

# Отчет к лабораторной работе №6

---

## Common information

discipline: Основы информационной безопасности

group: НПМбд-02-21

author: Бабина Ю. О.

---

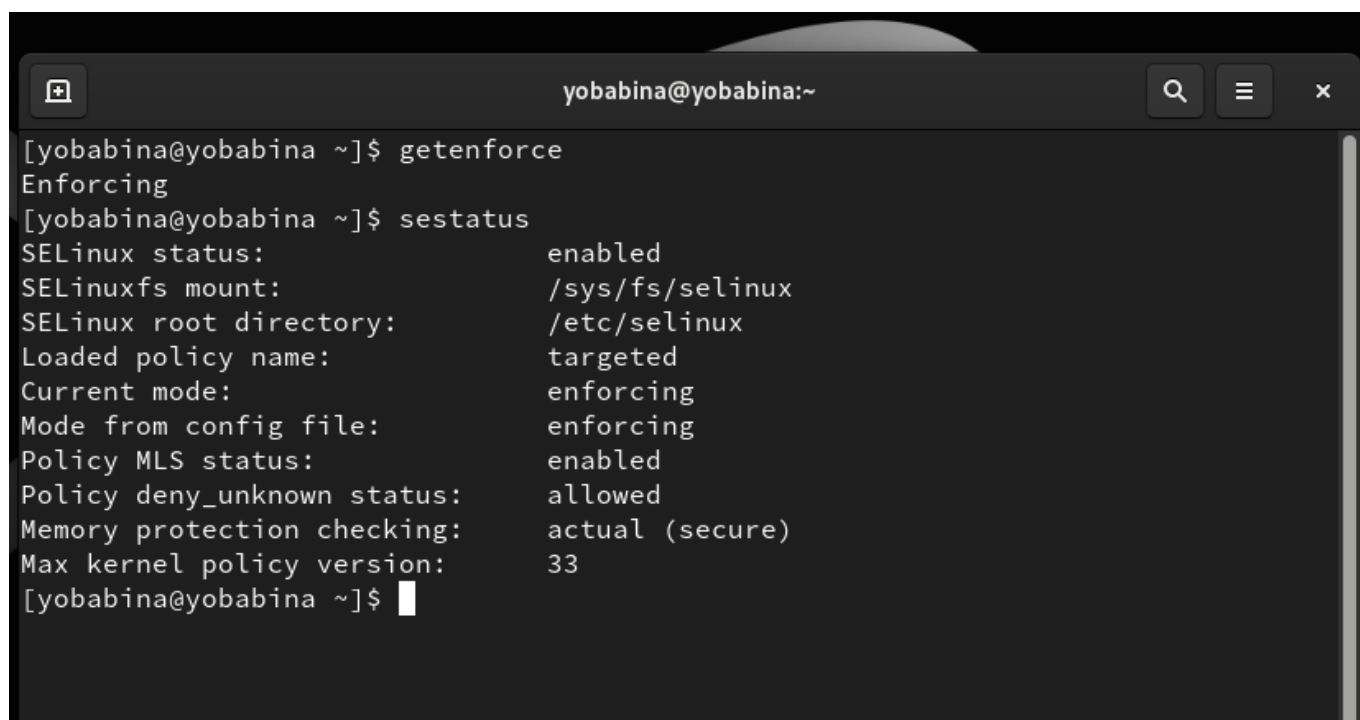
## Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1.

Проверить работу SELinux на практике совместно с веб-сервером Apache.

## Выполнение работы

Войдем в систему с полученными учётными данными и убедимся, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`.

A screenshot of a terminal window with a dark background. The window title is 'yobabina@yobabina:~'. The terminal shows the execution of two commands: 'getenforce' and 'sestatus'. The output of 'getenforce' is 'Enforcing'. The output of 'sestatus' is a multi-line status report for SELinux.

```
[yobabina@yobabina ~]$ getenforce
Enforcing
[yobabina@yobabina ~]$ 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
[yobabina@yobabina ~]$
```

Обратимся с помощью браузера к веб-серверу, запущенному на нашем компьютере, и убедимся, что последний работает

```

yobabina@yobabina:~$ sudo service httpd status
[sudo] password for yobabina:
Redirecting to /bin/systemctl status httpd.service
○ httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: d>
   Active: inactive (dead)
   Docs: man:httpd.service(8)
lines 1-4/4 (END)
[yobabina@yobabina ~]$ sudo service httpd start
Redirecting to /bin/systemctl start httpd.service
[yobabina@yobabina ~]$ sudo 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: d>
   Active: active (running) since Thu 2024-10-10 20:53:09 MSK; 4s ago
   Docs: man:httpd.service(8)
  Main PID: 115378 (httpd)
    Status: "Started, listening on: port 80"
     Tasks: 177 (limit: 23037)
    Memory: 22.1M
       CPU: 92ms
    CGroup: /system.slice/httpd.service
            └─115378 /usr/sbin/httpd -DFOREGROUND
              └─115379 /usr/sbin/httpd -DFOREGROUND

```

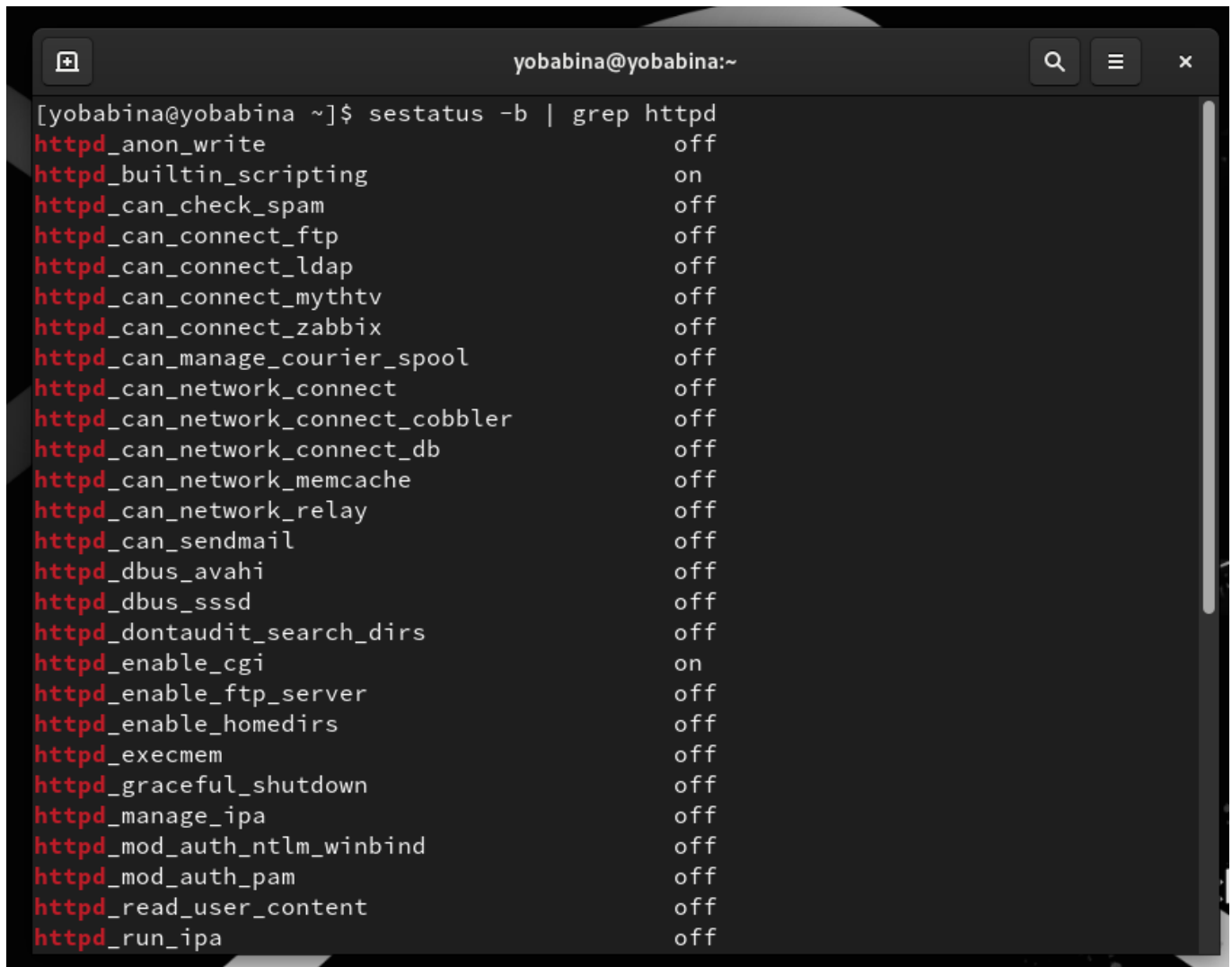
Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности:

```

yobabina@yobabina:~$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0  root      115378  0.1  0.3  20364 11520 ?
Ss  20:53   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache   115379  0.0  0.1  22096  7264 ?
S   20:53   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache   115381  0.0  0.3 1112656 13464 ?
Sl  20:53   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache   115384  0.0  0.2 981520 11212 ?
Sl  20:53   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache   115385  0.0  0.2 981520 11212 ?
Sl  20:53   0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 yobabina 115584 0.0  0.0 2
21664 2304 pts/0 S+ 20:53   0:00 grep --color=auto httpd
[yobabina@yobabina ~]$ ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      115378 ?           00:00:00 httpd
system_u:system_r:httpd_t:s0      115379 ?           00:00:00 httpd
system_u:system_r:httpd_t:s0      115381 ?           00:00:00 httpd
system_u:system_r:httpd_t:s0      115384 ?           00:00:00 httpd
system_u:system_r:httpd_t:s0      115385 ?           00:00:00 httpd
[yobabina@yobabina ~]$

```

Посмотрим текущее состояние переключателей SELinux для Apache с помощью команды: "sestatus -bigrep httpd".



```
[yobabina@yobabina ~]$ 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
```

Посмотрим статистику по политике с помощью команды `seinfo`, также определим множество пользователей, ролей, типов:

```
yobabina@yobabina:~$ 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:        5145     Attributes:       259
Users:        8       Roles:           15
Booleans:     356     Cond. Expr.:    388
Allow:        65504   Neverallow:      0
Auditallow:   176     Dontaudit:      8682
Type_trans:   271770  Type_change:     94
Type_member:  37      Range_trans:    5931
Role allow:   40      Role_trans:     417
Constraints:  70      Validatetrans:  0
MLS Constrain: 72     MLS Val. Tran:  0
Permissives:  4      Polcap:         6
Defaults:     7      Typebounds:     0
Allowxperm:   0      Neverallowxperm: 0
Auditallowxperm: 0    Dontauditxperm: 0
Ibendportcon: 0      Ibkeycon:       0
Initial SIDs: 27     Fs_use:         35
Genfscon:     109    Portcon:        665
Netifcon:     0      Nodecon:        0

[yobabina@yobabina ~]$
```

Определим тип файлов и поддиректорий, находящихся в директории /var/www, /var/www/html. Также определим круг пользователей, которым разрешено создание файлов в директории /var/www/html.

```
yobabina@yobabina:~$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Aug 8 19
:30 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Aug 8 19
:30 html

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

[yobabina@yobabina ~]$ ls -lah /var/www
total 4.0K
drwxr-xr-x. 4 root root 33 Oct 10 19:38 .
drwxr-xr-x. 21 root root 4.0K Oct 10 19:38 ..
drwxr-xr-x. 2 root root 6 Aug 8 19:30 cgi-bin
drwxr-xr-x. 2 root root 6 Aug 8 19:30 html

[yobabina@yobabina ~]$
```

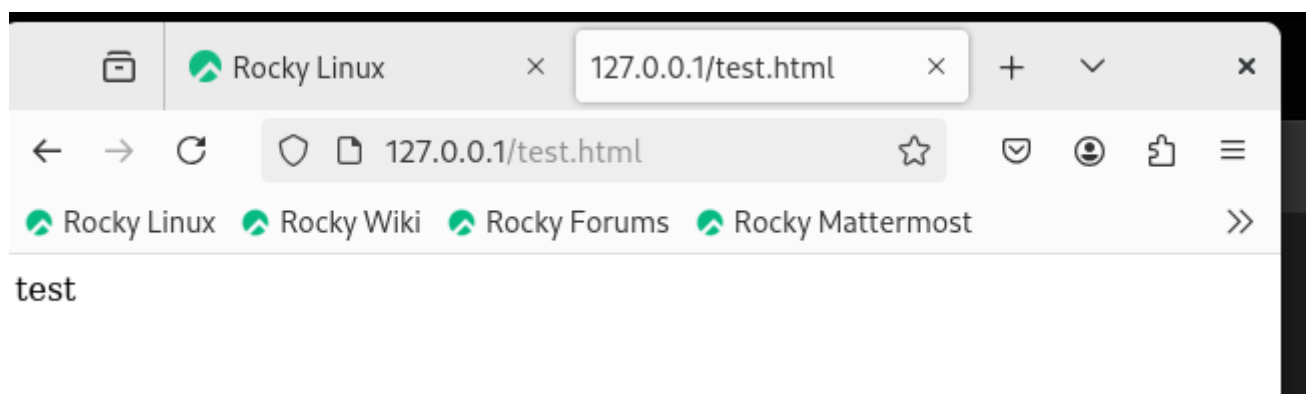
Создадим от имени суперпользователя html-файл /var/www/html/test.html:

```
yobabina@yobabina:~  
[yobabina@yobabina ~]$ sudo nano /var/www/html/test.html  
[yobabina@yobabina ~]$ cat /var/www/html/test.html  
<html>  
<body>test</body>  
</html>  
[yobabina@yobabina ~]$
```

Проверим контекст созданного файла:

```
yobabina@yobabina:~  
[yobabina@yobabina ~]$ ls -lZ /var/www/html  
total 4  
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 10 20:58 test.html  
[yobabina@yobabina ~]$
```

Обратимся к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>.



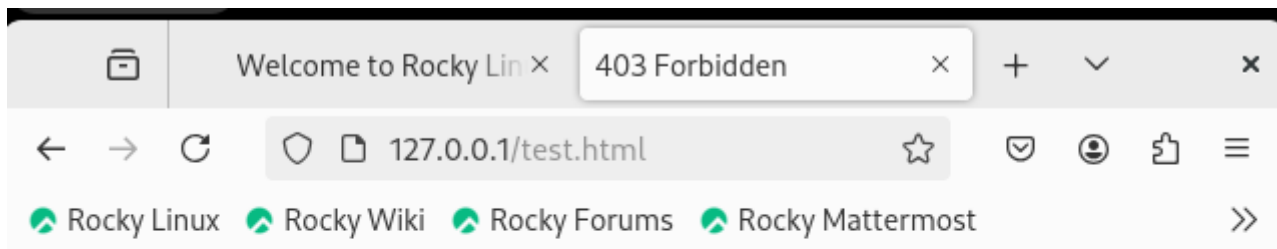
Изучим справку man httpd\_selinux. Проверим контекст файла:

```
yobabina@yobabina:~  
[yobabina@yobabina ~]$ ls -lZ /var/www/html/test.html  
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 10 20:58 /var/www/html/test.html  
[yobabina@yobabina ~]$
```

Изменим 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`:

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

Попробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес:  
"http://127.0.0.1/test.html":



# Forbidden

You don't have permission to access this resource.

Просмотрим log-файлы веб-сервера Apache и системный лог-файл:

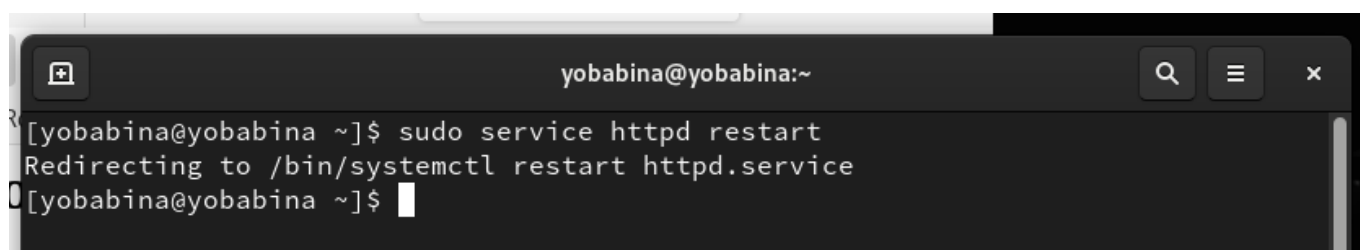
```
yobabina@yobabina:~$ ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct 10 20:58 /var/www/html/test.html
[yobabina@yobabina ~]$ sudo tail /var/log/httpd/access_log
127.0.0.1 - - [10/Oct/2024:21:00:19 +0300] "GET /test.html HTTP/1.1" 200 33 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
127.0.0.1 - - [10/Oct/2024:21:00:20 +0300] "GET /favicon.ico HTTP/1.1" 404 196 "
http://127.0.0.1/test.html" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/201
00101 Firefox/128.0"
127.0.0.1 - - [10/Oct/2024:21:03:43 +0300] "GET /test.html HTTP/1.1" 403 199 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
127.0.0.1 - - [10/Oct/2024:21:04:12 +0300] "GET /test.html HTTP/1.1" 403 199 "-"
"Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
[yobabina@yobabina ~]$ sudo tail /var/log/messages
Oct 10 21:04:14 yobabina setroubleshoot[116711]: failed to retrieve rpm info for
path '/var/www/html/test.html':
Oct 10 21:04:14 yobabina systemd[1]: Started dbus-1.1-org.fedoraproject.Setroub
leshootPrivileged@1.service.
Oct 10 21:04:17 yobabina setroubleshoot[116711]: SELinux is preventing /usr/sbin
/httpd from getattr access on the file /var/www/html/test.html. For complete SEL
inux messages run: sealert -l b84fac53-3ae2-49c9-a031-49b5aeb2df0b
Oct 10 21:04:17 yobabina setroubleshoot[116711]: SELinux is preventing /usr/sbin
/httpd from getattr access on the file /var/www/html/test.html.#012#012***** Pl
ugin restorecon (92.2 confidence) suggests *****#012#012If
you want to fix the label. #012/var/www/html/test.html default label should be h
ttpd_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 whi
ch case try to change the following command accordingly.#012Do#012# /sbin/restor
econ -v /var/www/html/test.html#012#012***** Plugin public_content (7.83 confid
```

Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81, для этого в файле /etc/httpd/httpd.conf найдем строчку Listen 80 и заменим её на Listen 81:



```
yobabina@yobabina:~ — sudo 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
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

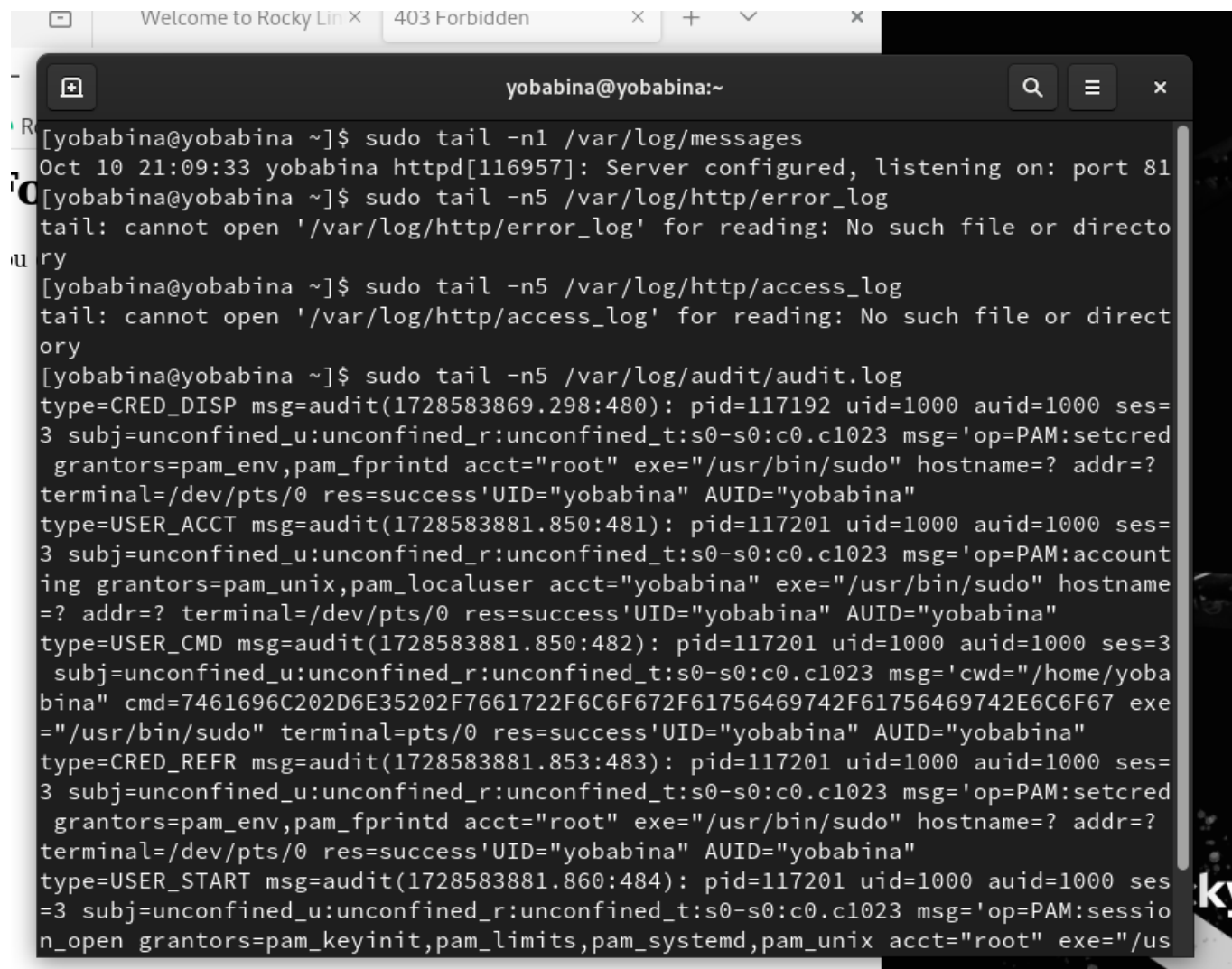
Выполним перезапуск веб-сервера Apache:



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

Проанализируем лог-файлы:





```
yobabina@yobabina:~  
[yobabina@yobabina ~]$ sudo tail -n1 /var/log/messages  
Oct 10 21:09:33 yobabina httpd[116957]: Server configured, listening on: port 81  
[yobabina@yobabina ~]$ sudo tail -n5 /var/log/http/error_log  
tail: cannot open '/var/log/http/error_log' for reading: No such file or directory  
[yobabina@yobabina ~]$ sudo tail -n5 /var/log/http/access_log  
tail: cannot open '/var/log/http/access_log' for reading: No such file or directory  
[yobabina@yobabina ~]$ sudo tail -n5 /var/log/audit/audit.log  
type=CRED_DISP msg=audit(1728583869.298:480): pid=117192 uid=1000 auid=1000 ses=  
3 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred  
grantors=pam_env,pam_fprintd acct="root" exe="/usr/bin/sudo" hostname=? addr=?  
terminal=/dev/pts/0 res=success'UID="yobabina" AUID="yobabina"  
type=USER_ACCT msg=audit(1728583881.850:481): pid=117201 uid=1000 auid=1000 ses=  
3 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:account  
ing grantors=pam_unix,pam_localuser acct="yobabina" exe="/usr/bin/sudo" hostname  
=? addr=? terminal=/dev/pts/0 res=success'UID="yobabina" AUID="yobabina"  
type=USER_CMD msg=audit(1728583881.850:482): pid=117201 uid=1000 auid=1000 ses=3  
subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='cwd="/home/yoba  
bina" cmd=7461696C202D6E35202F7661722F6C6F672F61756469742F61756469742E6C6F67 exe  
="/usr/bin/sudo" terminal=pts/0 res=success'UID="yobabina" AUID="yobabina"  
type=CRED_REFR msg=audit(1728583881.853:483): pid=117201 uid=1000 auid=1000 ses=  
3 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred  
grantors=pam_env,pam_fprintd acct="root" exe="/usr/bin/sudo" hostname=? addr=?  
terminal=/dev/pts/0 res=success'UID="yobabina" AUID="yobabina"  
type=USER_START msg=audit(1728583881.860:484): pid=117201 uid=1000 auid=1000 ses  
=3 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session  
_open grantors=pam_keyinit,pam_limits,pam_systemd,pam_unix acct="root" exe="/us
```

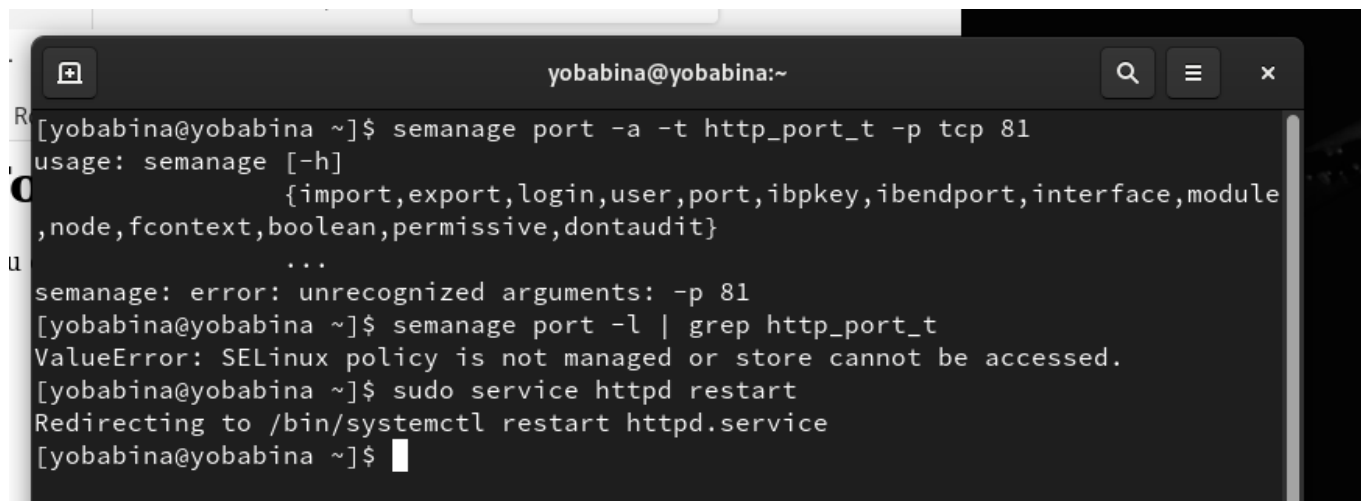
Выполним команду

```
semanage port -a -t http_port_t -p tcp 81
```

После этого проверим список портов командой

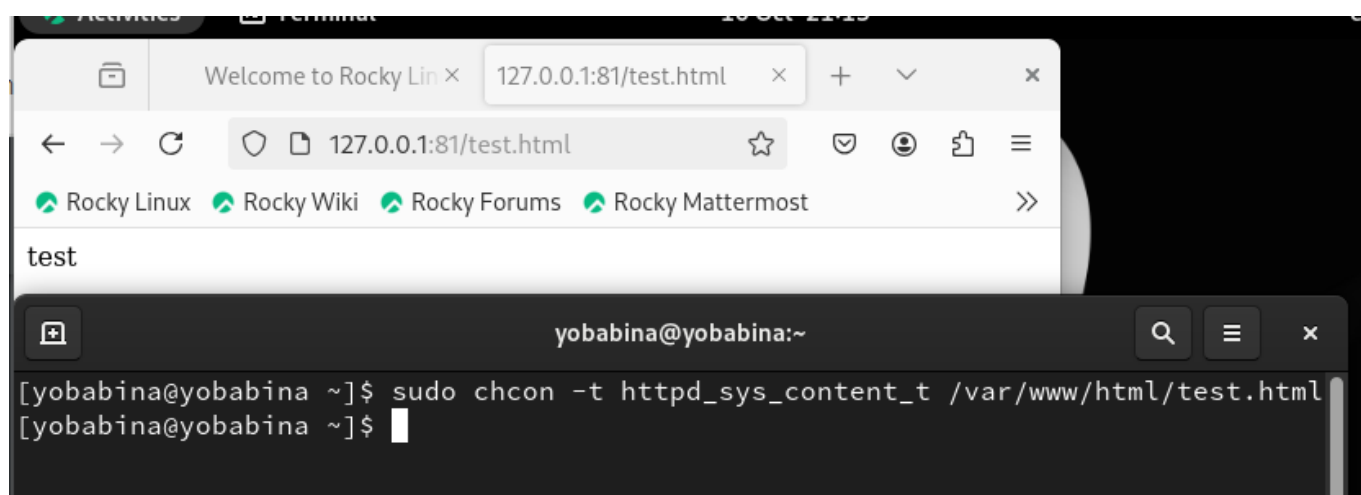
```
semanage port -l | grep http_port_t
```

После чего попробуем запустить веб-сервер Apache ещё раз:



```
yobabina@yobabina:~$ semanage port -a -t http_port_t -p tcp 81
usage: semanage [-h]
               {import,export,login,user,port,ibpkey,ibendport,interface,module
               ,node,fcontext,boolean,permissive,dontaudit}
               ...
semanage: error: unrecognized arguments: -p 81
[yobabina@yobabina ~]$ semanage port -l | grep http_port_t
ValueError: SELinux policy is not managed or store cannot be accessed.
[yobabina@yobabina ~]$ sudo service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[yobabina@yobabina ~]$
```

Вернем контекст `httpd_sys_content_t` к файлу `/var/www/html/ test.html`:



```
127.0.0.1:81/test.html
test

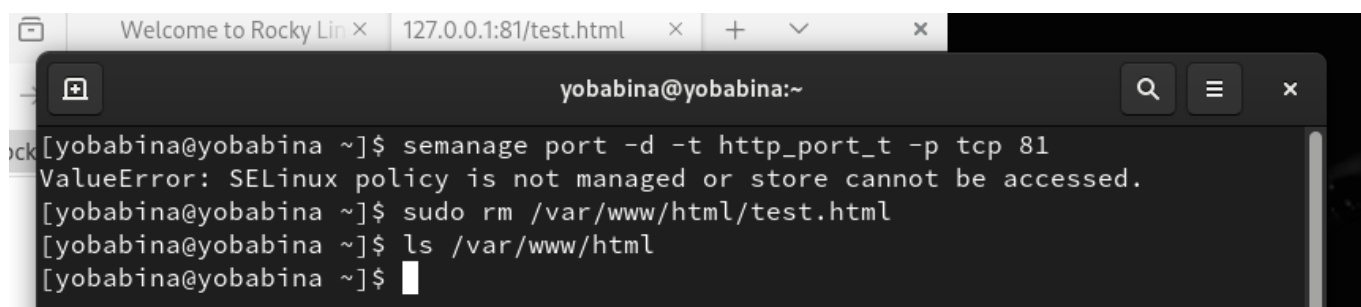
[yobabina@yobabina ~]$ sudo chcon -t httpd_sys_content_t /var/www/html/test.html
[yobabina@yobabina ~]$
```

Исправим конфигурационный файл `apache`, вернув `Listen 80`.



```
[yobabina@yobabina ~]$ cat /etc/httpd/conf/httpd.conf | grep Listen
# Listen: Allows you to bind Apache to specific IP addresses and/or
# Change this to Listen on a specific IP address, but note that if
#Listen 12.34.56.78:80
Listen 80
[yobabina@yobabina ~]$
```

Удалим привязку `http_port_t` к 81 порту. Затем удалим файл `/var/www/html/test.html`:



```
[yobabina@yobabina ~]$ semanage port -d -t http_port_t -p tcp 81
ValueError: SELinux policy is not managed or store cannot be accessed.
[yobabina@yobabina ~]$ sudo rm /var/www/html/test.html
[yobabina@yobabina ~]$ ls /var/www/html
[yobabina@yobabina ~]$
```

## Вывод

В рамках выполнения данной лабораторной работы я развила навыки администрирования ОС Linux. Получила первое практическое знакомство с технологией SELinux1.

Проверила работу SELinux на практике совместно с веб-сервером Apache.

## Список литературы

- <https://habr.com/ru/articles/469667/>
- <https://habr.com/ru/companies/kingservers/articles/209644/>
- <https://rockylinux.org/>