

Отчёт по лабораторной работе №6

Дисциплина: Основы информационной безопасности

Мишина Анастасия Алексеевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	15
	Список литературы	16

Список иллюстраций

2.1	Проверка режима SELinux, запуск веб-сервера	6
2.2	Контекст безопасности веб-сервера и состояние переключателей SELinux	7
2.3	Статистика по политике	8
2.4	Определение типов файлов и поддиректорий www и html. Создание файла test.html	8
2.5	Файл test.html	9
2.6	Контекст файла test.html	9
2.7	Запуск файла в браузере	9
2.8	Изучение справок, контекст файла test.html	9
2.9	Смена контекста файла, проверка	10
2.10	Попытка просмотра	10
2.11	Права доступа, системный log-файл	10
2.12	Файл httpd.conf	11
2.13	Перезапуск веб-сервера Apache	11
2.14	Сбой веб-сервера	12
2.15	Просмотр лог-файлов	12
2.16	Просмотр лог-файлов	13
2.17	Проверка списка портов, перезагрузка сервера, возвращение контекста	13
2.18	Просмотр файла через сервер	13
2.19	Файл httpd.conf	14
2.20	Удаляем привязку http_port_t к 81, выполняем проверку, удаляем файл test.html	14

Список таблиц

1 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].

2 Выполнение лабораторной работы

Открываем терминал и убеждаемся, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`. Запустим веб-сервер и убедимся, что он работает: `service httpd start` и `service httpd status` (рис. 2.1).

```
[aamishina@aamishina ~]$ getenforce
Enforcing
[aamishina@aamishina ~]$ 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
[aamishina@aamishina ~]$ 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: inactive (dead)
   Docs: man:httpd.service(8)
[aamishina@aamishina ~]$ service httpd start
Redirecting to /bin/systemctl start httpd.service
[aamishina@aamishina ~]$ 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 Wed 2024-04-24 12:30:55 MSK; 12s ago
   Docs: man:httpd.service(8)
  Main PID: 90060 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B"
    Tasks: 213 (limit: 50316)
   Memory: 40.9M
      CPU: 112ms
   CGroup: /system.slice/httpd.service
           └─90060 /usr/sbin/httpd -DFOREGROUND
             └─90068 /usr/sbin/httpd -DFOREGROUND
               └─90069 /usr/sbin/httpd -DFOREGROUND
                 └─90070 /usr/sbin/httpd -DFOREGROUND
                   └─90071 /usr/sbin/httpd -DFOREGROUND

Apr 24 12:30:55 aamishina.localdomain systemd[1]: Starting The Apache HTTP Server...
Apr 24 12:30:55 aamishina.localdomain httpd[90060]: Server configured, listening on: port 80
Apr 24 12:30:55 aamishina.localdomain systemd[1]: Started The Apache HTTP Server.
lines 1-19/19 (END)
```

Рис. 2.1: Проверка режима SELinux, запуск веб-сервера

Определим контекст безопасности веб-сервера: `ps auxZ | grep httpd`. Посмотрим текущее состояние переключателей SELinux для Apache с помощью команды: `sestatus -bigrep httpd`. Обратим внимание, что многие из них находятся в положении «off» (рис. 2.2).

```

[aaamishina@aaamishina ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 90060 0.0 0.1 29580 10376 ? Ss 12:30 0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 90068 0.0 0.1 31140 8724 ? S 12:30 0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 90069 0.0 0.2 2470132 24304 ? Sl 12:30 0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 90070 0.0 0.1 2338036 16116 ? Sl 12:30 0:00
/usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 90071 0.0 0.1 2338036 16116 ? Sl 12:30 0:00
/usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 aaamishi+ 90365 0.0 0.0 221368 2120 pts/0 S+
12:32 0:00 grep --color=auto httpd
[aaamishina@aaamishina ~]$ 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_avahi 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

```

Рис. 2.2: Контекст безопасности веб-сервера и состояние переключателей SELinux

Посмотрим статистику по политике с помощью команды `seinfo`, также определим множество пользователей, ролей, типов (рис. 2.3).

```
[aamishina@aamishina ~]$ 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:            5135     Attributes:         259
Users:            8        Roles:              15
Booleans:         357     Cond. Expr.:       390
Allow:            65409    Neverallow:         0
Auditallow:       172     Dontaudit:          8647
Type_trans:       267813  Type_change:        94
Type_member:       37     Range_trans:        6164
Role allow:       39      Role_trans:         419
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:             665
Netifcon:         0       Nodecon:             0

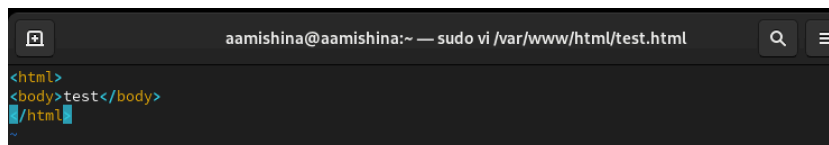
[aamishina@aamishina ~]$
```

Рис. 2.3: Статистика по политике

Определяем тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды `ls -lZ /var/www`. Определяем тип файлов, находящихся в директории /var/www/html: `ls -lZ /var/www/html`. Определяем круг пользователей, которым разрешено создание файлов в директории /var/www/html (рис. 2.4). Создаем от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html (рис. 2.5).

```
[aamishina@aamishina ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Oct 28 12:35 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0    6 Oct 28 12:35 html
[aamishina@aamishina ~]$ ls -lZ /var/www/html
total 0
[aamishina@aamishina ~]$
[aamishina@aamishina ~]$ sudo touch /var/www/html/test.html
[sudo] password for aamishina:
[aamishina@aamishina ~]$ sudo vi /var/www/html/test.html
[aamishina@aamishina ~]$
```

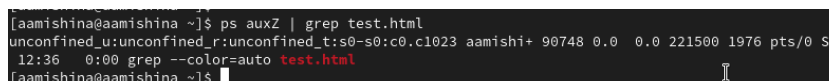
Рис. 2.4: Определение типов файлов и поддиректорий www и html. Создание файла test.html



```
aamishina@aamishina:~ — sudo vi /var/www/html/test.html
<html>
<body>test</body>
</html>
```

Рис. 2.5: Файл test.html

Проверяем контекст созданного файла (рис. 2.6).



```
[aamishina@aamishina ~]$ ps auxZ | grep test.html
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 aamishi+ 90748 0.0  0.0 221500 1976 pts/0 S+
12:36  0:00 grep --color=auto test.html
```

Рис. 2.6: Контекст файла test.html

Обратимся к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Убеждаемся, что файл был успешно отображён (рис. 2.7).

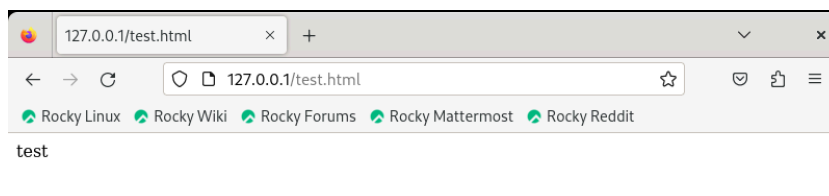
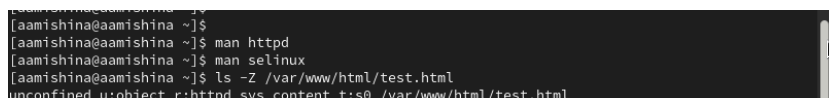


Рис. 2.7: Запуск файла в браузере

Изучаем справку `man httpd` и `man selinux`. Проверять контекст файла можно командой `ls -Z`, т.е. `ls -Z /var/www/html/test.html` (рис. 2.8).



```
[aamishina@aamishina ~]$
[aamishina@aamishina ~]$ man httpd
[aamishina@aamishina ~]$ man selinux
[aamishina@aamishina ~]$ ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
```

Рис. 2.8: Изучение справок, контекст файла test.html

Сменим контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на любой другой, к которому процесс `httpd` не должен иметь доступа, например, на `samba_share_t`: `chcon -t samba_share_t /var/www/html/test.html`, проверим, что контекст поменялся: `ls -Z /var/www/html/test.html` (рис. 2.9).

```

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

```

Рис. 2.9: Смена контекста файла, проверка

Попробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Получаем сообщение об ошибке: `Forbidden` You don't have permission to access `/test.html` on this server (рис. 2.10).

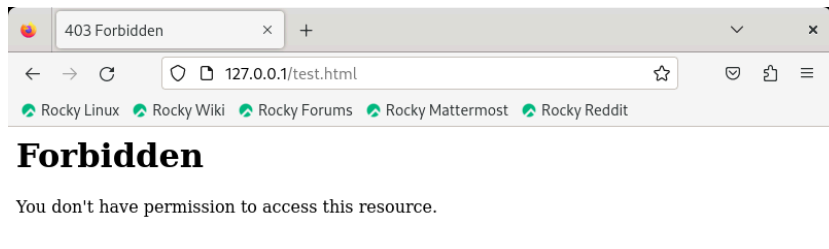


Рис. 2.10: Попытка просмотра

Смотрим права доступа на файл `ls -l /var/www/html/test.html`, также смотрим системный log-файл `tail /var/log/messages` (рис. 2.11).

```

[aamishina@aamishina ~]$
[aamishina@aamishina ~]$ ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Apr 24 12:36 /var/www/html/test.html
[aamishina@aamishina ~]$ tail /var/log/messages
tail: cannot open '/var/log/messages' for reading: Permission denied
[aamishina@aamishina ~]$ sudo tail /var/log/messages
Apr 24 12:40:20 aamishina systemd[1]: Started SETroubleshoot daemon for processing new SELinux denial logs.
Apr 24 12:40:20 aamishina setroubleshoot[91483]: failed to retrieve rpm info for path '/var/www/html/test.html':
Apr 24 12:40:20 aamishina systemd[1]: Created slice Slice /system/dbus-1.1-org.fedoraproject.SetroubleshootPrivileged.
Apr 24 12:40:20 aamishina systemd[1]: Started dbus-1.1-org.fedoraproject.SetroubleshootPrivileged@0.service.
Apr 24 12:40:21 aamishina setroubleshoot[91483]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/www/html/test.html. For complete SELinux messages run: sealert -l 30704005-1d09-4cc7-9293-52ae45c79f48
Apr 24 12:40:21 aamishina setroubleshoot[91483]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/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.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# restorecon -v '/var/www/html/test.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 bug.#012You can generate a local policy module to allow this access.#012Do#012allow this access for now by executing:#012# ausearch -c 'httpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
Apr 24 12:40:21 aamishina setroubleshoot[91483]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/www/html/test.html. For complete SELinux messages run: sealert -l 30704005-1d09-4cc7-9293-52ae45c79f48
Apr 24 12:40:21 aamishina setroubleshoot[91483]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/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 m

```

Рис. 2.11: Права доступа, системный log-файл

Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services). Для этого в файле /etc/httpd/httpd.conf находим строчку Listen 80 и меняем её на Listen 81 (рис. 2.12).

```
# 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
#
```

Рис. 2.12: Файл httpd.conf

Выполняем перезапуск веб-сервера Apache (рис. 2.13). Происходит сбой (рис. 2.14).

```
[aamishina@aamishina ~]$
[aamishina@aamishina ~]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[aamishina@aamishina ~]$
```

Рис. 2.13: Перезапуск веб-сервера Apache

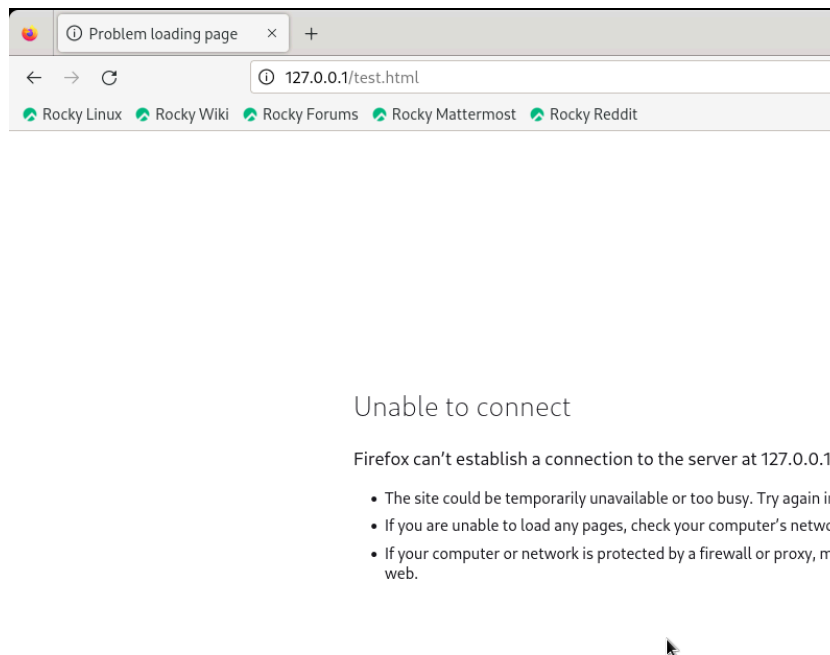


Рис. 2.14: Сбой веб-сервера

Анализируем лог-файлы: `tail -n1 /var/log/messages` (рис. 2.15). Просматриваем файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` (рис. 2.16).

```
[aamishina@aamishina ~]$ sudo tail -n1 /var/log/messages
Apr 24 12:44:40 aamishina systemd[1]: Started The Apache HTTP Server.
[aamishina@aamishina ~]$ sudo cat /var/log/httpd/error_log
[Wed Apr 24 12:30:55.811697 2024] [core:notice] [pid 90060:tid 90060] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Wed Apr 24 12:30:55.813003 2024] [suexec:notice] [pid 90060:tid 90060] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
[Wed Apr 24 12:30:55.834394 2024] [lbmethod_heartbeat:notice] [pid 90060:tid 90060] AH02282: No slottedmem from mod_heartbeat
[Wed Apr 24 12:30:55.838983 2024] [mpm_event:notice] [pid 90060:tid 90060] AH00489: Apache/2.4.57 (Rocky Linux) configured -- resuming normal operations
[Wed Apr 24 12:30:55.839031 2024] [core:notice] [pid 90060:tid 90060] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[Wed Apr 24 12:40:20.133544 2024] [core:error] [pid 90069:tid 90136] (13)Permission denied: [client 127.0.0.1:60880] AH00035: access to /test.html denied (filesystem path '/var/www/html/test.html') because search permissions are missing on a component of the path
[Wed Apr 24 12:44:39.880861 2024] [mpm_event:notice] [pid 90060:tid 90060] AH00492: caught SIGWINCH, shutting down gracefully
[Wed Apr 24 12:44:40.955333 2024] [core:notice] [pid 91629:tid 91629] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Wed Apr 24 12:44:40.956116 2024] [suexec:notice] [pid 91629:tid 91629] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
```

Рис. 2.15: Просмотр лог-файлов

```
[aamishina@aamishina ~]$ sudo cat /var/log/httpd/access_log
127.0.0.1 - - [24/Apr/2024:12:37:27 +0300] "GET /test.html HTTP/1.1" 200 33 "-" "Mozilla/5.0 (X11; Linux aarch64; rv:109.0) Gecko/20100101 Firefox/115.0"
127.0.0.1 - - [24/Apr/2024:12:37:27 +0300] "GET /favicon.ico HTTP/1.1" 404 196 "http://127.0.0.1/test.html" "Mozilla/5.0 (X11; Linux aarch64; rv:109.0) Gecko/20100101 Firefox/115.0"
127.0.0.1 - - [24/Apr/2024:12:40:20 +0300] "GET /test.html HTTP/1.1" 403 199 "-" "Mozilla/5.0 (X11; Linux aarch64; rv:109.0) Gecko/20100101 Firefox/115.0"
[aamishina@aamishina ~]$ sudo cat /var/log/audit/audit.log
type=DAEMON_START msg=audit(1708113402.918:1845): op=start ver=3.0.7 format=enriched kernel=5.14.0-362.8.1.el9_3.aarch64 auid=4294967295 pid=722 uid=0 ses=4294967295 subj=system_u:system_r:auditd_t:s0 res=successAUID="unset" UID="root"
type=SERVICE_START msg=audit(1708113402.910:5): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=systemd-journal-catalog-update comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'UID="root" AUID="unset"
type=CONFIG_CHANGE msg=audit(1708113402.950:6): op=set audit_backlog_limit=8192 old=64 auid=4294967295 ses=4294967295 subj=system_u:system_r:unconfined_service_t:s0 res=1AUID="unset"
type=SYSCALL msg=audit(1708113402.950:6): arch=c00000b7 syscall=206 success=yes exit=60 a0=3 a1=ffffcab55f0 a2=3c a3=0 items=0 ppid=727 pid=737 auid=4294967295 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) ses=4294967295 comm="auditctl" exe="/usr/sbin/auditctl" subj=system_u:system_r:unconfined_service_t:s0 key=(null)ARCH=aarch64 SYSCALL=sendto AUID="unset" UID="root" GID="root" EUID="root" SUID="root" FSUID="root" EGID="root" SGID="root" FSGID="root"
type=PROCTITLE msg=audit(1708113402.950:6): proctitle=2F7362696E2F617564697463746C002D52002F657463
```

Рис. 2.16: Просмотр лог-файлов

Выполняем команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверяем список портов командой `semanage port -l | grep http_port_t`. Убеждаемся, что порт 81 появился в списке. Перезагружаем веб-сервер еще раз. Возвращаем контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html`: `chcon -t httpd_sys_content_t /var/www/html/test.html` (рис. 2.17).

```
[aamishina@aamishina ~]$ sudo semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[aamishina@aamishina ~]$ sudo 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
[aamishina@aamishina ~]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[aamishina@aamishina ~]$ sudo chcon -t httpd_sys_content_t /var/www/html/test.html
[aamishina@aamishina ~]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[aamishina@aamishina ~]$
```

Рис. 2.17: Проверка списка портов, перезагрузка сервера, возвращение контекста

После этого попробуем получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`. Видим содержимое файла — слово «test» (рис. 2.18).

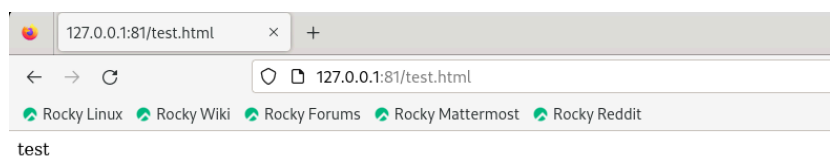


Рис. 2.18: Просмотр файла через сервер

Исправляем обратно конфигурационный файл apache, вернув Listen 80 (рис. 2.19).

```
# Change this to Listen on a specific IP address
# httpd.service is enabled to run at boot time,
# available when the service starts. See the
# page for more information.
#
#Listen 12.34.56.78:80
Listen 80

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module
# have to place corresponding 'LoadModule' line
# directives contained in it are actually available
-- INSERT --
```

Рис. 2.19: Файл httpd.conf

Удалим привязку http_port_t к 81 порту: semanage port -d -t http_port_t -p tcp 81 и проверим, что порт 81 удалён. Удаляем файл var/www/html/test.html: rm /var/www/html/test.html (рис. 2.20).

```
[aamishina@aamishina ~]$ sudo vi /etc/httpd/conf
[aamishina@aamishina ~]$ semanage port -d -t http_port_t -p tcp 81
ValueError: SELinux policy is not managed or store cannot be accessed.
[aamishina@aamishina ~]$ sudo semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
[aamishina@aamishina ~]$ sudo vi /etc/httpd/conf
[aamishina@aamishina ~]$ rm /var/www/html/test.html
rm: remove write-protected regular file '/var/www/html/test.html'? y
rm: cannot remove '/var/www/html/test.html': Permission denied
[aamishina@aamishina ~]$ sudo rm /var/www/html/test.html
[aamishina@aamishina ~]$
```

Рис. 2.20: Удаляем привязку http_port_t к 81, выполняем проверку, удаляем файл test.html

3 Выводы

В ходе выполнения данной лабораторной работы, я развила навыки администрирования ОС Linux, получила первое практическое знакомство с технологией SELinux¹, а также проверила работу SELinx на практике совместно с веб-сервером Apache.

Список литературы

1. Кулябов Д.С., Королькова А.В., Геворкян М.Н. Информационная безопасность компьютерных сетей. Лабораторные работы, учебное пособие. Москва: РУДН, 2015. 64 с.