toc179632581)

[II. Выполнение работы 1](#_Toc179632582)

[1. Подготовка лабораторного стенда 1](#_Toc179632583)

[2. Выполнение работы 2](#_Toc179632584)

[III. Вывод 7](#_Toc179632585)

# I.Цель работы

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

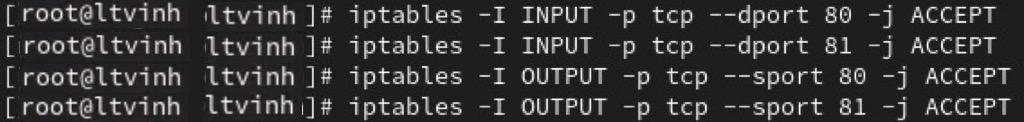
# II. Выполнение работы

## 1. Подготовка лабораторного стенда

* Задать параметр ServerName в конфигурационном файле /etc/httpd/httpd.conf.

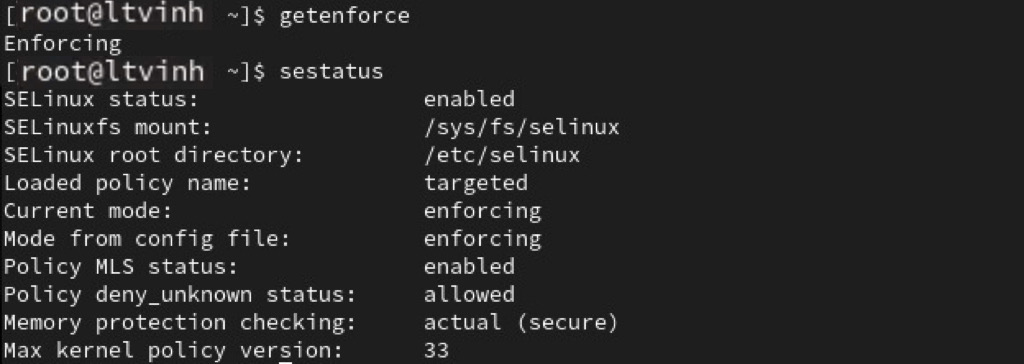


* Проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp.



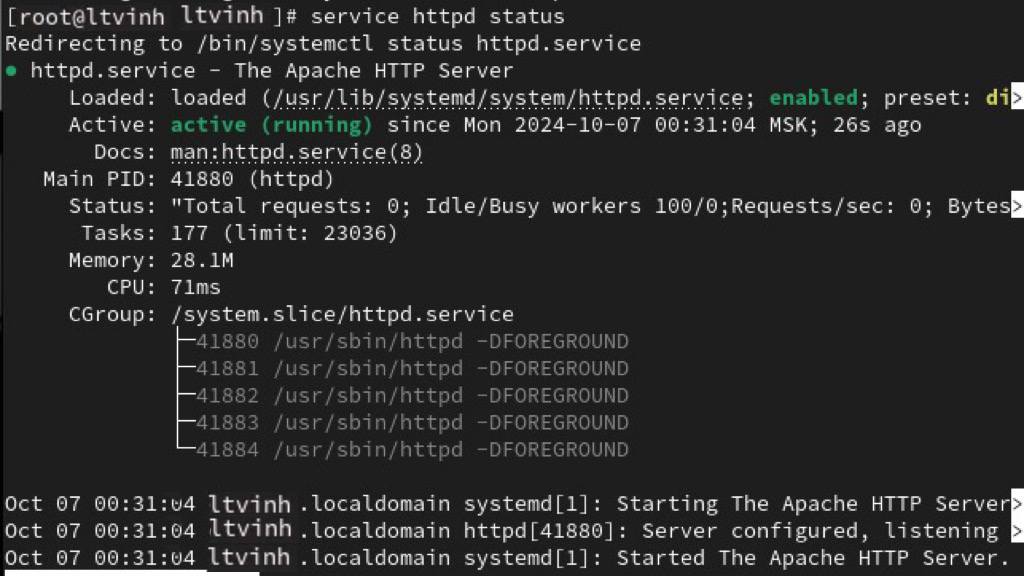
## 2. Выполнение работы

* Убедиться, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus.



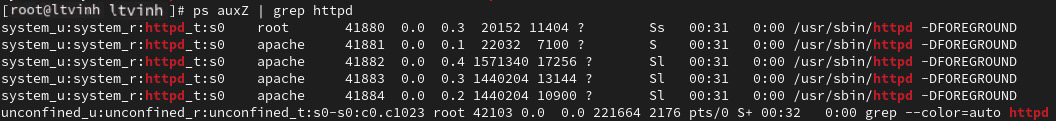
* Проверять, что услуга httpd работает. Если она не работает, то запустить её с параметром start.



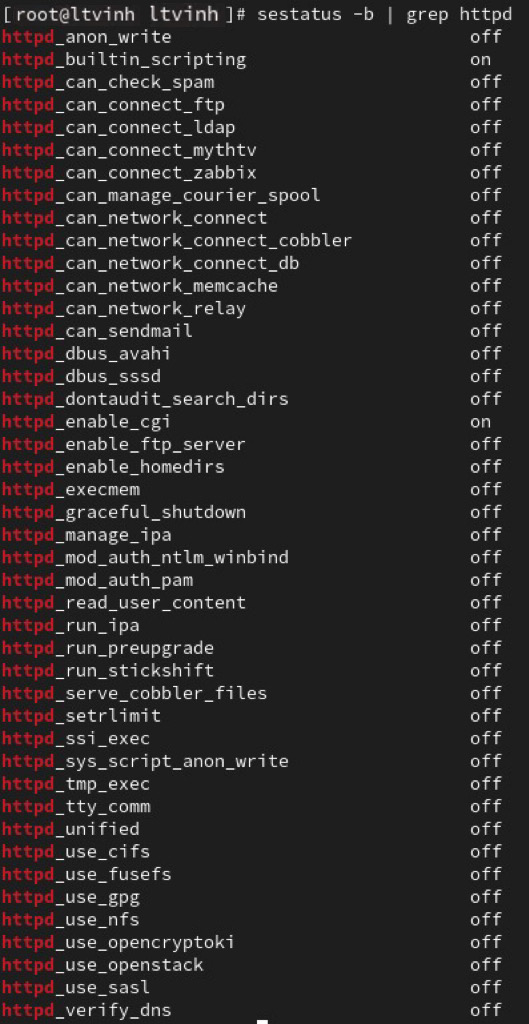


* Использовать команду ps auxZ | grep httpd, найти веб-сервер Apache в списке процессов. В нем находится контекст безопасности “system\_u:system\_r:httpd\_t:s0”, где:

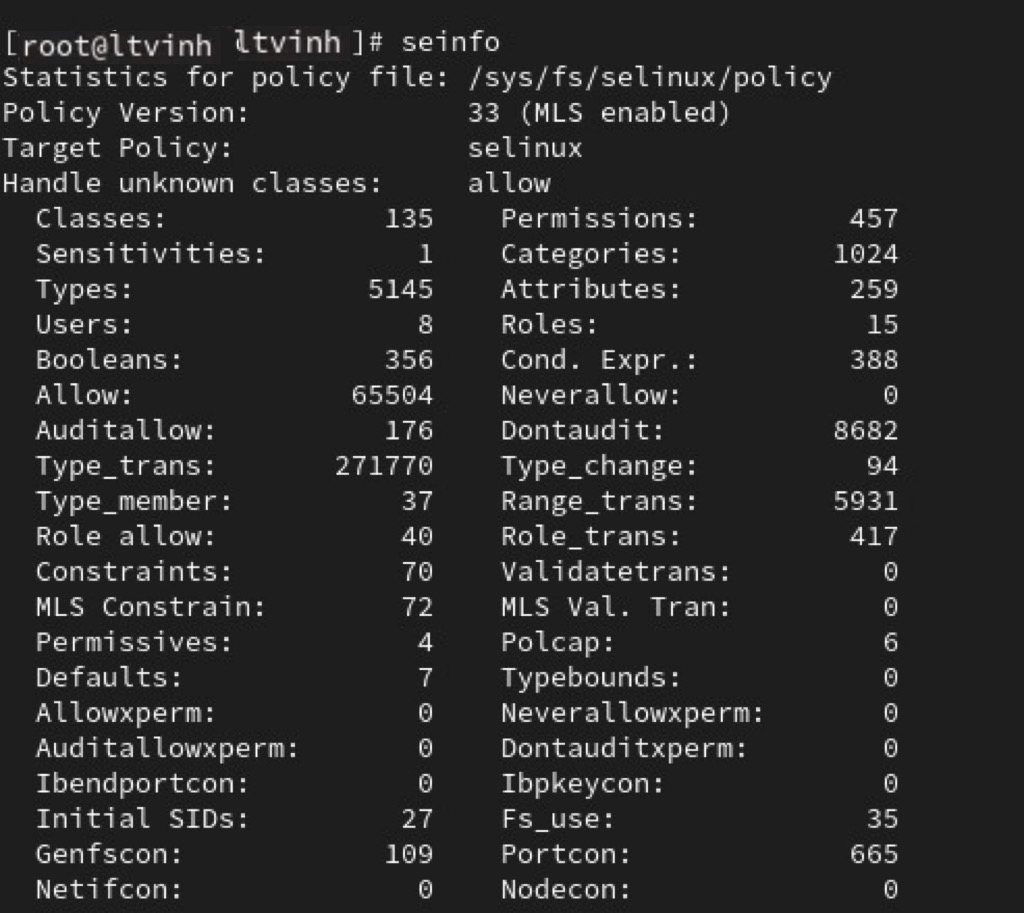
1. system\_u — это системный пользователь, который обычно используется для системных служб, управляемых SELinux.
2. system\_r — это системная роль, которая позволяет процессам выполнять различные задачи на системном уровне.
3. httpd\_t — это тип, используемый Apache HTTP Server (httpd).
4. s0 — это уровень безопасности по умолчанию в SELinux, обычно связанный с несекретными данными.



* Посмотрить текущее состояние переключателей SELinux для Apache с помощью команды.



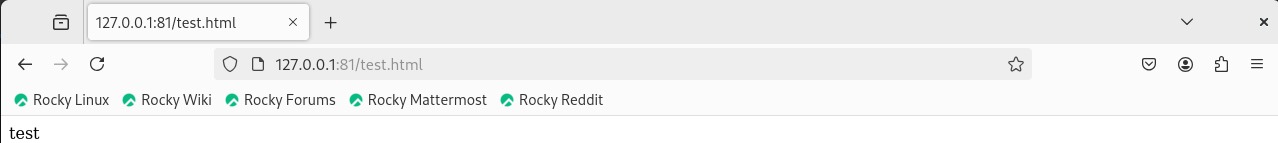
* Посмотрить статистику по политике с помощью команды seinfo, результат даёт количество пользователей, типов, ролей и т.д.



* Создать от имени суперпользователя html-файл /var/www/html/test.html следующего содержания.

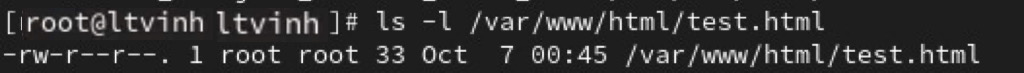


* Обратиться к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html.

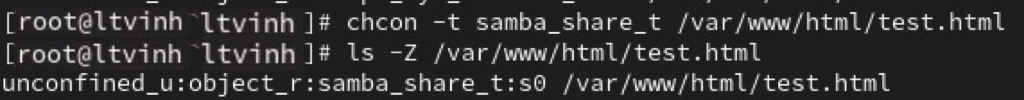


* Проверить контекст файла test.html можно командой ls -Z /var/www/html/test.html.

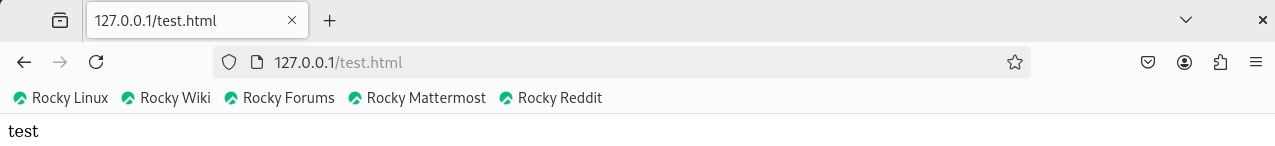
1. Поскольку по умолчанию пользователи не ограничены, созданный нами файл test.html был сопоставлен с SELinux, пользователем unconfined\_u
2. Роль object\_r используется по умолчанию для файлов на «постоянных» носителях и в сетевых файловых системах
3. Тип httpd\_sys\_content\_t позволяет процессу httpd получать доступ к файлустипа, с ним мы получили доступ к файлу при доступе к нему через браузер



* Изменить контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на samba\_share\_t, которого процесс httpd не иметь доступа.

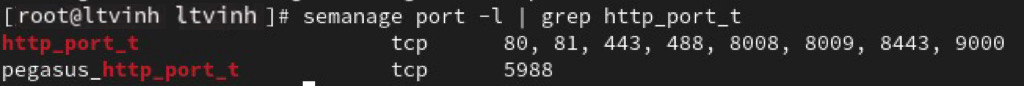


* Попробовать ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html.

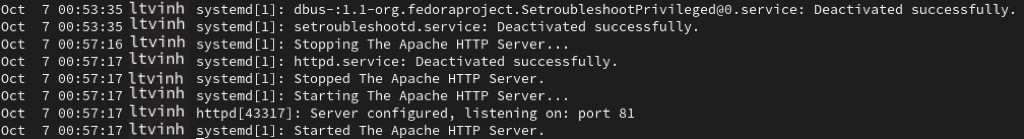


* Попробовать запустить веб-сервер Apache на прослушивание ТСР-порта 81 и убедиться, что порт 81 появился в списке.





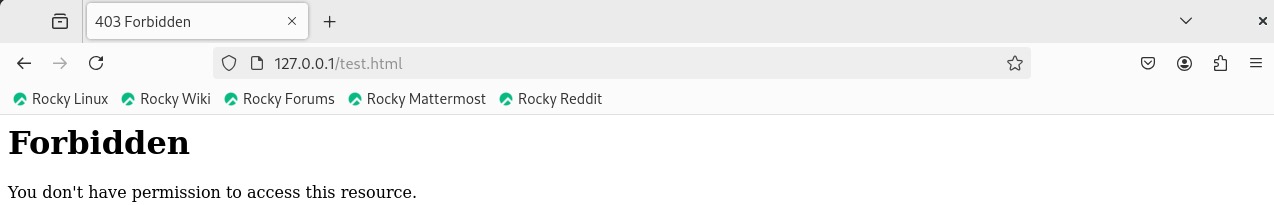
* Выполнять перезапуск веб-сервера Apache и проанализировать лог-файлы.



* Вернуть контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html.



* Попробовать получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html.



* Удалить файл /var/www/html/test.html.

# III. Вывод

После работы я получил практическое знакомство с технологией SELinux и развил навыки работы с ним.