

# **Лабораторная работа №6**

**Основы информационной безопасности**

Кондрашова Анастасия Андреевна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>13</b>

## Список иллюстраций

2.1	Getenforce и sestatus . . . . .	6
2.2	Работа сервера . . . . .	6
2.3	Контекст безопасности . . . . .	7
2.4	Текущее состояние переключателей . . . . .	7
2.5	Статистика по политике . . . . .	8
2.6	тип файлов и круг пользователей . . . . .	8
2.7	Создание файла . . . . .	9
2.8	Контекст файла . . . . .	9
2.9	Корректное отображение контента . . . . .	9
2.10	Смена контекста . . . . .	9
2.11	Ошибка доступа . . . . .	10
2.12	Просмотр логов . . . . .	10
2.13	Прослушивание 81 порта . . . . .	11
2.14	Перезапуск сервера . . . . .	11
2.15	Установка порта 81 . . . . .	11
2.16	Смена контекста 2 . . . . .	12
2.17	Удаление 81 порта . . . . .	12
2.18	Удаление файла . . . . .	12

## Список таблиц

# 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 Выполнение лабораторной работы

1. С помощью команды `getenforce` убеждаемся, что SELinux работает в режиме `enforcing`, а с помощью команды `sestatus` устанавливаем политику `targeted`

```
[aakondrashova@aakondrashova ~]$ getenforce
Enforcing
[aakondrashova@aakondrashova ~]$ sestatus targeted
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. 2.1: Getenforce и sestatus

2. Убеждаемся, что сервер работает с помощью команды `service httpd status`

```
[aakondrashova@aakondrashova init.d]$ sudo systemctl start httpd
[aakondrashova@aakondrashova init.d]$ 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; vendor pre
   Active: active (running) since Fri 2022-10-14 14:52:51 MSK; 18s ago
     Docs: man:httpd.service(8)
   Main PID: 79268 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes
   Tasks: 213 (limit: 12164)
   Memory: 45.1M
     CPU: 1.039s
   CGroup: /system.slice/httpd.service
           └─79268 /usr/sbin/httpd -DFOREGROUND
             └─79276 /usr/sbin/httpd -DFOREGROUND
               └─79277 /usr/sbin/httpd -DFOREGROUND
                 └─79278 /usr/sbin/httpd -DFOREGROUND
                   └─79308 /usr/sbin/httpd -DFOREGROUND

Oct 14 14:52:10 aakondrashova.localdomain systemd[1]: Starting The Apache HTTP
Oct 14 14:52:51 aakondrashova.localdomain httpd[79268]: Server configured, list
```

Рис. 2.2: Работа сервера

3. С помощью команды `ps -eZ` находим, что контекст безопасности Apache - `httpd_t`

```
[aakondrashova@aakondrashova init.d]$ ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      79268 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      79276 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      79277 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      79278 ?        00:00:00 httpd
system_u:system_r:httpd_t:s0      79308 ?        00:00:00 httpd
```

Рис. 2.3: Контекст безопасности

4. Смотрим текущее состояние переключателей командой `sestatus -b httpd`

```
[aakondrashova@aakondrashova init.d]$ sestatus -b httpd
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

Policy booleans:
abrt_anon_write                  off
abrt_handle_event                off
abrt_upload_watch_anon_write     on
antivirus_can_scan_system        off
```

Рис. 2.4: Текущее состояние переключателей

5. Смотрим статистику по политике командой `seinfo`. Узнаём, что множество пользователей — 8, ролей — 14, типов — 5002

```

[aakondrashova@aakondrashova init.d]$ 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
Allow:            63996   Neverallow:           0
Auditallow:       168     Dontaudit:            8417
Type_trans:      258486   Type_change:          87
Type_member:      35      Range_trans:          5960
Role_allow:       38      Role_trans:           420
Constraints:      72      Validatetrans:         0
MLS Constrain:    72      MLS Val. Tran:         0
Permissives:      0       Polcap:                5
Defaults:         7       Typebounds:           0
Allowxperm:       0       Neverallowxperm:       0
Auditallowxperm:  0       Dontauditxperm:        0
Ibendportcon:     0       Ibpkeycon:             0
Initial SIDs:     27      Fs_use:                33
Genfscon:         106     Portcon:               651
Netifcon:         0       Nodecon:               0

```

Рис. 2.5: Статистика по политике

6. Определяем тип файлов и круг пользователей с правой на создание и под-директорий в директориях /var/www и /var/www/html командой `ls -lZ`

```

[aakondrashova@aakondrashova init.d]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system u:object_r:httpd_sys_script_exec_t:s0 6 May 13 15:56 cgi-bin
drwxr-xr-x. 2 root root system u:object_r:httpd_sys_content_t:s0 6 May 13 15:56 html
[aakondrashova@aakondrashova init.d]$ ls -lZ /var/www/html
total 0
[aakondrashova@aakondrashova init.d]$ ls -l /var/www/html
total 0
[aakondrashova@aakondrashova init.d]$ ls -l /var/www
total 0
drwxr-xr-x. 2 root root 6 May 13 15:56 cgi-bin
drwxr-xr-x. 2 root root 6 May 13 15:56 html

```

Рис. 2.6: тип файлов и круг пользователей

7. От имени суперпользователя создаём файл /var/www/html/test.html



```

[aakondrashova@aakondrashova init.d]$ su
Password:
[root@aakondrashova init.d]# touch /var/www/html/test.html
[root@aakondrashova init.d]# nano /var/www/html/test.html
[root@aakondrashova init.d]# cat /var/www/html/test.html
<html>
<body>test</body>
</html>

```

Рис. 2.7: Создание файла

8. Командой `matchpathcon` узнаём контекст файла `test.html` и директории `/var/www/html` — это `httpd_sys_content_t`.

```

[root@aakondrashova init.d]# matchpathcon /var/www/html/test.html
/var/www/html/test.html system_u:object_r:httpd_sys_content_t:s0
[root@aakondrashova init.d]# matchpathcon -V /var/www/html
/var/www/html verified.
[root@aakondrashova init.d]# matchpathcon /var/www/html
/var/www/html system_u:object_r:httpd_sys_content_t:s0

```

Рис. 2.8: Контекст файла

9. Обращаемся к файлу через ссылку в веб-браузере. Контент отображён корректно.

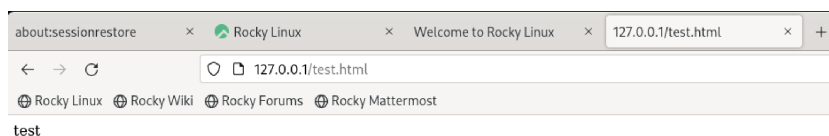


Рис. 2.9: Корректное отображение контента

10. Изучая справку `man httpd_selunix` узнаём, что для `httpd` определены следующие контексты: `httpd_sys_content_t`, `httpd_sys_script_exec_t`, `httpd_sys_script_ro_t`, `httpd_sys_script_rw_t`, `httpd_sys_script_ra_t`, `httpd_unconfined_script_exec_t`. Меняем контекст файла `test.html` командой `chcon -t`.

```

[root@aakondrashova init.d]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@aakondrashova init.d]# chcon -t samba_share_t /var/www/html/test.html
[root@aakondrashova init.d]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html

```

Рис. 2.10: Смена контекста

11. При повторной попытке открыть файл через веб-браузер получаем ошибку доступа.

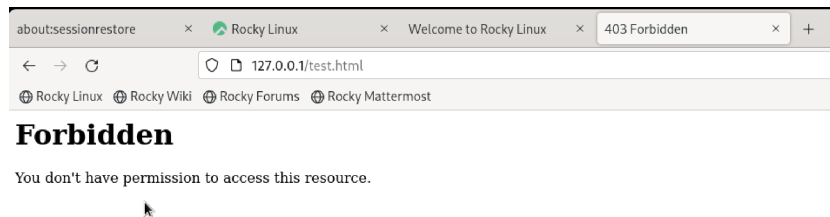


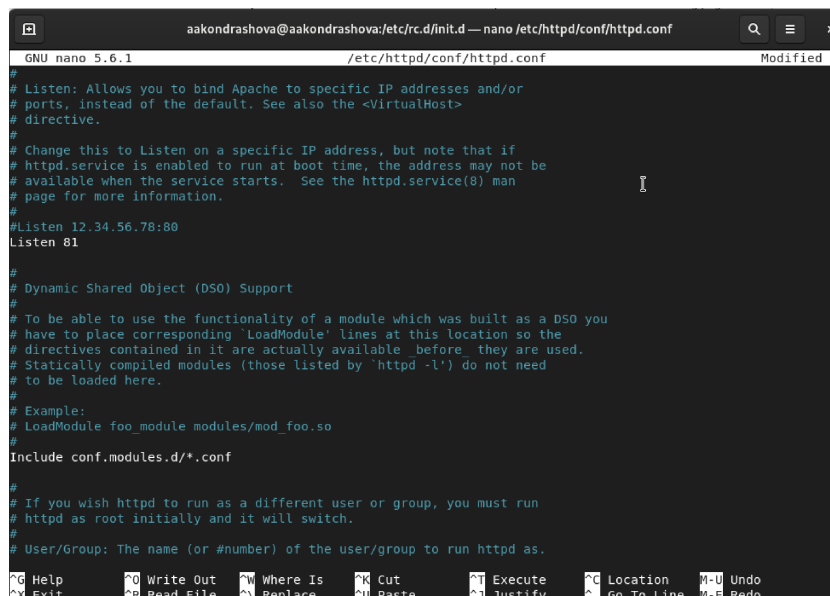
Рис. 2.11: Ошибка доступа

12. Убеждаемся, что файл доступен для чтения всем пользователям командой `ls -l`. Далее смотрим log-файлы веб-сервера Apache командой `tail`, где показаны ошибки.



Рис. 2.12: Просмотр логов

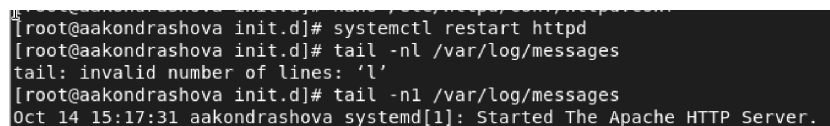
13. Устанавливаем веб-сервер Apache на прослушивание TCP-порта 81, изменяя строку `Listen` в файле `/etc/httpd/conf/httpd.conf`.



```
aakondrashova@aakondrashova:/etc/rc.d/init.d$ nano /etc/httpd/conf/httpd.conf
GNU nano 5.6.1 /etc/httpd/conf/httpd.conf Modified
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts. See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 81
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available before they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf
#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
```

Рис. 2.13: Прослушивание 81 порта

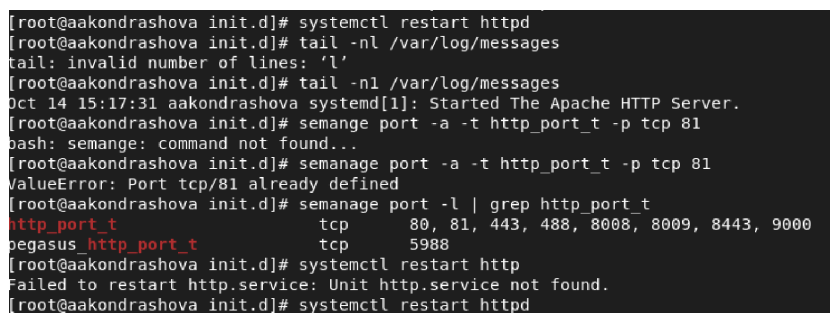
14. Перезапускаем сервер и смотри данные log-файлов веб-сервера Apache.



```
[root@aakondrashova init.d]# systemctl restart httpd
[root@aakondrashova init.d]# tail -nl /var/log/messages
tail: invalid number of lines: 'l'
[root@aakondrashova init.d]# tail -n1 /var/log/messages
Oct 14 15:17:31 aakondrashova systemd[1]: Started The Apache HTTP Server.
```

Рис. 2.14: Перезапуск сервера

15. Устанавливаем для веб-сервера Apache порт TCP-81 и проверяем его наличие в списке портов командой semanage.



```
[root@aakondrashova init.d]# systemctl restart httpd
[root@aakondrashova init.d]# tail -nl /var/log/messages
tail: invalid number of lines: 'l'
[root@aakondrashova init.d]# tail -n1 /var/log/messages
Oct 14 15:17:31 aakondrashova systemd[1]: Started The Apache HTTP Server.
[root@aakondrashova init.d]# semanage port -a -t http_port_t -p tcp 81
bash: semanage: command not found...
[root@aakondrashova init.d]# semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[root@aakondrashova init.d]# 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
[root@aakondrashova init.d]# systemctl restart httpd
failed to restart http.service: Unit http.service not found.
[root@aakondrashova init.d]# systemctl restart httpd
```

Рис. 2.15: Установка порта 81

16. Возвращаем файлу test.html контекст httpd\_sys\_content\_t и снова успешно просматриваем страницу в веб-браузере.

```
[root@aakondrashova init.d]# chcon -t httpd_sys_content_t /var/www/html/test.html
```

Рис. 2.16: Смена контекста 2

17. Возвращаем в конфигурационный файл прослушивание порта 80 и удаляем порт 81 из списка портов.

```
[root@aakondrashova init.d]# nano /etc/httpd/conf/httpd.conf
[root@aakondrashova init.d]# semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
[root@aakondrashova init.d]# semanage port -l | grep http port t
grep: port: No such file or directory
grep: t: No such file or directory
BrokenPipeError: [Errno 32] Broken pipe
```

Рис. 2.17: Удаление 81 порта

18. Удаляем файл test.html.

```
[root@aakondrashova init.d]# rm /var/www/html/test.html
rm: remove regular file '/var/www/html/test.html'? y
[root@aakondrashova init.d]# ls /var/www/html
```

Рис. 2.18: Удаление файла

## **3 Выводы**

Я получила основные навыки администрирования в ОС Linux и проверила работу SELinux на практике совместно с веб-сервером Apache.