

# **Отчёт по лабораторной работе №6**

**Знакомство с SELinux**

Боровиков Даниил Александрович НПИбд-01-22

# Содержание

|          |                                       |           |
|----------|---------------------------------------|-----------|
| <b>1</b> | <b>Цель работы</b>                    | <b>4</b>  |
| <b>2</b> | <b>Выполнение лабораторной работы</b> | <b>5</b>  |
| 2.1      | Подготовка . . . . .                  | 5         |
| 2.2      | Изучение механики SetUID . . . . .    | 5         |
| <b>3</b> | <b>Выводы</b>                         | <b>13</b> |
|          | <b>Список литературы</b>              | <b>14</b> |

# List of Figures

|     |  |    |
|-----|--|----|
| 2.1 | запуск http . . . . .                              | 6  |
| 2.2 | контекст безопасности http . . . . .               | 7  |
| 2.3 | переключатели SELinux для http . . . . .           | 7  |
| 2.4 | создание html-файла и доступ по http . . . . .     | 9  |
| 2.5 | ошибка доступа после изменения контекста . . . . . | 10 |
| 2.6 | лог ошибок . . . . .                               | 10 |
| 2.7 | переключение порта . . . . .                       | 11 |
| 2.8 | доступ по http на 81 порт . . . . .                | 12 |

# 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

## 2 Выполнение лабораторной работы

### 2.1 Подготовка

1. Установили httpd
2. Задали имя сервера
3. Открыли порты для работы с протоколом http

### 2.2 Изучение механики SetUID

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`.
2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает: `service httpd status` или `/etc/rc.d/init.d/httpd status` Если не работает, запустите его так же, но с параметром `start`.

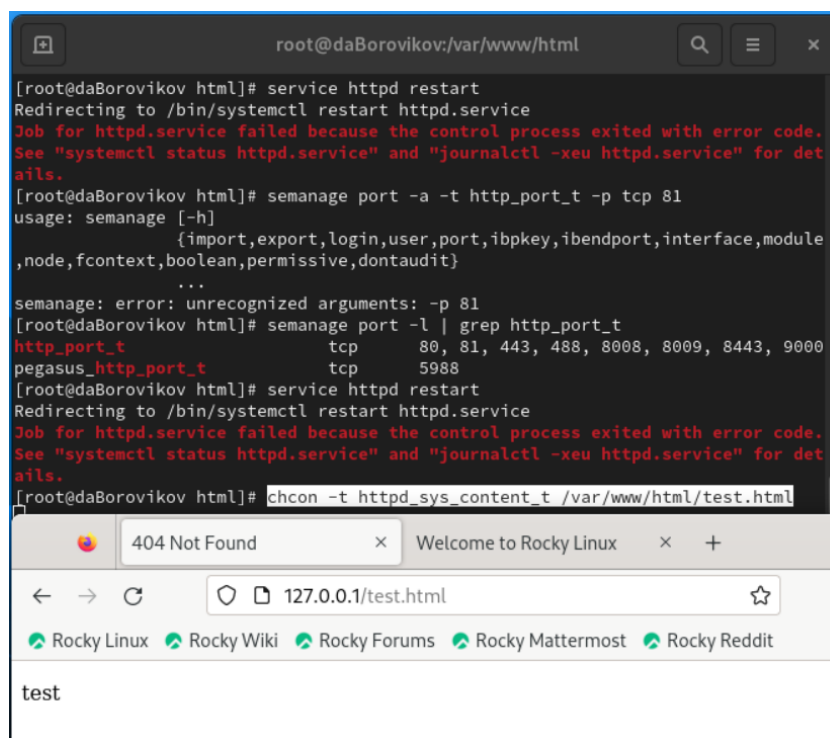


Figure 2.1: запуск http

3. Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт. Например, можно использовать команду `ps auxZ | grep httpd` или `ps -eZ | grep httpd`

```

root@daBorovikov:~
[daborovikov@daBorovikov ~]$ su -
Password:
su: Authentication failure
[daborovikov@daBorovikov ~]$ su -
Password:
[root@daBorovikov ~]# nano /etc/httpd/conf/httpd.conf
[root@daBorovikov ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@daBorovikov ~]# service httpd status
The service command supports only basic LSB actions (start, stop, restart, try-r
estart, reload, reload-or-restart, try-reload-or-restart, force-reload, status,
condrestart). For other actions, please try to use systemctl.
[root@daBorovikov ~]# service httpd status
Redirecting to /bin/systemctl status httpd.service
• httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: di
   Active: active (running) since Sat 2024-04-27 18:47:10 MSK; 1h 0min ago
     Docs: man:httpd.service(8)
    Main PID: 76468 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes
     Tasks: 213 (limit: 24605)
    Memory: 41.8M
       CPU: 2.618s
    CGroup: /system.slice/httpd.service
            └─76468 /usr/sbin/httpd -DFOREGROUND
              └─76469 /usr/sbin/httpd -DFOREGROUND
                └─76470 /usr/sbin/httpd -DFOREGROUND
                  └─76471 /usr/sbin/httpd -DFOREGROUND
                    └─76472 /usr/sbin/httpd -DFOREGROUND

```

Figure 2.2: контекст безопасности http

- Посмотрите текущее состояние переключателей SELinux для Apache с по-  
мощью команды `sestatus -bigrep httpd` Обратите внимание, что многие из  
них находятся в положении «off».

```

[root@daBorovikov ~]# ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      76468 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      76469 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      76470 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      76471 ?          00:00:00 httpd
system_u:system_r:httpd_t:s0      76472 ?          00:00:00 httpd
[root@daBorovikov ~]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0      root      76468  0.0  0.2 20340 11368 ?
  Ss  18:47  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache    76469  0.0  0.1 21676 7448 ?
  S   18:47  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache    76470  0.0  0.3 2324680 13060 ?
  Sl  18:47  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache    76471  0.0  0.4 2324680 17144 ?
  Sl  18:47  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0      apache    76472  0.0  0.4 2521352 19188 ?
  Sl  18:47  0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 172969 0.0  0.0 22179
6 2312 pts/0 S+  19:49  0:00 grep --color=auto httpd
[root@daBorovikov ~]#

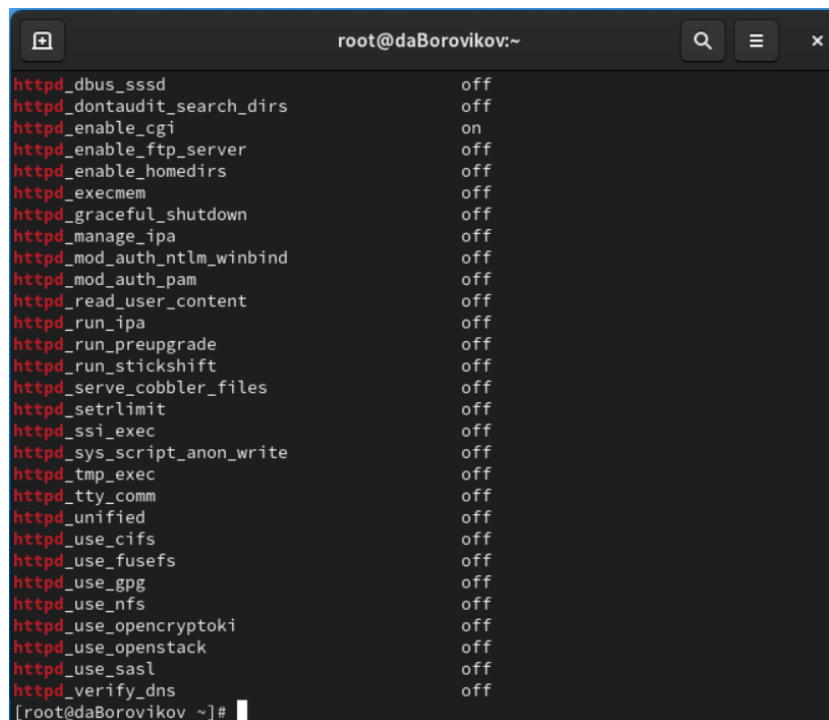
```

Figure 2.3: переключатели SELinux для http

- Посмотрите статистику по политике с помощью команды `seinfo`, также  
определите множество пользователей, ролей, типов.

6. Определите тип файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды `ls -lZ /var/www`. В поддиректориях могут располагаться системные скрипты и контент для `http`.
7. Определите тип файлов, находящихся в директории `/var/www/html`: `ls -lZ /var/www/html`. В директории изначально нет файлов.
8. Определите круг пользователей, которым разрешено создание файлов в директории `/var/www/html`. Создавать файлы может только `root`.
9. Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) `html`-файл `/var/www/html/test.html` следующего содержания: `Test`
10. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории `/var/www/html`.
11. Обратитесь к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Убедитесь, что файл был успешно отображён.



A terminal window titled 'root@daBorovikov:~' showing a list of httpd configuration options and their status. The options are listed in red text, and their status is in white text. The options are: 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). The prompt is [root@daBorovikov ~]#.

```
root@daBorovikov:~
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
[root@daBorovikov ~]#
```

Figure 2.4: создание html-файла и доступ по http

12. Изучите справку `man httpd_selinux` и выясните, какие контексты файлов определены для `httpd`. Сопоставьте их с типом файла `test.html`. Проверить контекст файла можно командой `ls -Z. ls -Z /var/www/html/test.html`. Основным контекстом является `httpd_sys_content_t`, его мы и увидели в выводе команды.
13. Измените 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` `ls -Z /var/www/html/test.html` После этого проверьте, что контекст поменялся.
14. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Вы должны получить сообщение об ошибке: `Forbidden You don't have permission to access /test.html on this server.` При изменении контекста файл стал считаться чужим для `http` и программа не может его прочитать.

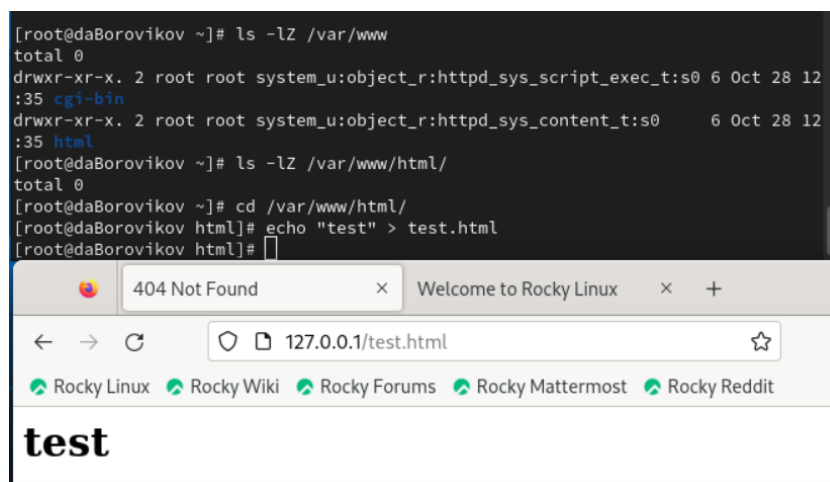


Figure 2.5: ошибка доступа после изменения контекста

15. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? `ls -l /var/www/html/test.html` Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: `tail /var/log/messages` Если в системе окажутся запущенными процессы `setroubleshootd` и `audtd`, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле `/var/log/audit/audit.log`. Проверьте это утверждение самостоятельно.

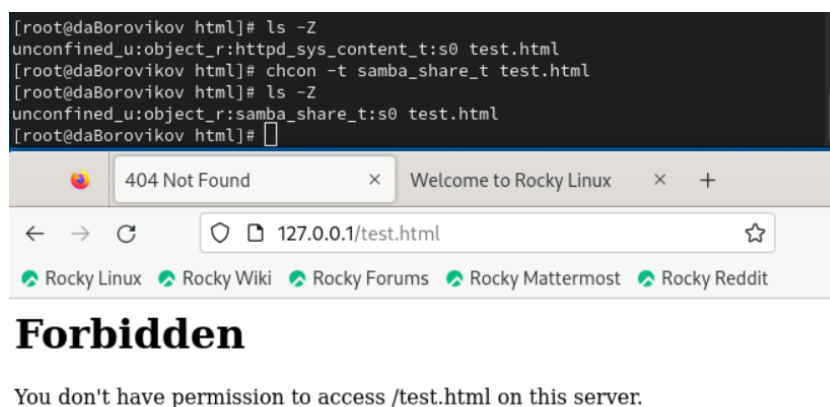


Figure 2.6: лог ошибок

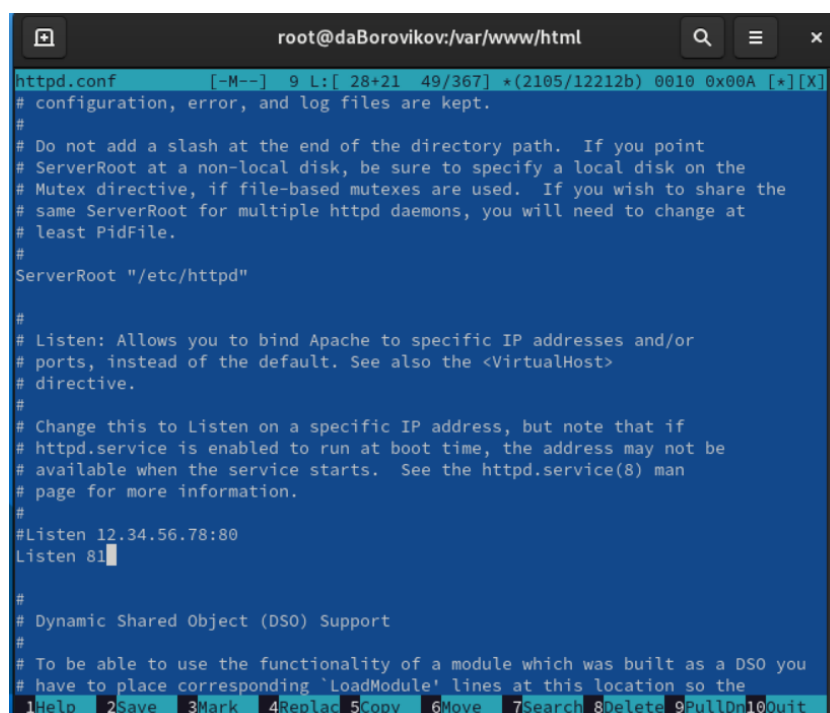
16. Попробуйте запустить веб-сервер Apache на прослушивание TCP-порта 81 (а не 80, как рекомендует IANA и прописано в `/etc/services`). Для этого

в файле `/etc/httpd/httpd.conf` найдите строчку `Listen 80` и замените её на `Listen 81`.

```
[root@daBorovikov html]# ls -l test.html
-rw-r--r--. 1 root root 33 Apr 27 20:03 test.html
[root@daBorovikov html]# tail /var/log/messages
Apr 27 19:54:20 daBorovikov systemd[1]: packagekit.service: Main process exited,
code=dumped, status=11/SEGV
Apr 27 19:54:20 daBorovikov systemd[1]: packagekit.service: Failed with result '
core-dump'.
Apr 27 19:54:20 daBorovikov systemd[1]: packagekit.service: Consumed 15.132s CPU
time.
Apr 27 19:54:22 daBorovikov systemd[1]: Starting PackageKit Daemon...
Apr 27 19:54:22 daBorovikov systemd[1]: Started PackageKit Daemon.
Apr 27 19:55:29 daBorovikov cupsd[137817]: REQUEST localhost - - "POST / HTTP/1.
1" 200 190 Renew-Subscription successful-ok
Apr 27 19:57:57 daBorovikov systemd[2453]: Started Application launched by gnome
-shell.
Apr 27 19:57:59 daBorovikov rtkit-daemon[1056]: Successfully made thread 173442
of process 173261 (/usr/lib64/firefox/firefox) owned by '1000' RT at priority 10
.
Apr 27 20:00:33 daBorovikov firefox.desktop[173261]: Crash Annotation GraphicsCr
iticalError: |[0][GFX1-]: RenderCompositorSWGL failed mapping default framebuffe
r, no dt (t=155.8) [GFX1-]: RenderCompositorSWGL failed mapping default framebuf
fer, no dt
Apr 27 20:01:01 daBorovikov root[173812]: This message is written at Sat Apr 27
08:01:01 PM MSK 2024
[root@daBorovikov html]#
```

Figure 2.7: переключение порта

17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему? Сбой не происходит, порт 81 уже вписан в разрешенные
18. Проанализируйте лог-файлы: `tail -nl /var/log/messages` Просмотрите фай-  
лы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и  
выясните, в каких файлах появились записи.
19. Выполните команду `semanage port -a -t http_port_t -p tcp 81` После этого про-  
верьте список портов командой `semanage port -l | grep http_port_t` Убедитесь,  
что порт 81 появился в списке.
20. Попробуйте запустить веб-сервер Apache ещё раз.
21. Верните контекст `httpd_sys_content_t` к файлу `/var/www/html/ test.html`:  
`chcon -t httpd_sys_content_t /var/www/html/test.html` После этого попро-  
буйте получить доступ к файлу через веб-сервер, введя в браузере адрес  
`http://127.0.0.1:81/test.html`. Вы должны увидеть содержимое файла — слово  
«test».



```
root@daBorovikov:/var/www/html
httpd.conf [-M--] 9 L: [ 28+21 49/367] *(2105/12212b) 0010 0x00A [*][X]
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path. If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used. If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"
#
# 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
1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn 10Quit
```

Figure 2.8: доступ по http на 81 порт

22. Исправьте обратно конфигурационный файл apache, вернув Listen 80.
23. Удалите привязку http\_port\_t к 81 порту: `semanage port -d -t http_port_t -p tcp 81` и проверьте, что порт 81 удалён.
24. Удалите файл `/var/www/html/test.html`: `rm /var/www/html/test.html`

## **3 Выводы**

В процессе выполнения лабораторной работы мною были получены базовые навыки работы с технологией seLinux.

# Список литературы

1. SELinux в CentOS
2. Веб-сервер Apache