Лабораторная работа 6

Попов Дмитрий Павлович, НФИбд-01-19

Содержание

List of Figures

List of Tables

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

дисциплина: Информационная безопасность

Преподователь: Кулябов Дмитрий Сергеевич

Студент: Попов Дмитрий Павлович

Группа: НФИбд-01-19

МОСКВА

2022 г.

1 Цель работы

Целью данной лабораторной работы является развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1. Проверить работу SELinx на практике совместно с веб-сервером Apache.

2 Подготовка лабораторного стенда и методические рекомендации

1. Установили веб-сервер Apache.
2. В конфигурационном файле /etc/httpd/httpd.conf задали параметр ServerName.
3. Отключили пакетный фильтр.

3 Выполнение лабораторной работы

1. Входим в систему с полученными учётными данными. Проверили, что SELinux работает в режиме enforcing политики targeted с помощью команд **getenforce** и **sestatus**. (fig. 3.1)

```
[dpopov@dpopov ~]$ getenforce
Enforcing
[dpopov@dpopov ~]$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33
```

Figure 3.1: Выполнение команд getenforce и sestatus

2. Запустили веб-сервер и обратились к нему с помощью команды (fig. 3.2): service httpd status

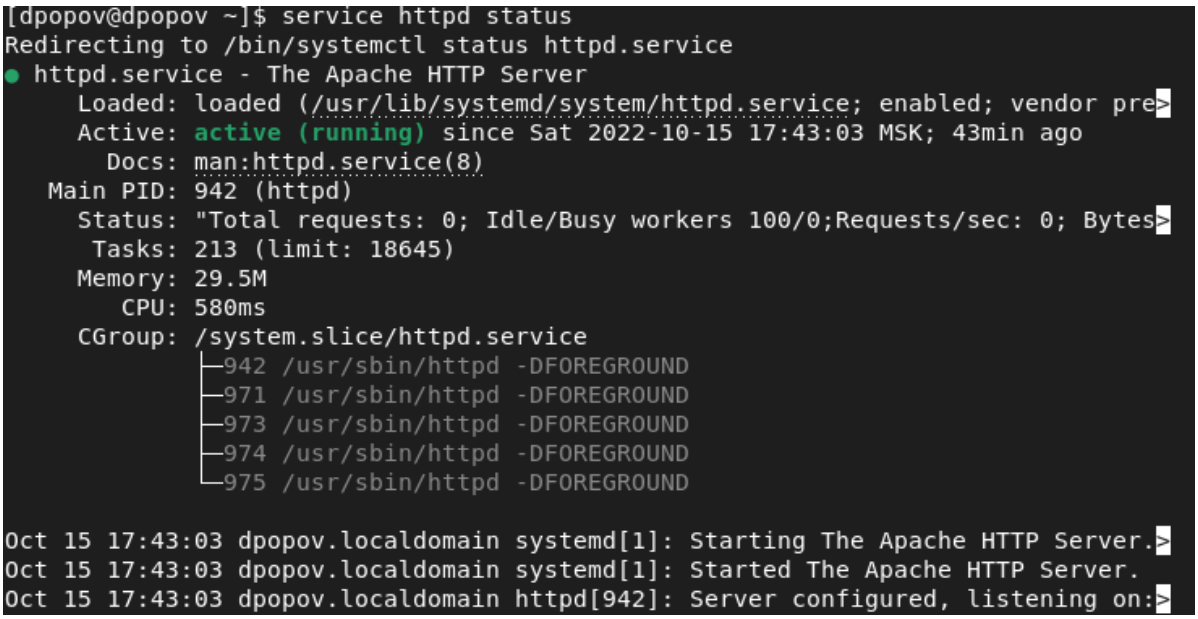

Figure 3.2: Выполнение команды service httpd status

3. Нашли веб-сервер Apache в списке процессов. Контекст безопасности - unconfined_u:unconfined_r:unconfined_t. (fig. 3.3)

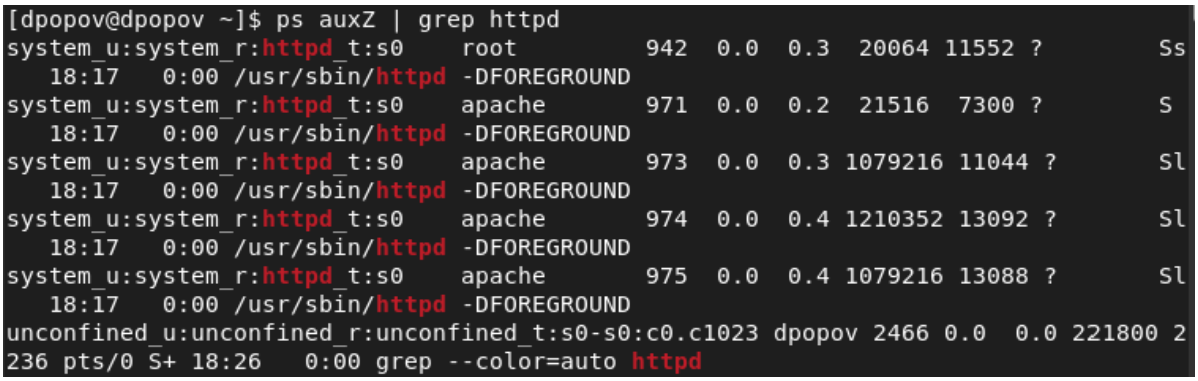

Figure 3.3: Выполнение команды ps auxZ | grep httpd

4. Посмотрели текущее состояние переключателей SELinux для Apache с помощью команды **sestatus -b | grep httpd**. (fig. 3.4)

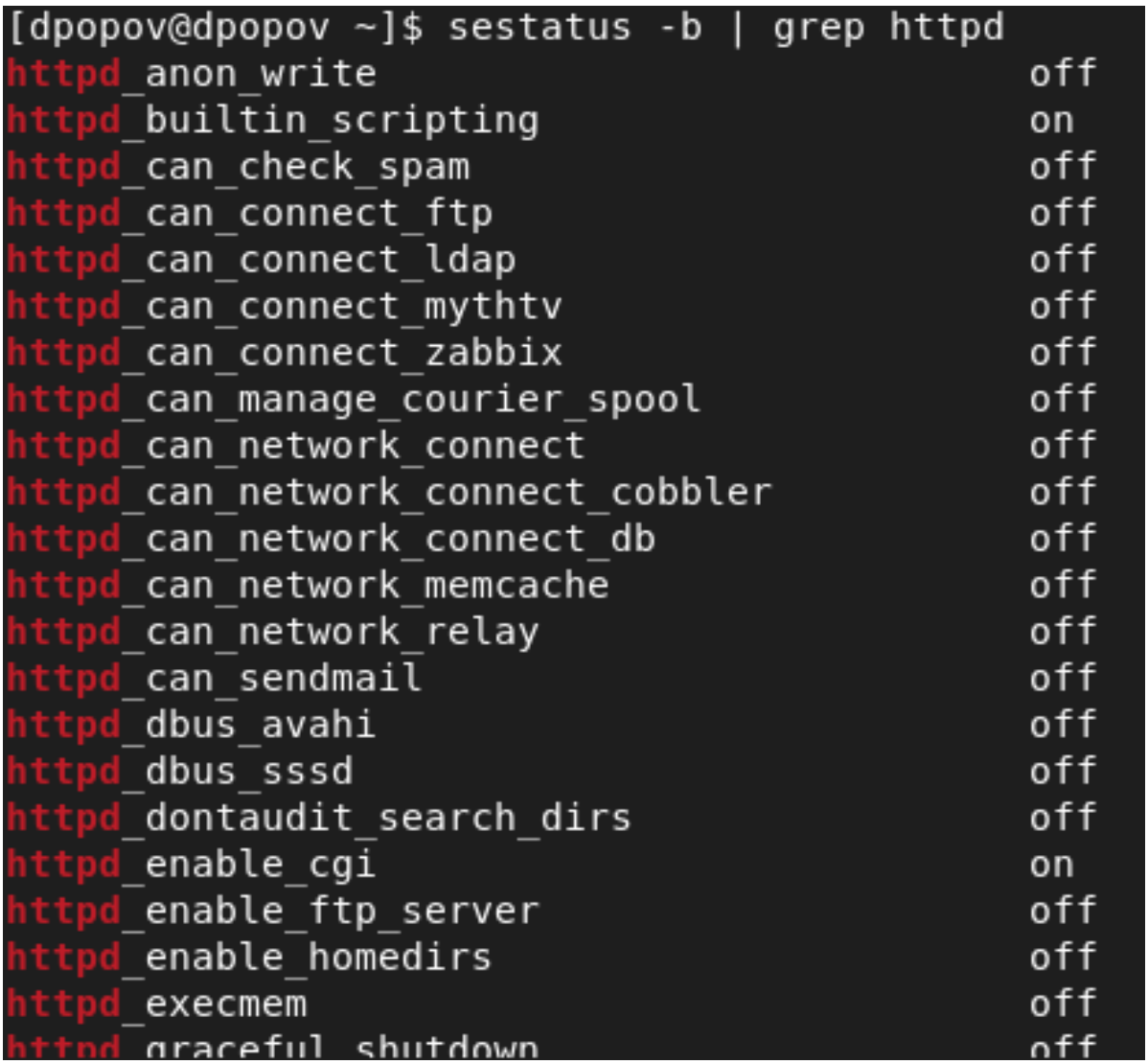

Figure 3.4: Выполнение команды sestatus -b | grep httpd

5. Посмотрели статистику по политике с помощью команды **seinfo**. Определили, что множество пользователей = 8; ролей = 14; типов = 5002. (fig. 3.5)

```
[dpopov@dpopov ~]$ seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:             33 (MLS enabled)
Target Policy:              selinux
Handle unknown classes:     allow
  Classes:              133    Permissions:          454
  Sensitivities:          1    Categories:          1024
  Types:               5002    Attributes:           254
  Users:                  8    Roles:                 14
  Booleans:             347    Cond. Expr.:          381
```

Figure 3.5: Статистика по политике

6. Определили тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды **ls -lZ /var/www**. (fig. 3.6)

```
[dpopov@dpopov ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 May 16 15:10
 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0     6 May 16 15:10
 html
```

Figure 3.6: Выполнение команды ls -lZ /var/www

7. Необходимо было определить тип файлов, находящихся в директории /var/www/html, с помощью команды **ls -lZ /var/www/html**. Но в данной директории файлов не обнаружилось. (fig. 3.7)

```
[dpopov@dpopov ~]$ ls -lZ /var/www/html
total 0
```

Figure 3.7: Выполнение команды ls -lZ /var/www/html

8. Определим круг пользователей, которым разрешено создание файлов в директории /var/www/html - только root. (fig. 3.8)

```
[dpopov@dpopov ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 May 16 15:10
 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0     6 May 16 15:10
 html
```

Figure 3.8: Выполнение команды ls -lZ /var/www

9. Создали от имени суперпользователя html-файл /var/www/html/test.html следующего содержания: (fig. 3.9)

Figure 3.9: Содержимое файла test.html

10. Проверили контекст созданного файла - httpd_sys_content_t. (fig. 3.10)

```
[dpopov@dpopov ~]$ ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 15 18:3
0 /var/www/html/test.html
```

Figure 3.10: Контекст файла test.html

11. Обратитились к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html и убедились, что файл был успешно отображён. (fig. 3.11)

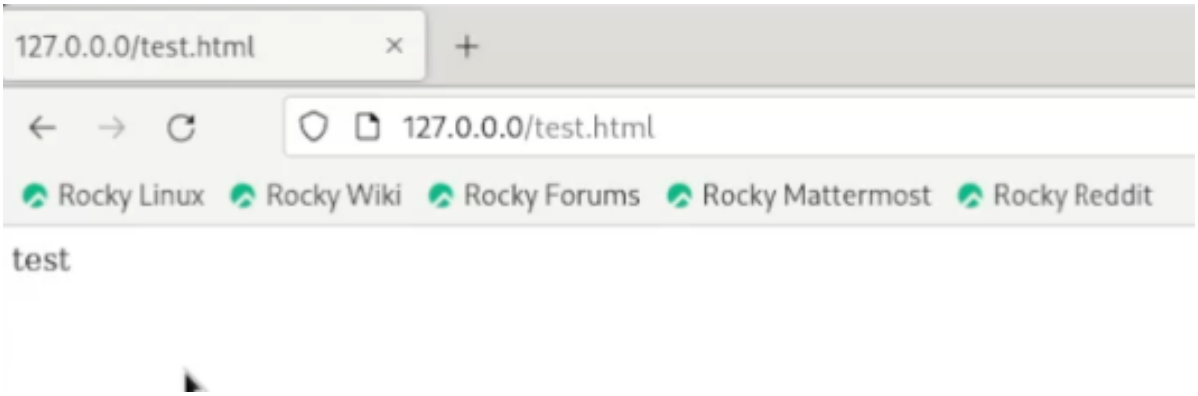

Figure 3.11: Обращение к файлу test.html через веб-сервер

12. Изучили справку man httpd_selinux. Тип файла test.html - контекст созданного файла - httpd_sys_content_t. (fig. 3.12)

```
[dpopov@dpopov ~]$ ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 15 18:3
0 /var/www/html/test.html
```

Figure 3.12: Контекст файла test.html

13. Изменили 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 И проверили, что контекст поменялся. (fig. 3.13)

```
[dpopov@dpopov ~]$ sudo chcon -t samba_share_t /var/www/html/test.html
[sudo] password for dpopov:
[dpopov@dpopov ~]$ ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Figure 3.13: Изменение контекста файла /var/www/html/test.html

14. Пробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. В результате получили ошибку. (fig. 3.14)

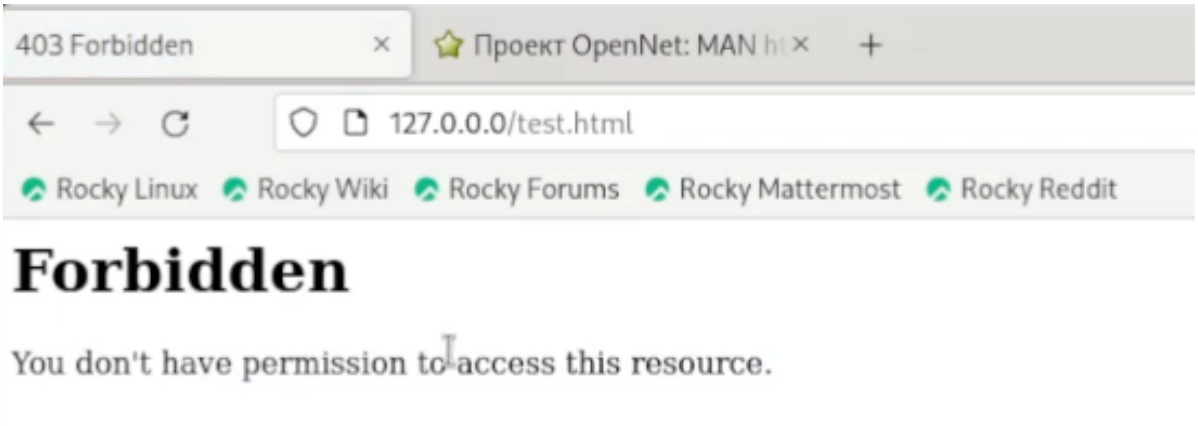

Figure 3.14: Обращение к файлу test.html через веб-сервер после изменения контекста

15. Проанализируем ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? ls -l /var/www/html/test.html Просмотрим log-файлы веб-сервера Apache и системный лог-файл: tail /var/log/messages В системе оказались запущенны процессы **setroubleshootd** и **audtd**. (fig. 3.15)

```
[dpopov@dpopov ~]$ ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct 15 18:30 /var/www/html/test.html
[dpopov@dpopov ~]$ sudo tail /var/log/messages
Oct 15 19:07:03 dpopov setroubleshoot[3553]: SELinux is preventing /usr/sbin/httpd
from getattr access on the file /var/www/html/test.html. For complete SELinux messa
ges run: sealert -l 0681191b-d658-4025-bb2a-21399effb9d7
Oct 15 19:07:03 dpopov setroubleshoot[3553]: SELinux is preventing /usr/sbin/httpd
from getattr access on the file /var/www/html/test.html.#012#012*****  Plugin resto
recon (92.2 confidence) suggests   ************************#012#012If you want to f
ix the label. #012/var/www/html/test.html default label should be httpd_sys_content
_t.#012Then you can run restorecon. The access attempt may have been stopped due to
 insufficient permissions to access a parent directory in which case try to change
the following command accordingly.#012Do#012# /sbin/restorecon -v /var/www/html/tes
t.html#012#012*****  Plugin public_content (7.83 confidence) suggests   ***********
*********#012#012If you want to treat test.html as public content#012Then you need
to change the label on test.html to public_content_t or public_content_rw_t.#012Do#
012# semanage fcontext -a -t public_content_t '/var/www/html/test.html'#012# restor
econ -v '/var/www/html/test.html'#012#012*****  Plugin catchall (1.41 confidence) s
uggests   ************************#012#012If you believe that httpd should be all
owed getattr access on the test.html file by default.#012Then you should report thi
s as a bug.#012You can generate a local policy module to allow this access.#012Do#0
12allow this access for now by executing:#012# ausearch -c 'httpd' --raw | audit2al
low -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
```

Figure 3.15: Вывод команд ls -l /var/www/html/test.html и tail /var/log/messages

16. Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81. Для этого в файле /etc/httpd/httpd.conf находим строчку Listen 80 и заменяем её на Listen 81. (fig. 3.16)

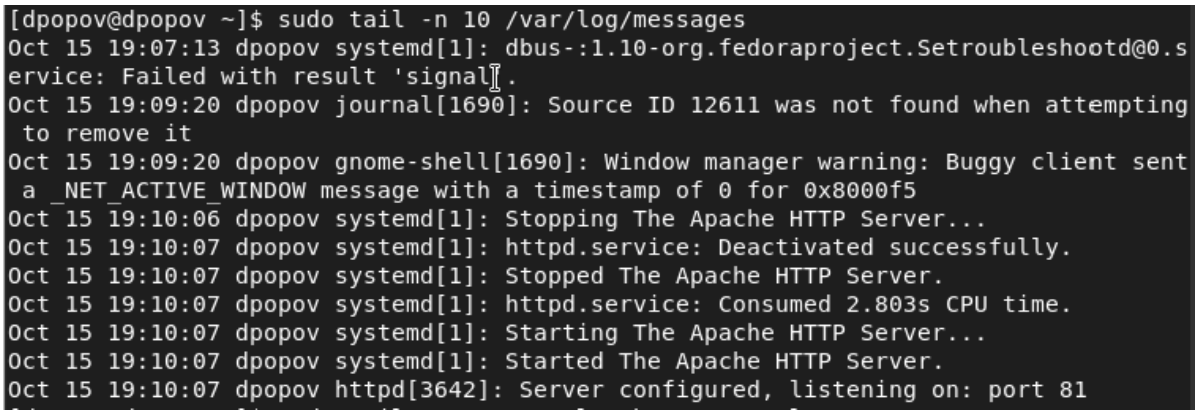

Figure 3.16: Запуск веб-сервера Apache на прослушивание TCP-порта 81

17. Выполним перезапуск веб-сервера Apache. Произошёл сбой? Нет.
18. Проанализируем лог-файлы: tail -nl /var/log/messages Просмотрим файлы /var/log/http/error_log, /var/log/http/access_log и /var/log/audit/audit.log. (fig. 3.17)

Figure 3.17: Перезапуск веб-сервера Apache

19. Выполним команду **semanage port -a -t http_port_t -p tcp 81**. Вылетает ValueError в связи с тем, что порт уже определен. После этого проверим список портов командой **semanage port -l | grep http_port_t** и убедились, что порт 81 появился в списке. (fig. 3.18)

Figure 3.18: Проверка установления 81 порта tcp

20. Попробуем запустить веб-сервер Apache ещё раз. (fig. 3.19)

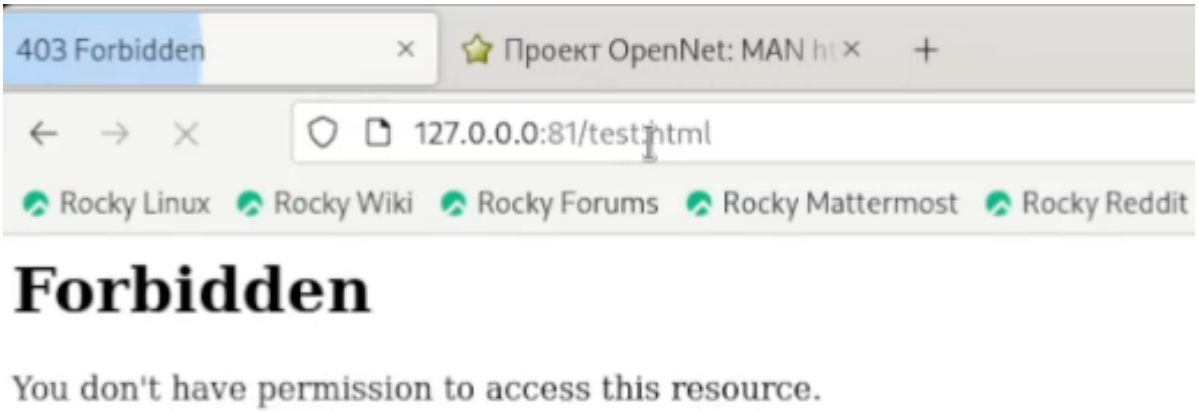

Figure 3.19: Перезапуск веб-сервера Apache

21. Вернули контекст httpd_sys_content_t к файлу /var/www/html/test.html: **chcon -t httpd_sys_content_t /var/www/html/test.html** (fig. 3.20)

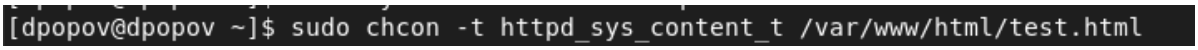

Figure 3.20: Возвращение контекста httpd_sys_content_t к файлу test.html

После этого пробуем получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. В результате увидели содержимое файла — слово «test». (fig. 3.21)

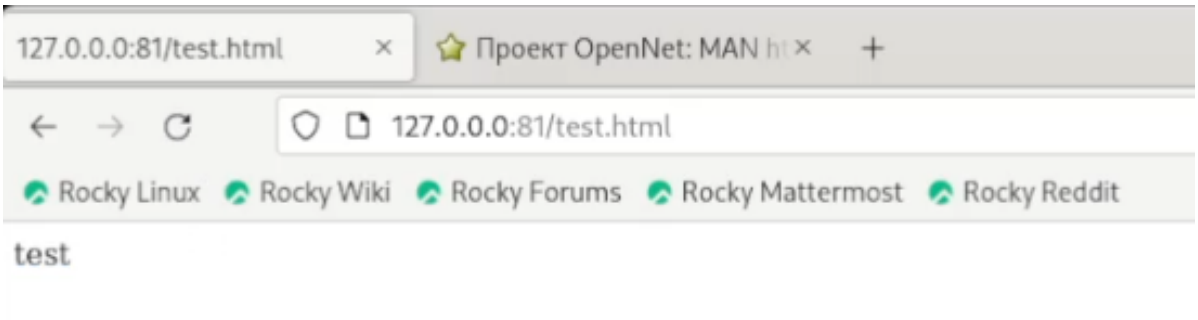

Figure 3.21: Обращение к файлу test.html через веб-сервер

22. Исправим обратно конфигурационный файл apache, вернув Listen 80. (fig. 3.22)

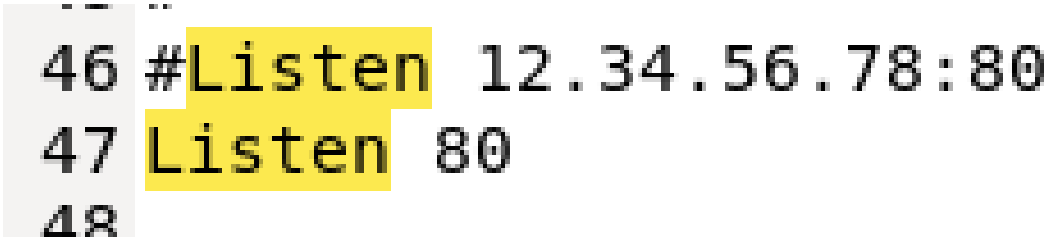

Figure 3.22: Исправление конфигурационного файла apache

23. Удалим привязку http_port_t к 81 порту: **semanage port -d -t http_port_t -p tcp 81** и проверим, что порт 81 удалён. Данная команда не была выполнена. (fig. 3.23)

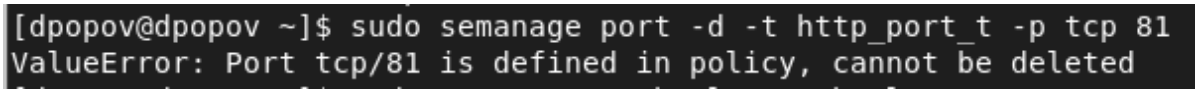

Figure 3.23: Удаление привязки http_port_t к 81 порту

24. Удалим файл /var/www/html/test.html: **rm /var/www/html/test.html**. (fig. 3.24)

Figure 3.24: Удаление файла test.html

4 Вывод

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

5 Библиография

1. Кулябов Д. С., Королькова А. В., Геворкян М. Н. Мандатное разграничение прав в Linux [Текст] / Кулябов Д. С., Королькова А. В., Геворкян М. Н. - Москва: - 5 с. [^1]: Мандатное разграничение прав в Linux.
2. Справочник 70 основных команд Linux: полное описание с примерами (https://eternalhost.net/blog/sozdanie-saytov/osnovnye-komandy-linux)