Лабораторная работа №6

Мандатное разграничение прав в Linux

Щербак Маргарита Романовна, НПИбд-02-21

2024

Содержание

# Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux. Проверить работу SELinux на практике совместно с веб-сервером Apache [1].

# Теоретическое введение

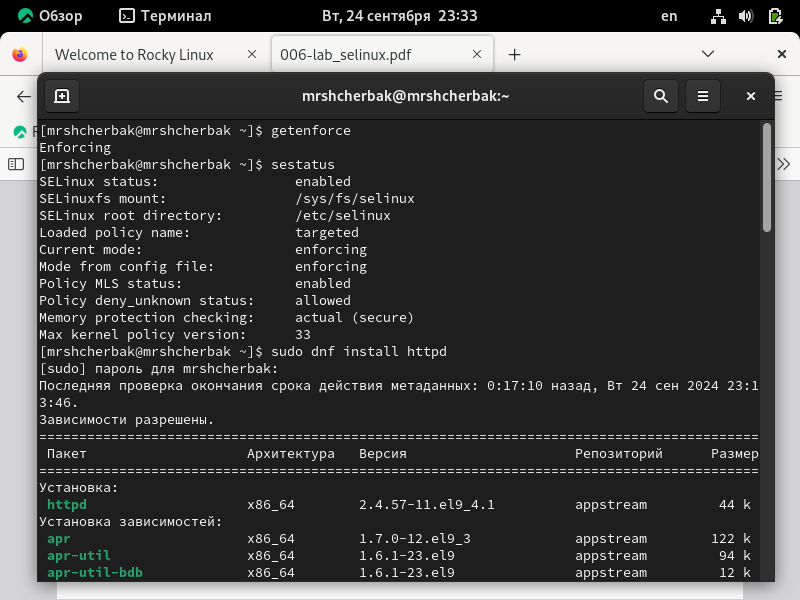
Информационная безопасность представляет собой защиту данных и поддерживающей инфраструктуры от случайных или преднамеренных воздействий природного или искусственного характера, которые могут нанести ущерб владельцам или пользователям этой информации и инфраструктуры [1].

Мандатное разграничение прав (Mandatory Access Control, MAC) — это метод управления доступом, где права пользователей и процессов определяются на системном уровне и не могут быть изменены без административного вмешательства. В отличие от дискреционного контроля доступа (DAC), где пользователи могут самостоятельно устанавливать права на свои файлы, MAC жестко контролирует доступ к ресурсам на основании политик безопасности [2].

SELinux (Security-Enhanced Linux) — это система MAC в Linux, которая реализует разграничение прав через политики безопасности [3].

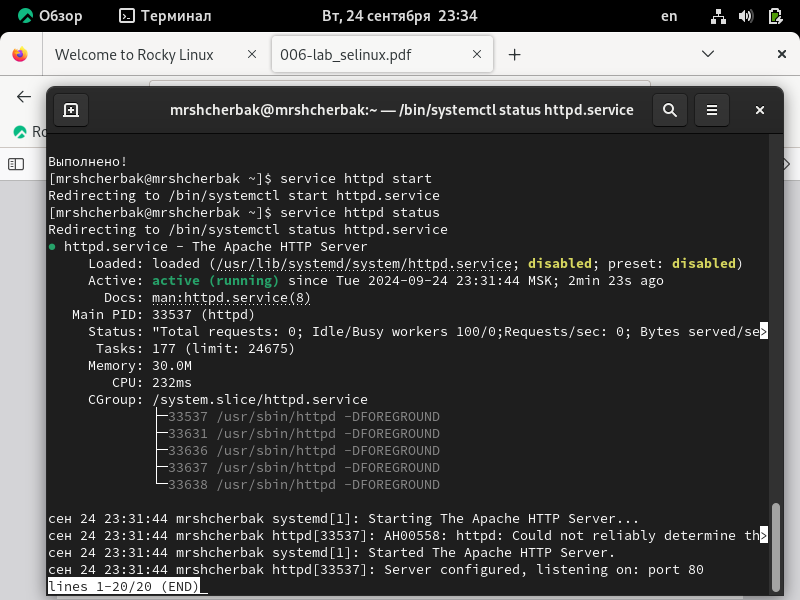
# Выполнение лабораторной работы

1. Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus (рис.1).



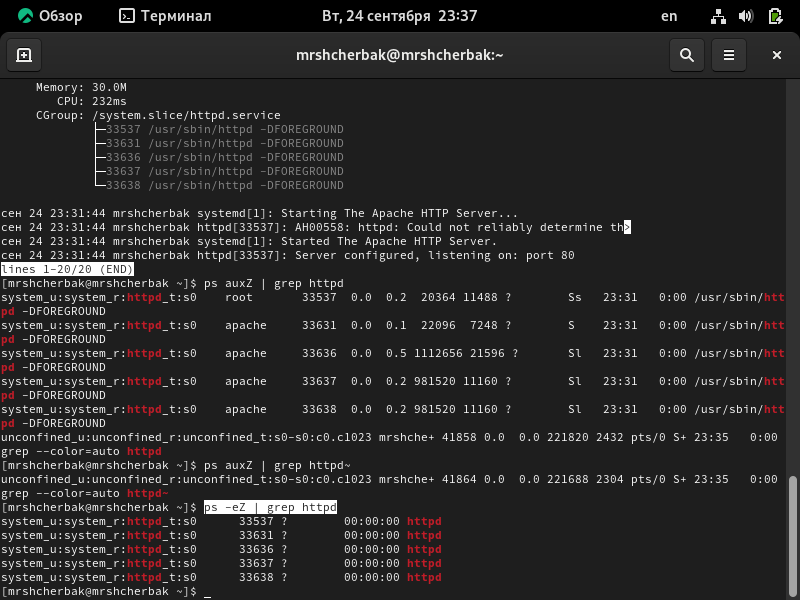
getenforce и sestatus

1. Обратитилась с помощью браузера к веб-серверу, запущенному на компьютере, и убедилась, что последний работает (рис.2).



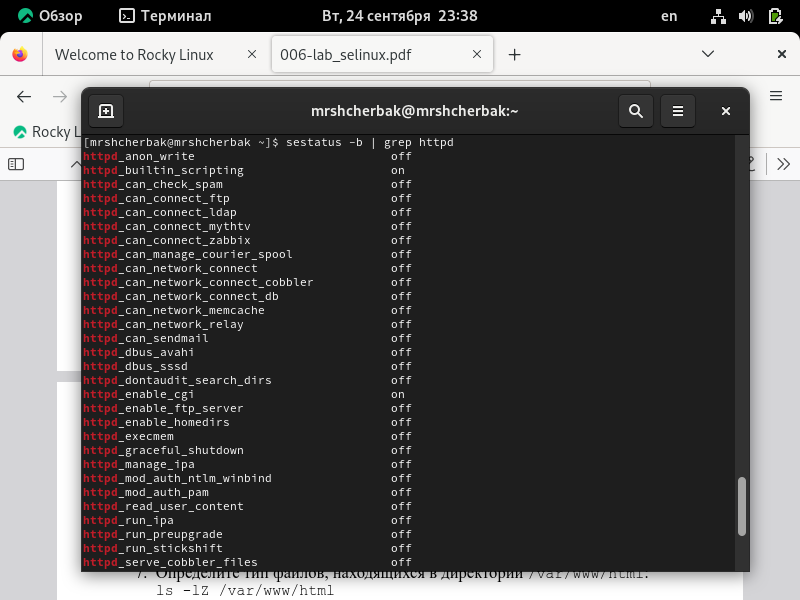
Веб-сервер httpd

1. Нашла веб-сервер Apache в списке процессов. Его контекст безопасности httpd\_sys\_cоntent\_t (рис.3).



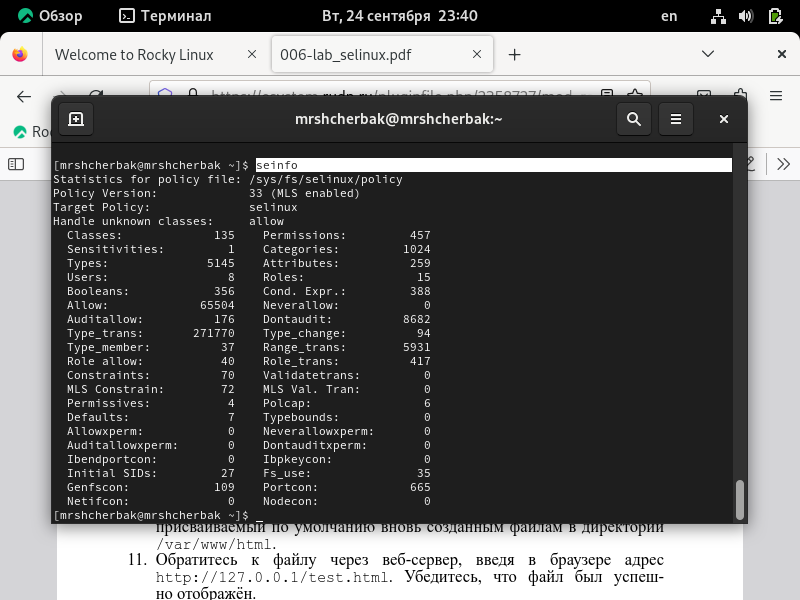
Веб-сервер Apache

1. Посмотрела текущее состояние переключателей SELinux для Apache (рис.4).



Cостояние переключателей

1. Посмотрела статистику по политике с помощью команды seinfo, также определила множество пользователей, ролей, типов (рис.5).



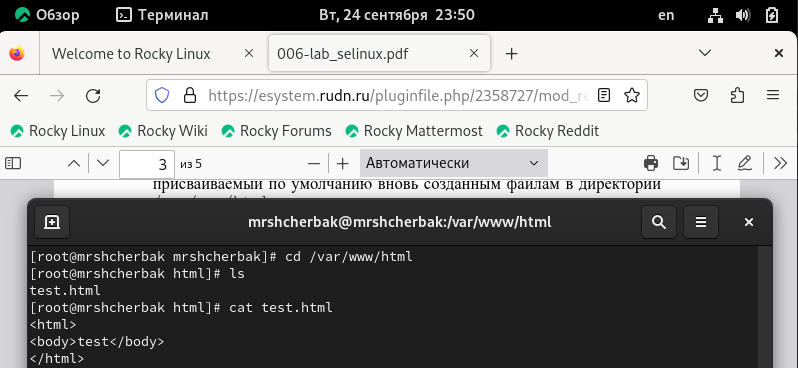
Статистика по политике с помощью команды seinfo

1. Определила тип файлов и поддиректорий, находящихся в директории /var/www. Определила тип файлов, находящихся в директории /var/www/html. Определила круг пользователей, которым разрешено создание файлов в директории /var/www/html - только владелец (рис.6).



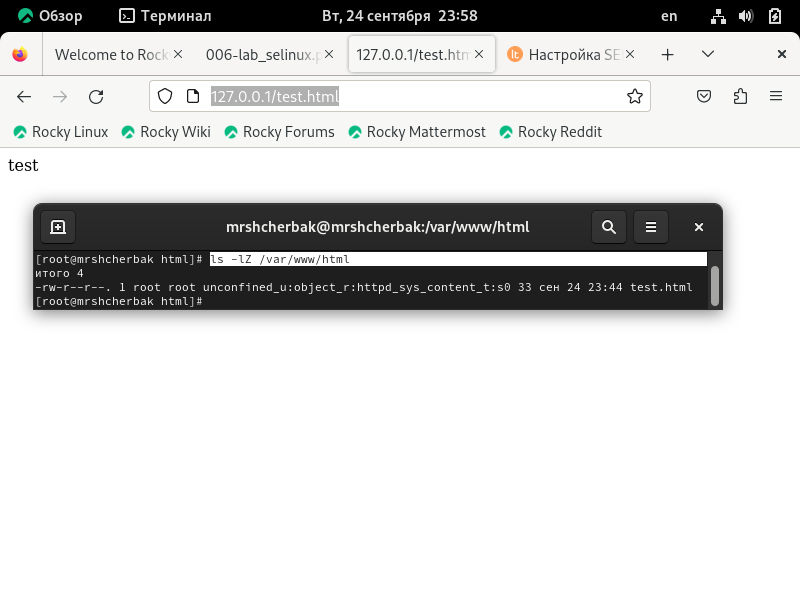
Просмотр типов файлов и поддиректорий

1. Создала от имени суперпользователя html-файл /var/www/html/test.html следующего содержания. А также проверила контекст созданного файла - httpd\_sys\_cоntent\_t (рис.7).



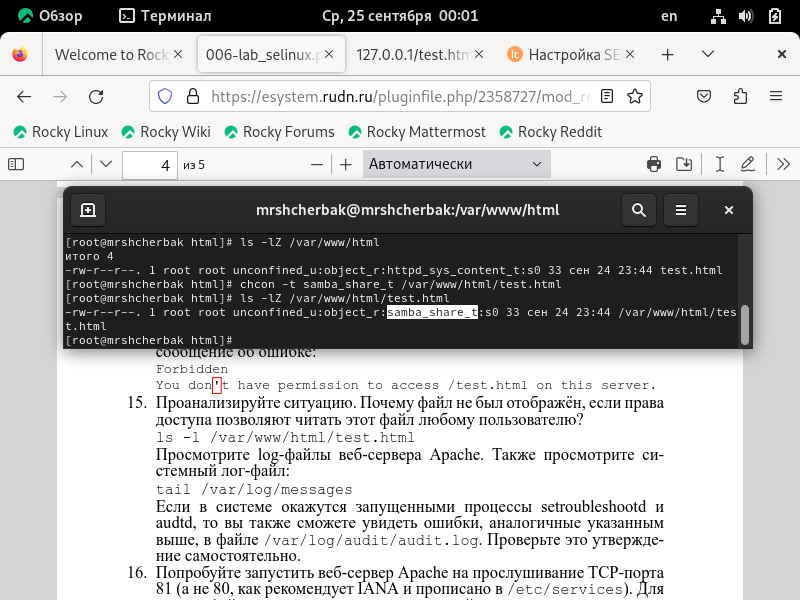
Содержимое файла

1. Проверила контекст созданного файла. Обратилась к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Убедилась, что файл был успешно отображён (рис.8). Файл был успешно отображён.



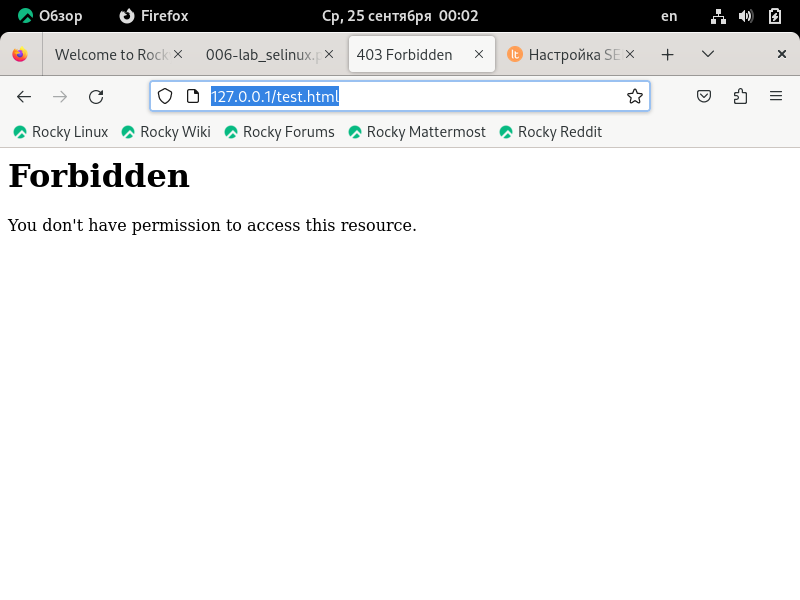
Контекст и отображение файла

1. Проверила контекст файла /var/www/html/test.html командой ls -Z. Изменила контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на samba\_share\_t (рис.9).



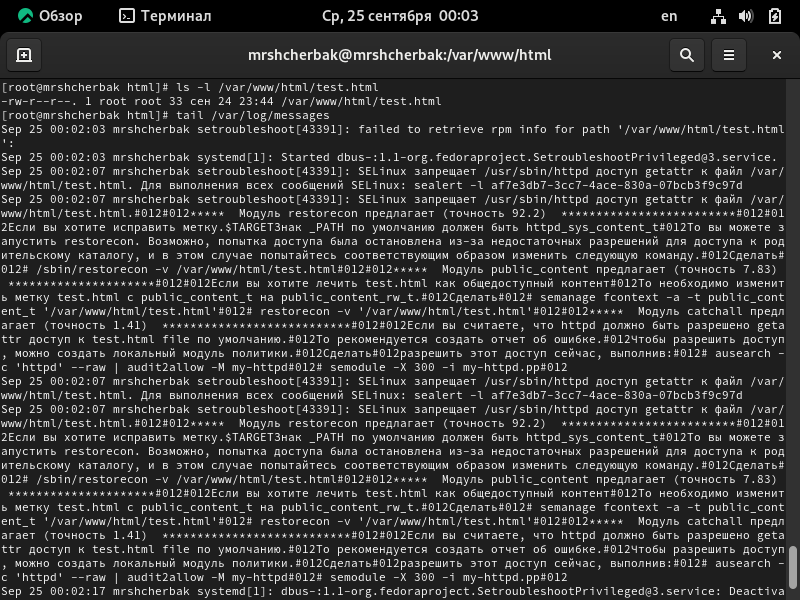
Проверка контекста и его изменение

1. Попробовала ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Получила сообщение об ошибке (рис.10).

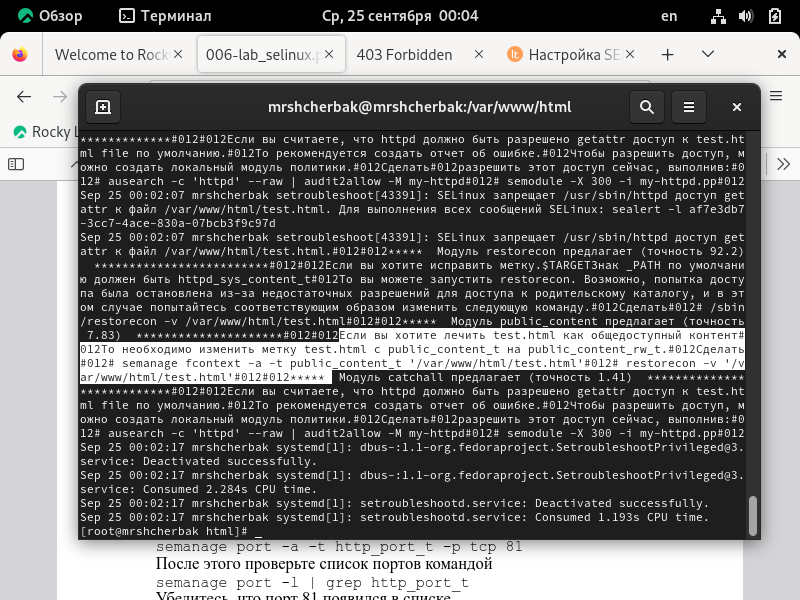


Ошибка

1. Просмотрела log-файлы веб-сервера Apache. Также просмотрела системный лог-файл: tail /var/log/messages (рис.11 - рис.12).

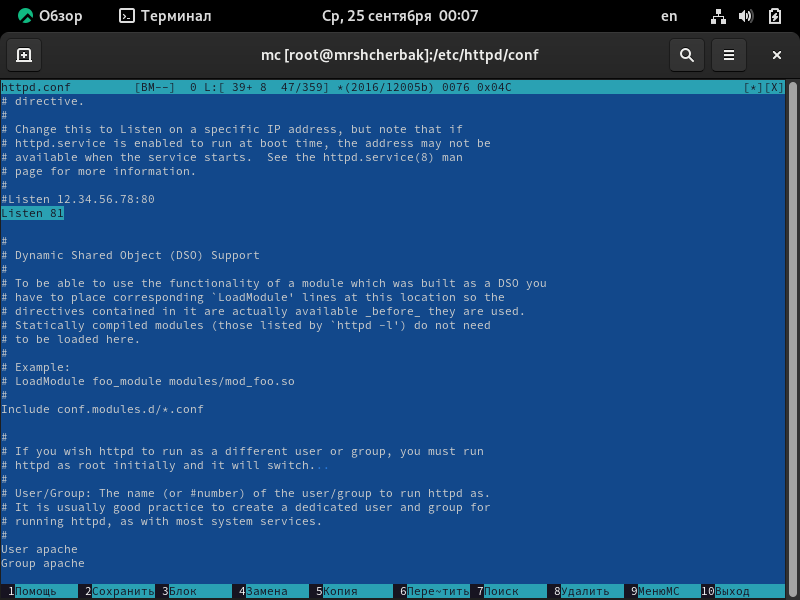


log-файлы



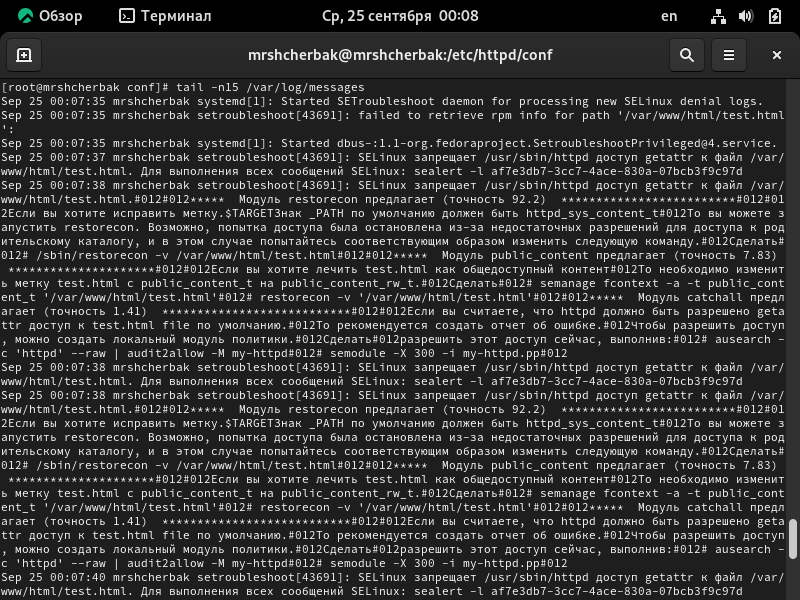
log-файлы

1. Попробовала запустить веб-сервер Apache на прослушивание ТСР-порта 81. Для этого в файле /etc/httpd/httpd.conf нашла строчку Listen 80 и заменила её на Listen 81 (рис.13).

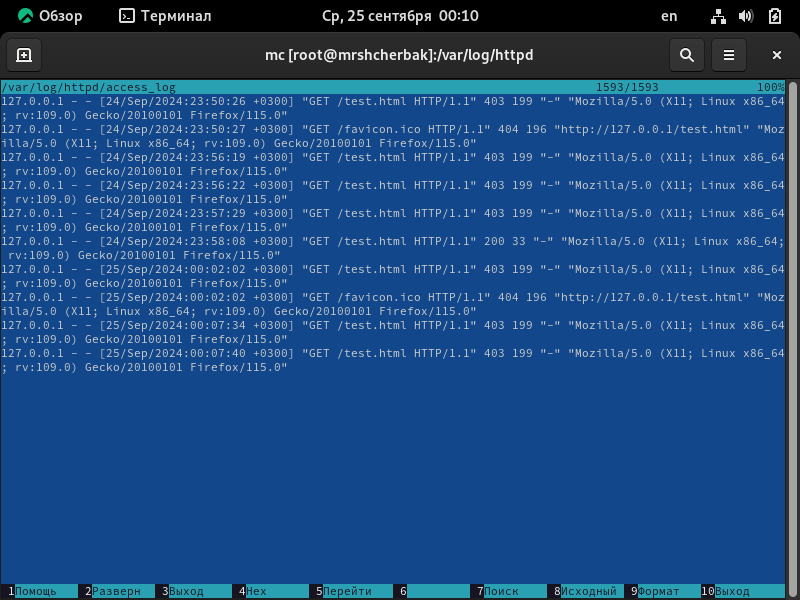


Замена порта в файле

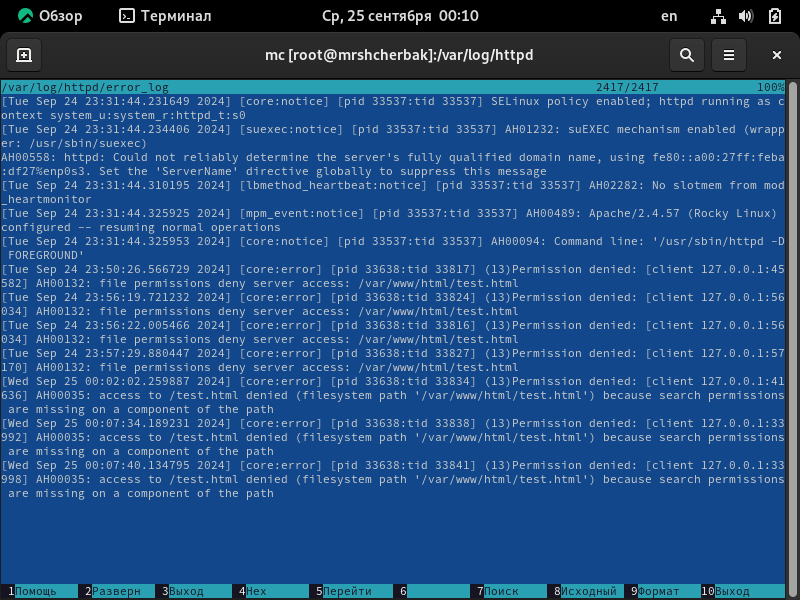
1. Выполните перезапуск веб-сервера Apache. Сбоя нет.Проанализировала лог-файлы: tail -nl /var/log/messages (рис.14). Просмотрела файлы /var/log/http/error\_log, /var/log/http/access\_log и /var/log/audit/audit.log (рис.15 - рис.16).



/var/log/messages

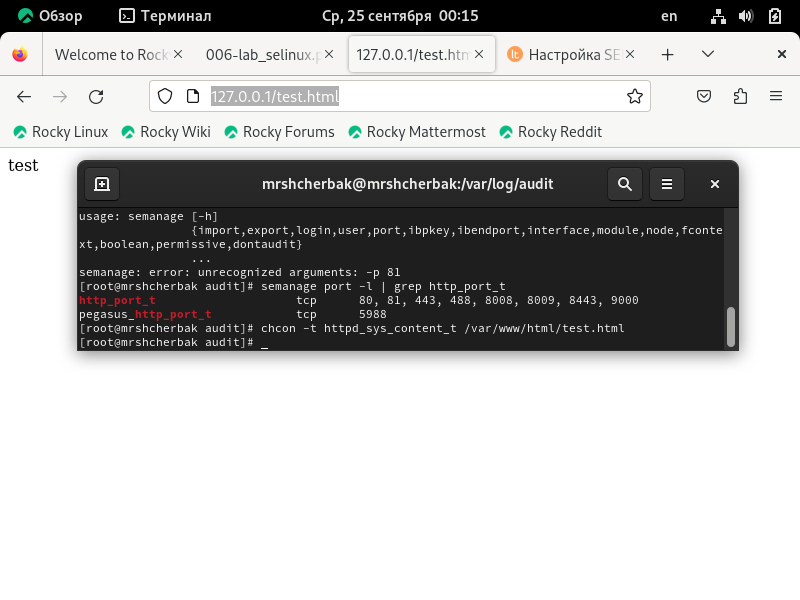


/var/log/http/access\_log



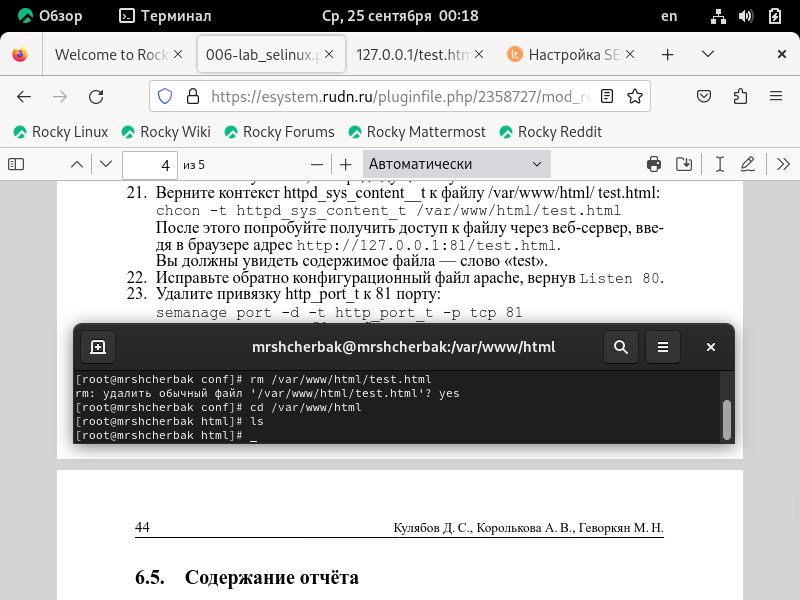
/var/log/http/error\_log

1. Выполнила команду semanage port -a -t http\_port\_t -р tcp 81. После этого проверила список портов командой semanage port -l | grep http\_port\_t. Убедилась, что порт 81 появился в списке. Попробовала запустить веб-сервер Apache ещё раз. Порт 81 был в списке до этого, поэтому сбоя не было. Вернула 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 попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. Увидела содержимое файла — слово «test» (рис.17).



Semanage и запуск веб-сервера

1. Исправила обратно конфигурационный файл apache, вернув Listen 80.Удалила привязку http\_port\_t к 81 порту: semanage port -d -t http\_port\_t -p tcp 81. Удалила файл /var/www/html/test.html (рис.18).



Удаление файла

# Вывод

Таким образом, в ходе ЛР№6 я развила навыки администрирования ОС Linux. Получила первое практическое знакомство с технологией SELinux. Проверила работу SELinux на практике совместно с веб-сервером Apache.

# Библиография

1. Методические материалы курса.
2. Linux Kernel Security [Электронный ресурс]: URL: http://www.cyberciti.biz/tips/selinux-vs-apparmor-vs-grsecurity.html (дата обращения 24.09.2024)
3. Security-Enhanced Linux [Электронный ресурс]: Официальный сайт SELinux. URL: http://www.nsa.gov/research/selinux/index.shtml (дата обращения 24.09.2024)