Отчет по лабораторной работе 5

Мандатное разграничение прав в Linux

Шалыгин Георгий Эдуардович

Содержание

# 1 Цель работы

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

# 2 Теоретическое введение

В Linux, как и в любой многопользовательской системе, абсолютно естественным образом возникает задача разграничения доступа субъектов — пользователей к объектам — файлам дерева каталогов.

Один из подходов к разграничению доступа — так называемый дискреционный (от англ, discretion — чье-либо усмотрение) — предполагает назначение владельцев объектов, которые по собственному усмотрению определяют права доступа субъектов (других пользователей) к объектам (файлам), которыми владеют.

Дискреционные механизмы разграничения доступа используются для разграничения прав доступа процессов как обычных пользователей, так и для ограничения прав системных программ в (например, служб операционной системы), которые работают от лица псевдопользовательских учетных записей.

В Linux у каждого файла и каждого каталога есть два владельца: пользователь и группа.

Эти владельцы устанавливаются при создании файла или каталога. Пользователь, который создаёт файл становится владельцем этого файла, а первичная группа, в которую входит этот же пользователь, так же становится владельцем этого файла. Чтобы определить, есть ли у вас как у пользователя права доступа к файлу или каталогу, оболочка проверяет владение ими.

**Это происходит в следующем порядке:**

1. Оболочка проверяет, являетесь ли вы владельцем файла, к которому вы хотите получить доступ. Если вы являетесь этим владельцем, вы получаете разрешения и оболочка прекращает проверку.
2. Если вы не являетесь владельцем файла, оболочка проверит, являетесь ли вы участником группы, у которой есть разрешения на этот файл. Если вы являетесь участником этой группы, вы получаете доступ к файлу с разрешениями, которые для группы установлены, и оболочка прекратит проверку.
3. Если вы не являетесь ни пользователем, ни владельцем группы, вы получаете права других пользователей (Other).

Чтобы увидеть текущие назначения владельца, вы можете использовать команду **ls -l**. Эта команда показывает пользователя и группу-владельца.

Подробнее в [1].

### 2.0.1 Изменение владельца

Чтобы применить соответствующие разрешения, первое, что нужно учитывать, это владение. Для этого есть команда **chown**. Синтаксис этой команды несложен для понимания:

chown кто что

Например, следующая команда меняет владельца каталога /home/account на пользователя linda:

chown linda /home/account

## 2.1 Использование chmod

Для управления правами используется команда **chmod**. При использовании **chmod** вы можете устанавливать разрешения для пользователя (user), группы (group) и других (other). Вы можете использовать эту команду в двух режимах: относительный режим и абсолютный режим. В абсолютном режиме три цифры используются для установки основных разрешений.

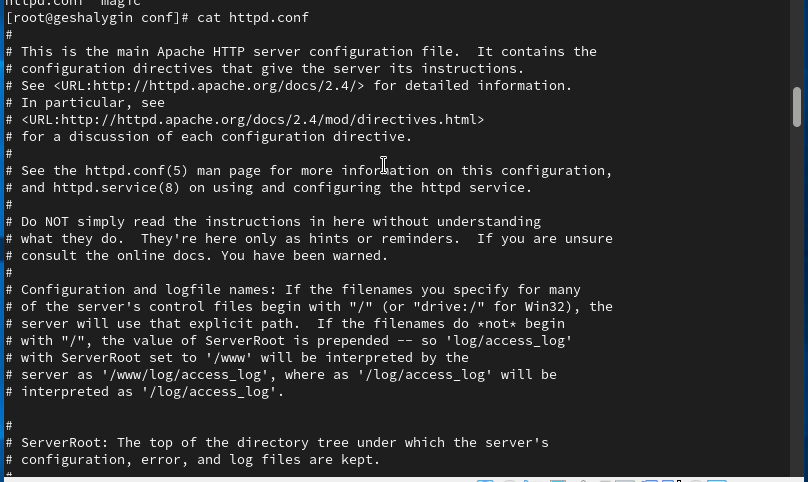
При настройке разрешений рассчитайте необходимое вам значение. Если вы хотите установить чтение, запись и выполнение для пользователя, чтение и выполнение для группы, а также чтение и выполнение для других в файле /somefile, то вы используете следующую команду **chmod**:

chmod 755 /somefile

Подробнее в [2].

# 3 Выполнение лабораторной работы

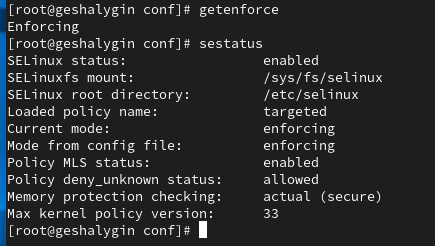
1. Файл /etc/httpd/httpd.conf (fig. 1).

* 
* Рис. 1: Файл httpd.conf

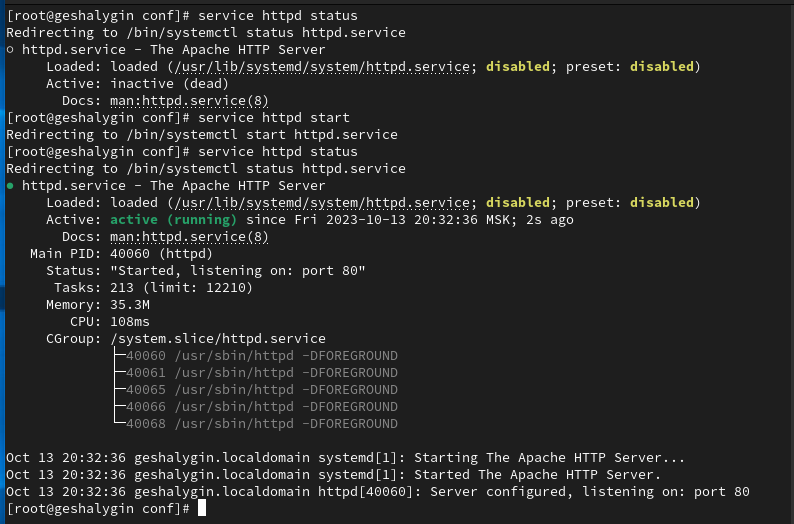
1. Также необходимо проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp. Отключить фильтр можно командами (fig. 2)

* Задание имени сервера
* Рис. 2: Задание имени сервера

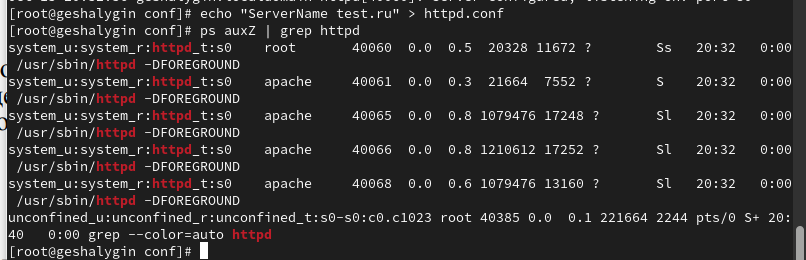
1. Убедимся, что SELinux работает в режиме enforcing политики targeted с помощью команд getenforce и sestatus.(fig. 3).

* 
* Рис. 3: Проверка работы

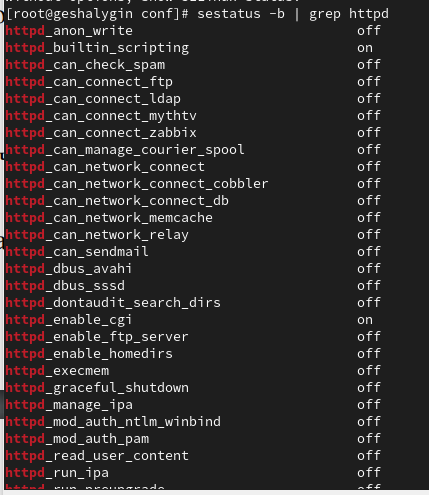
1. Запустим веб-сервер: service httpd start (fig. 4).

* 
* Рис. 4: Запуск веб-сервера

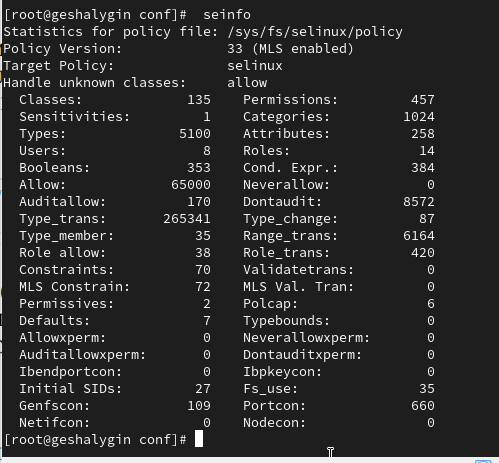
1. Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности. Например, можно использовать команду ps auxZ | grep httpd (fig. 5).

* 
* Рис. 5: Контекст сервера

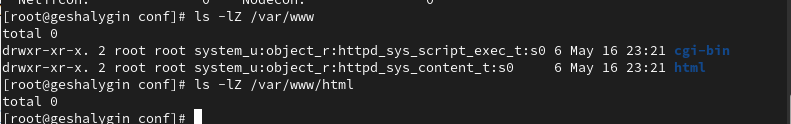
1. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды sestatus -bigrep httpd. Многие отключены (fig. 6).

* 
* Рис. 6: Состояние переключателей

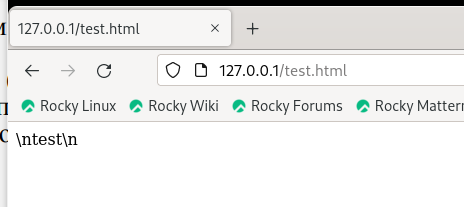
1. Посмотрим статистику по политике с помощью команды seinfo (fig. 7).

* 
* Рис. 7: Статистика по политике

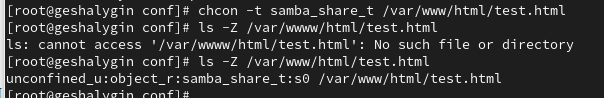
1. Определим тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды ls -lZ /var/www. (fig. 8). Проверим что guest не имеет доступ к файлу.

* 
* Рис. 8: Проверка контекста

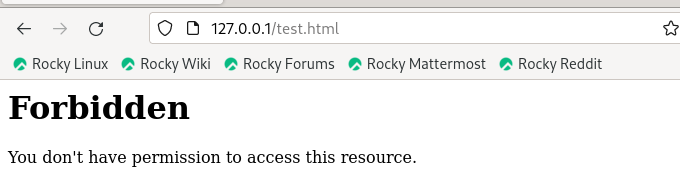
1. Создадим от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html

* Обратиvcz к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Убедиvcz, что файл был успешно отображён. (fig. 9).
* 
* Рис. 9: Доступ к серверу

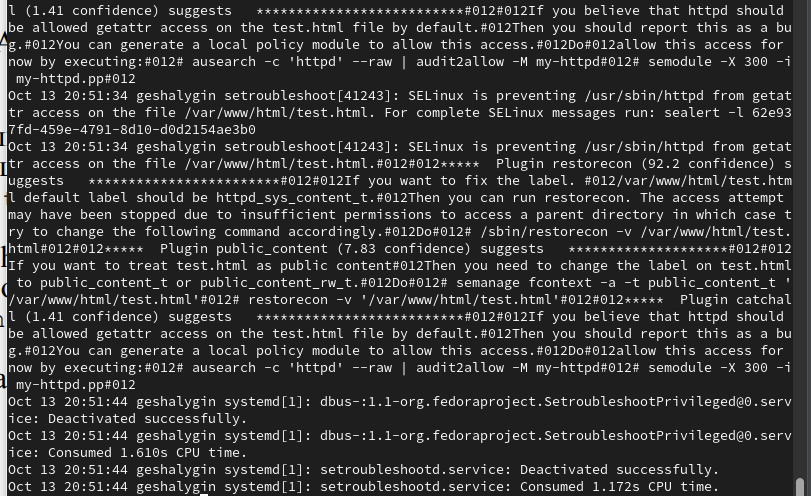
1. Измениv контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на samba\_share\_t: chcon -t samba\_share\_t /var/www/html/test.html ls -Z /var/www/html/test.html (fig. 10).

* 
* Рис. 10: Изменение контекста

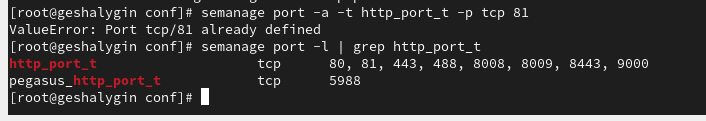
1. После этого файл недоступен (fig. 11).

* 
* Рис. 11: Доступа нет

1. Просмотрим системный лог-файл: tail /var/log/messages (fig. 12).

* 
* Рис. 12: Лог файл

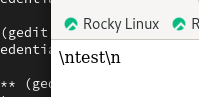
1. Выполним команду semanage port -a -t http\_port\_t -р tcp 81. После этого проверим список портов командой semanage port -l | grep http\_port\_t. Убедимся, что порт 81 появился в списке. Теперь доступ к серверу есть, мы добавили порт 81. (fig. 13).

* 
* Рис. 13: Добавление порта

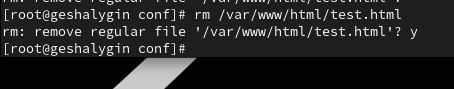
1. Вернем контекст httpd\_sys\_cоntent\_\_t к файлу /var/www/html/ test.html: chcon -t httpd\_sys\_content\_t /var/www/html/test.html После этого попробуем получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. (fig. 14).

* Возвращение контекста
* Рис. 14: Возвращение контекста

1. Сервер снова доступен (fig. 15).

* 
* Рис. 15: Сервер доступен

1. Исправим конфигурацию, удалим привязку к 81 порту и файл test (fig. 16).

* 
* Рис. 16: Окончание работы

# 4 Выводы

В результате выполнения работы мы развили навыки администрирования ОС Linux. Получили первое практическое знакомство с технологией SELinux1 . Проверили работу SELinx на практике совместно с веб-сервером Apache.

# Список литературы

1. Кетов Д.В. Внутреннее устройство Linux. BHV, 2017. 124 с.

2. Л. М. Ухлинов. Управление доступом в ОС GNU /Linux . ОКБ САПР», Москва, Россия, 2010.