

# **Лабораторная работа № 6**

**Мандатное разграничение прав в Linux**

Тарусов Артём Сергеевич

# Содержание

<b>Цель работы</b>	<b>4</b>
<b>Задание</b>	<b>5</b>
<b>Теоретическое введение</b>	<b>6</b>
<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>Выводы</b>	<b>17</b>
<b>Список литературы</b>	<b>18</b>

## Список иллюстраций

1	Конфигурация SELinux . . . . .	7
2	Обращение к веб-серверу . . . . .	8
3	Контекст безопасности веб-сервера Apache . . . . .	8
4	Текущее состояние переключателей SELinux для Apache . . . . .	9
5	Статистика по политике . . . . .	10
6	Тип файлов и поддиректорий, находящихся в директории /var/www . . .	10
7	Тип файлов, находящихся в директории /var/www/html . . . . .	11
8	Круг пользователей, которым разрешено создание файлов в директории /var/www/html . . . . .	11
9	Создание файла /var/www/html/test.html . . . . .	11
10	Работа с параметрами readfile . . . . .	11
11	Файл test.html в браузере . . . . .	12
12	Вызов справки и тип файла test.html . . . . .	12
13	Изменение контекста . . . . .	12
14	Файл test.html в браузере после изменения контекста . . . . .	13
15	Содержимое логов . . . . .	13
16	Изменение содержимого файла /etc/httpd/httpd.conf . . . . .	14
17	Перезапуск веб-сервера . . . . .	14
18	Лог-файл tail -nl /var/log/messages . . . . .	14
19	Попытка добавления порта 81 в список и вывод списка допустимых портов	15
20	Повторный запуск веб-сервера . . . . .	15
21	Файл test.html в браузере после возвращения контекста . . . . .	15
22	Параметр Listen после возвращения значения . . . . .	16
23	Попытка удаления привязки к порту 81 . . . . .	16
24	Удаление файла /var/www/html/test.html . . . . .	16

## **Цель работы**

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux<sup>1</sup>. Проверить работу SELinx на практике совместно с веб-сервером Apache.

## **Задание**

1. Настроить и запустить сервер Apache.
2. Исследовать влияние параметров сервера на его работу.

# Теоретическое введение

- Операционная система — это комплекс программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем [1].
- Права доступа определяют, какие действия конкретный пользователь может или не может совершать с определенными файлами и каталогами. С помощью разрешений можно создать надежную среду — такую, в которой никто не может менять содержимое ваших документов или повредить системные файлы. [2].

## Выполнение лабораторной работы

1. Войдем в систему с полученными учётными данными и убедимся, что SELinux работает в режиме enforcing политики targeted (fig. 1).

```
[astarusov@user ~]$ getenforce
Enforcing
[astarusov@user ~]$ 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
```

Рис. 1: Конфигурация SELinux

2. Обратимся с помощью браузера к веб-серверу, запущенному на нашем компьютере, и убедимся, что последний работает (fig. 2).

```
[astarusov@user ~]$ service httpd start
Redirecting to /bin/systemctl start httpd.service
[astarusov@user ~]$ service httpd status
Redirecting to /bin/systemctl status httpd.service
• httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Thu 2023-10-12 14:11:31 MSK; 4s ago
     Docs: man:httpd.service(8)
    Main PID: 40939 (httpd)
      Status: "Started, listening on: port 80"
      Tasks: 213 (limit: 12221)
     Memory: 27.3M
        CPU: 187ms
    CGroup: /system.slice/httpd.service
           └─40939 /usr/sbin/httpd -DFOREGROUND
             └─40948 /usr/sbin/httpd -DFOREGROUND
               └─40949 /usr/sbin/httpd -DFOREGROUND
                 └─40950 /usr/sbin/httpd -DFOREGROUND
                   └─40951 /usr/sbin/httpd -DFOREGROUND

Oct 12 14:11:30 user.localdomain systemd[1]: Starting The Apache HTTP Server...
Oct 12 14:11:31 user.localdomain systemd[1]: Started The Apache HTTP Server.
Oct 12 14:11:31 user.localdomain httpd[40939]: Server configured, listening on: port 80
```

Рис. 2: Обращение к веб-серверу

3. Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности (fig. 3).

```
[astarusov@user ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root      40939  0.4  0.5 20116 11428 ?        Ss   14:11   0:00 /usr/sbin/httpd -D
FOREGROUND
system_u:system_r:httpd_t:s0 apache  40948  0.0  0.3 21600  7428 ?        S    14:11   0:00 /usr/sbin/httpd -D
FOREGROUND
system_u:system_r:httpd_t:s0 apache  40949  0.1  0.6 1669260 13040 ?      Sl   14:11   0:00 /usr/sbin/httpd -D
FOREGROUND
system_u:system_r:httpd_t:s0 apache  40950  0.1  0.5 1538124 10992 ?      Sl   14:11   0:00 /usr/sbin/httpd -D
FOREGROUND
system_u:system_r:httpd_t:s0 apache  40951  0.1  0.5 1538124 10992 ?      Sl   14:11   0:00 /usr/sbin/httpd -D
FOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-c0.c1023 astarus+ 41198 0.0  0.1 221664 2368 pts/0  S+  14:11   0:00 grep
--color=auto httpd
```

Рис. 3: Контекст безопасности веб-сервера Apache

4. Посмотрим текущее состояние переключателей SELinux для Apache (fig. 4).



```
[astarusov@user ~]$ sestatus -b | grep httpd
httpd_anon_write off
httpd_builtin_scripting on
httpd_can_check_spam off
httpd_can_connect_ftp off
httpd_can_connect_ldap off
httpd_can_connect_mythtv off
httpd_can_connect_zabbix off
httpd_can_manage_courier_spool off
httpd_can_network_connect off
httpd_can_network_connect_cobbler off
httpd_can_network_connect_db off
httpd_can_network_memcache off
httpd_can_network_relay off
httpd_can_sendmail off
httpd_dbus_avaahi off
httpd_dbus_sssd off
httpd_dontaudit_search_dirs off
httpd_enable_cgi on
httpd_enable_ftp_server off
httpd_enable_homedirs off
httpd_execmem off
httpd_graceful_shutdown off
httpd_manage_ipa off
httpd_mod_auth_ntlm_winbind off
httpd_mod_auth_pam off
httpd_read_user_content off
httpd_run_ipa off
httpd_run_preupgrade off
httpd_run_stickshift off
httpd_serve_cobbler_files off
httpd_setrlimit off
httpd_ssi_exec off
httpd_sys_script_anon_write off
httpd_tmp_exec off
httpd_tty_comm off
httpd_unified off
httpd_use_cifs off
httpd_use_fusefs off
httpd_use_gpg off
httpd_use_nfs off
httpd_use_openscryptoki off
httpd_use_openstack off
httpd_use_sasl off
httpd_verify_dns off
```

Рис. 4: Текущее состояние переключателей SELinux для Apache

5. Посмотрим статистику по политике с помощью команды seinfo (fig. 5).

```
[astarusov@user ~]$ seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:          33 (MLS enabled)
Target Policy:           selinux
Handle unknown classes:  allow

Classes:          135      Permissions:          457
Sensitivities:     1      Categories:          1024
Types:            5100    Attributes:           258
Users:            8      Roles:               14
Booleans:         353    Cond. Expr.:         384
Allow:            65000   Neverallow:           0
Auditallow:       170    Dontaudit:           8572
Type_trans:       265341  Type_change:          87
Type_member:       35    Range_trans:         6164
Role allow:        38    Role_trans:          420
Constraints:       70    Validatetrans:        0
MLS Constrain:     72    MLS Val. Tran:        0
Permissives:       2     Polcap:               6
Defaults:          7     Typebounds:           0
Allowxperm:        0     Neverallowxperm:      0
Auditallowxperm:   0     Dontauditxperm:       0
Ibendportcon:      0     Ibpkeycon:            0
Initial SIDs:      27    Fs_use:               35
Genfscon:          109   Portcon:              660
Netifcon:          0     Nodecon:              0
```

Рис. 5: Статистика по политике

6. Определим тип файлов и поддиректорий, находящихся в директории /var/www (fig. 6).

```
[astarusov@user ~]$ 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 23:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 May 16 23:21 html
```

Рис. 6: Тип файлов и поддиректорий, находящихся в директории /var/www

7. Определим тип файлов, находящихся в директории /var/www/html (fig. 7).

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

Рис. 7: Тип файлов, находящихся в директории /var/www/html

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

```
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 May 16 23:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 May 16 23:21 html
```

Рис. 8: Круг пользователей, которым разрешено создание файлов в директории /var/www/html

9. Создадим от имени суперпользователя html-файл /var/www/html/test.html (fig. 9).

```
[root@user astarusov]# touch /var/www/html/test.html
```

Рис. 9: Создание файла /var/www/html/test.html

Заполним его следующим содержимым:

```
<html>
<body>test</body>
</html>
```

10. Проверим контекст созданного нами файла (fig. 10).

```
[root@user html]# ls -lZ /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
```

Рис. 10: Работа с параметрами readfile

Как видим по умолчанию присваивается контекст unconfined\_u:object\_r:httpd\_sys\_content\_t:s0

11. Обратимся к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>.  
Убедимся, что файл был успешно отображён (fig. 11).

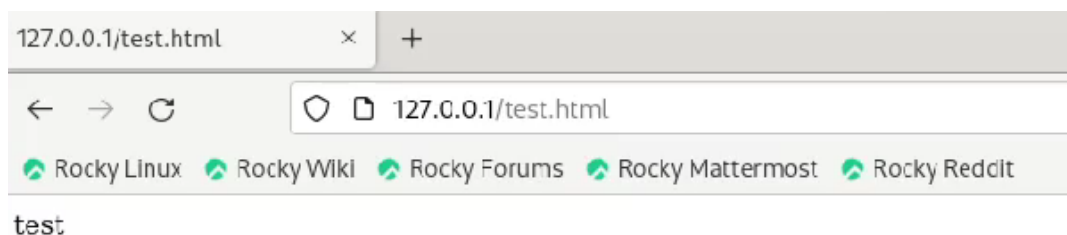


Рис. 11: Файл test.html в браузере

12. Изучим справку `man httpd_selinux` и выясним, какие контексты файлов определены для `httpd`. Сопоставим их с типом файла `test.html` (fig. 12).

```
[root@user html]# man httpd_selinux
No manual entry for httpd_selinux
[root@user html]# man selinux
[root@user html]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
```

Рис. 12: Вызов справки и тип файла test.html

13. Изменим контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на `samba_share_t` (fig. 13).

```
[root@user html]# chcon -t samba_share_t /var/www/html/test.html
[root@user html]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Рис. 13: Изменение контекста

14. Попробуем ещё раз получить доступ к файлу через веб-сервер (fig. 14).

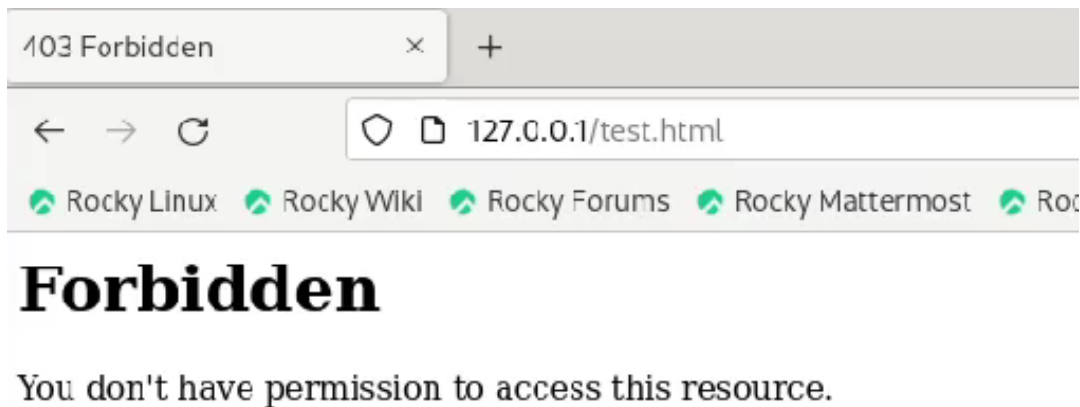


Рис. 14: Файл test.html в браузере после изменения контекста

15. Просмотрим log-файлы веб-сервера Apache и системный лог-файл (fig. 15).

```
[root@user html]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct 12 14:22 /var/www/html/test.html
[root@user html]# tail /var/log/messages
Oct 12 14:33:16 user systemd[1]: Created slice Slice /system/dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged.
Oct 12 14:33:16 user systemd[1]: Started dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@0.service.
Oct 12 14:33:18 user setroubleshoot[42422]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /va
r/www/html/test.html. For complete SELinux messages run: sealert -l 8742a7c7-5214-4df2-bf18-6972a2b5bb78
Oct 12 14:33:18 user setroubleshoot[42422]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /va
r/www/html/test.html.#012#012***** Plugin restorecon (92.2 confidence) suggests *****#012#012If
you want to fix 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/test.htm
l#012#012***** Plugin public_content (7.83 confidence) suggests *****#012#012If you want to treat t
est.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# restorecon -v '/var/www/html/t
est.html'#012#012***** Plugin catchall (1.41 confidence) suggests *****#012#012If you believe
that httpd should be allowed getattr access on the test.html file by default.#012Then you should report this as a bu
g.#012You can generate a local policy module to allow this access.#012Do#012allow this access for now by executing:#0
12# ausearch -c 'httpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
Oct 12 14:33:18 user setroubleshoot[42422]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /va
r/www/html/test.html. For complete SELinux messages run: sealert -l 8742a7c7-5214-4df2-bf18-6972a2b5bb78
Oct 12 14:33:18 user setroubleshoot[42422]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /va
r/www/html/test.html.#012#012***** Plugin restorecon (92.2 confidence) suggests *****#012#012If
you want to fix 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/test.htm
l#012#012***** Plugin public_content (7.83 confidence) suggests *****#012#012If you want to treat t
est.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# restorecon -v '/var/www/html/t
est.html'#012#012***** Plugin catchall (1.41 confidence) suggests *****#012#012If you believe
that httpd should be allowed getattr access on the test.html file by default.#012Then you should report this as a bu
g.#012You can generate a local policy module to allow this access.#012Do#012allow this access for now by executing:#0
12# ausearch -c 'httpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
Oct 12 14:33:28 user systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@0.service: Deactivated successf
```

Рис. 15: Содержимое логов

Как видим, нам не удалось получить доступ к файлу как раз из-за измененного контекста.

16. Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81 (fig. 16).

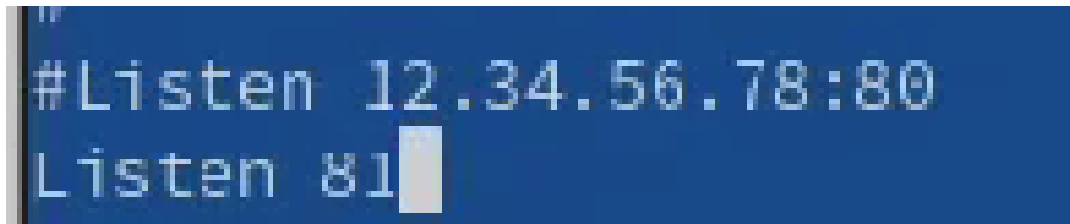


Рис. 16: Изменение содержимого файла /etc/httpd/httpd.conf

17. Выполним перезапуск веб-сервера. Сбоя не произошло (fig. 17).

```
[root@user conf]# service httpd stop
Redirecting to /bin/systemctl stop httpd.service
[root@user conf]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@user conf]# service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Thu 2023-10-12 14:44:58 MSK; 15s ago
     Docs: man:httpd.service(8)
  Main PID: 42774 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 213 (limit: 12221)
   Memory: 23.1M
      CPU: 172ms
  CGroup: /system.slice/httpd.service
          └─42774 /usr/sbin/httpd -DFOREGROUND
            └─42775 /usr/sbin/httpd -DFOREGROUND
              └─42776 /usr/sbin/httpd -DFOREGROUND
                └─42820 /usr/sbin/httpd -DFOREGROUND
                  └─42837 /usr/sbin/httpd -DFOREGROUND

Oct 12 14:44:58 user.localdomain systemd[1]: Starting The Apache HTTP Server...
Oct 12 14:44:58 user.localdomain systemd[1]: Started The Apache HTTP Server.
Oct 12 14:44:58 user.localdomain httpd[42774]: Server configured, listening on: port 81
```

Рис. 17: Перезапуск веб-сервера

18. Проанализируем лог-файлы (fig. 18).

```
[root@user conf]# tail /var/log/messages
Oct 12 14:43:54 user systemd[1]: Started Fingerprint Authentication Daemon.
Oct 12 14:43:58 user NetworkManager[927]: <info> [1697111038.6420] agent-manager: agent[ef6fe3c18efc9fb5,:1.71/org.g
nome.Shell.NetworkAgent/1000]: agent registered
Oct 12 14:44:24 user systemd[1]: fprintd.service: Deactivated successfully.
Oct 12 14:44:52 user systemd[1]: Stopping The Apache HTTP Server...
Oct 12 14:44:53 user systemd[1]: httpd.service: Deactivated successfully.
Oct 12 14:44:53 user systemd[1]: Stopped The Apache HTTP Server.
Oct 12 14:44:53 user systemd[1]: httpd.service: Consumed 12.225s CPU time.
Oct 12 14:44:58 user systemd[1]: Starting The Apache HTTP Server...
Oct 12 14:44:58 user systemd[1]: Started The Apache HTTP Server.
Oct 12 14:44:58 user httpd[42774]: Server configured, listening on: port 81
```

Рис. 18: Лог-файл tail -nl /var/log/messages

19. Выполним команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверим список портов командой `semanage port -l | grep http_port_t`. Убедимся, что порт 81 есть в списке. (fig. 19).

```
[root@user audit]# semanage port -a -t http_port_t tcp 81
usage: semanage [-h]
               {import,export,login,user,port,ibpkey,ibendport,interface,module,
               taudit}
               ...
semanage: error: unrecognized arguments: 81
[root@user audit]# semanage port -l | grep http_port_t
http_port_t      tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t tcp      5988
```

Рис. 19: Попытка добавления порта 81 в список и вывод списка допустимых портов

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

```
Redirecting to /bin/systemctl start httpd.service
[root@user audit]# service httpd status
Redirecting to /bin/systemctl status httpd.service
• httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
  Active: active (running) since Thu 2023-10-12 14:44:58 MSK; 7min ago
    Docs: man:httpd.service(8)
 Main PID: 42774 (httpd)
  Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
   Tasks: 213 (limit: 12221)
  Memory: 23.1M
    CPU: 2.407s
  CGroup: /system.slice/httpd.service
          └─42774 /usr/sbin/httpd -DFOREGROUND
            └─42775 /usr/sbin/httpd -DFOREGROUND
              └─42776 /usr/sbin/httpd -DFOREGROUND
                └─42820 /usr/sbin/httpd -DFOREGROUND
                  └─42837 /usr/sbin/httpd -DFOREGROUND

Oct 12 14:44:58 user.localdomain systemd[1]: Starting The Apache HTTP Server...
Oct 12 14:44:58 user.localdomain systemd[1]: Started The Apache HTTP Server.
Oct 12 14:44:58 user.localdomain httpd[42774]: Server configured, listening on: port 81
```

Рис. 20: Повторный запуск веб-сервера

21. Вернем контекст `httpd_sys_content__t` к файлу `/var/www/html/ test.html`. Попробуем получить доступ к файлу через веб-сервер (fig. 21).

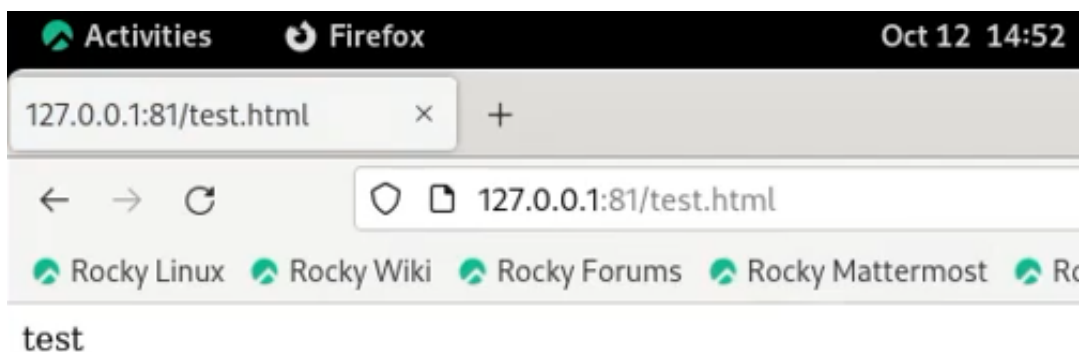
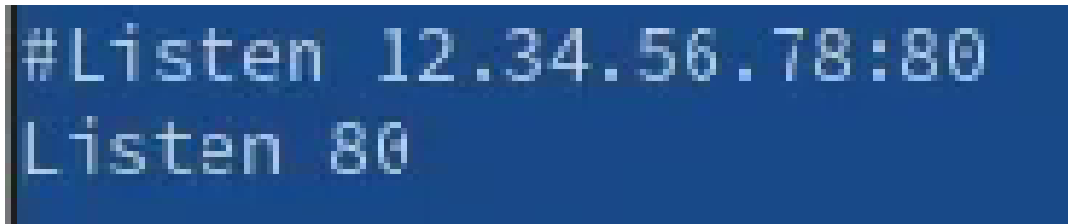


Рис. 21: Файл test.html в браузере после возвращения контекста

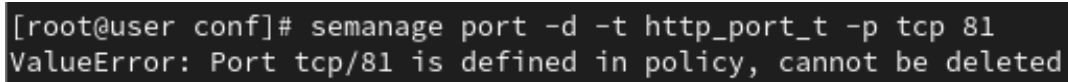
22. Исправим обратно конфигурационный файл apache, вернув Listen 80 (fig. 22).



```
#Listen 12.34.56.78:80
Listen 80
```

Рис. 22: Параметр Listen после возвращения значения

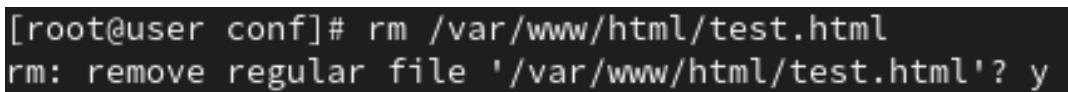
23. Попробуем удалить привязку http\_port\_t к 81. Удаление невозможно(fig. 23).



```
[root@user conf]# semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
```

Рис. 23: Попытка удаления привязки к порту 81

24. Удалим файл /var/www/html/test.html(fig. 24).



```
[root@user conf]# rm /var/www/html/test.html
rm: remove regular file '/var/www/html/test.html'? y
```

Рис. 24: Удаление файла /var/www/html/test.html



## **Выводы**

В рамках данной лабораторной работы были развиты навыки администрирования ОС Linux. Получено первое практическое знакомство с технологией SELinux<sup>1</sup>. Проверена работа SELinx на практике совместно с веб-сервером Apache.

## **Список литературы**

[1] <https://blog.skillfactory.ru/glossary/operaczionnaya-sistema/>

[2] <https://codechick.io/tutorials/unix-linux/unix-linux-permissions>