

# Лабораторная работа №6

Мандатное разграничение прав в Linux

---

Легиньких Г.А.

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Легиньких Галина Андреевна
- НФИбд-02-21
- Российский университет дружбы народов
- 1032216447@pfur.ru
- <https://github.com/galeginkikh>

## Выполнение

---

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux. Проверить работу SELinux на практике совместно с веб-сервером Apache.

## 1. Скачала httpd.

```
[galeginkikh@galeginkikh ~]$ su
Password:
[root@galeginkikh galeginkikh]# yum install httpd
Last metadata expiration check: 1 day, 16:49:54 ago on Wed 18 Sep 2024 09:15:27 PM
MSK.
Dependencies resolved.
=====
Package           Arch    Version                               Repo                Size
=====
Installing:
  httpd            x86_64  2.4.37-65.module+el8.10.0+1842+4a9649e8.2 appstream 1.4 M
Installing dependencies:
  apr              x86_64  1.6.3-12.el8                        appstream 128 k
  apr-util         x86_64  1.6.1-9.el8                         appstream 105 k
  httpdfilesystem
```

Рис. 1: install httpd

2. В конфигурационном файле `/etc/httpd/httpd.conf` задала параметр `ServerName`.  
Отключила пакетный фильтр.

```
[root@galeginkikh galeginkikh]# cd /etc/httpd
[root@galeginkikh httpd]# echo "ServerName test.ru" >> httpd.conf
[root@galeginkikh httpd]# iptables -F
[root@galeginkikh httpd]# iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT
Bad argument `iptables'
Try `iptables -h' or 'iptables --help' for more information.
[root@galeginkikh httpd]# iptables -P INPUT ACCEPT
[root@galeginkikh httpd]# iptables -P OUTPUT ACCEPT
[root@galeginkikh httpd]#
```

Рис. 2: Пакетные фильтры

3. Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted.

```
[root@galeginkikh httpd]# getenforce
Enforcing
[root@galeginkikh httpd]# 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
[root@galeginkikh httpd]#
```

Рис. 3: SELinux



4. Обратилась с помощью браузера к веб-серверу, запущенному на компьютере, и убедилась, что последний работает. Он не работает, запустила его так же, но с параметром start.

```
[root@galeginkikh httpd]# 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 preset: enabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
lines 1-4/4 (END)
```

Рис. 4: Сервер\_1

```
[root@galeginkikh httpd]# service httpd start  
Redirecting to /bin/systemctl start httpd.service  
[root@galeginkikh httpd]#
```

Рис. 5: Сервер\_2

5. Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности.

```
without options, show SELinux status.  
[root@galeginkikh httpd]# ps auxZ | grep httpd  
system_u:system_r:httpd_t:s0 root 41502 0.0 0.1 265184 11448 ? S  
s 14:14 0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 41511 0.0 0.1 269888 8552 ? S  
l 14:14 0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 41512 0.0 0.2 1786432 12168 ? S  
l 14:14 0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 41513 0.0 0.2 1917560 14220 ? S  
l 14:14 0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 41514 0.0 0.2 1786432 12168 ? S  
l 14:14 0:00 /usr/sbin/httpd -DFOREGROUND  
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 41744 0.0 0.0 222012 1  
212 pts/0 S+ 14:15 0:00 grep --color=auto httpd
```

Рис. 6: Веб-сервер Apache

6. Посмотрела текущее состояние переключателей SELinux для Apache. Обратила внимание, что многие из них находятся в положении «off».

```
[root@galeginkikh httpd]# 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_network_connect off
httpd_can_network_connect_cobbler off
httpd_can_network_connect_db off
httpd_can_network_memcache off
httpd_can_network_redis 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
```

7. Посмотрела статистику по политике с помощью команды `seinfo`, также определила множество пользователей, ролей, типов.

```
[root@galeginkikh httpd]# seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:          31 (MLS enabled)
Target Policy:           selinux
Handle unknown classes:  allow

Classes:          132      Permissions:        464
Sensitivities:    1       Categories:        1024
Types:           5015     Attributes:         258
Users:           8       Roles:             15
Booleans:        349     Cond. Expr.:       399
Allow:           116272   Neverallow:         0
Auditallow:      172     Dontaudit:         10529
Type_trans:      262670   Type_change:        94
Type_member:     37      Range_trans:       5989
Role_allow:      40      Role_trans:        421
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:             34
Genfscon:        107    Portcon:            649
Netifcon:        0      Nodecon:            0
[root@galeginkikh httpd]#
```

Рис. 8: `seinfo`

8. Определила тип файлов и поддиректорий, находящихся в директории /var/www.

```
[root@galeginkikh httpd]# ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Aug 12 11:1
4 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Aug 12 11:1
4 html
```

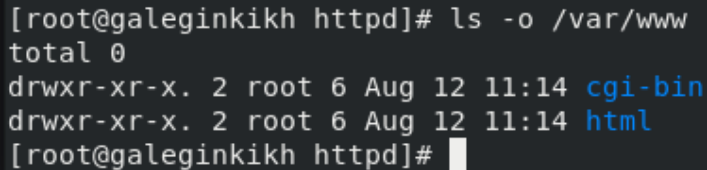
Рис. 9: Тип файлов и поддиректорий в /var/www

9. Определила тип файлов, находящихся в директории `/var/www/html`.

```
[root@galeginkikh httpd]# ls -lZ /var/www/html  
total 0
```

Рис. 10: Тип файлов и поддиректорий в `/var/www/html`

10. Определила круг пользователей, которым разрешено создание файлов в директории /var/www/html.

A terminal window with a dark background and light gray text. The prompt is [root@galeginkikh httpd]#. The command ls -o /var/www is entered. The output shows 'total 0' followed by two lines of file permissions: 'drwxr-xr-x. 2 root 6 Aug 12 11:14 cgi-bin' and 'drwxr-xr-x. 2 root 6 Aug 12 11:14 html'. The text 'cgi-bin' and 'html' are highlighted in blue. The prompt [root@galeginkikh httpd]# is followed by a white cursor block.

```
[root@galeginkikh httpd]# ls -o /var/www
total 0
drwxr-xr-x. 2 root 6 Aug 12 11:14 cgi-bin
drwxr-xr-x. 2 root 6 Aug 12 11:14 html
[root@galeginkikh httpd]#
```

Рис. 11: Права на создание файлов



11. Создала от имени суперпользователя html-файл /var/www/html/test.html.

```
[root@galeginkikh httpd]# touch /var/www/html test.html
[root@galeginkikh httpd]# cd ..
[root@galeginkikh etc]# cd
[root@galeginkikh ~]# ls
anaconda-ks.cfg  initial-setup-ks.cfg
[root@galeginkikh ~]# echo '<html>' >> /var/www/html/test.html
[root@galeginkikh ~]# echo '<body>test</body>' >> /var/www/html/test.html
[root@galeginkikh ~]# echo '</html>' >> /var/www/html/test.html
[root@galeginkikh ~]#
```

Рис. 12: test.html

12. Проверила контекст созданного файла.

```
[root@galeginkikh ~]# ls -lZ /var/www/html/test.html  
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Sep 20 14:  
26 /var/www/html/test.html
```

Рис. 13: Контекст test.html

13. Обратилась к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`.

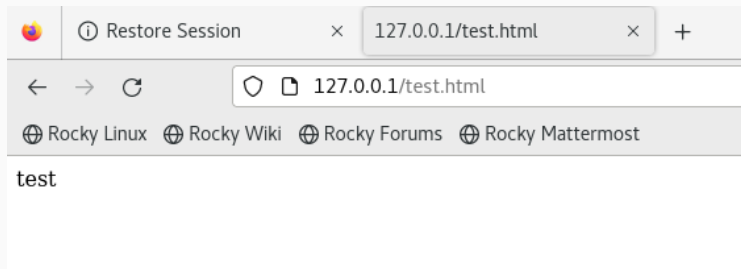


Рис. 14: Веб-сервер

14. Изучила справку `man httpd_selinux` и выяснила, какие контексты файлов определены для `httpd`.

```
HTTPD(8)                                httpd                                HTTPD(8)

NAME
    httpd - Apache Hypertext Transfer Protocol Server

SYNOPSIS
    httpd [ -d serverroot ] [ -f config ] [ -C directive ] [ -c directive ]
    [ -D parameter ] [ -e level ] [ -E file ] [ -k start|restart|graceful-stop|graceful-stop ] [ -h ] [ -l ] [ -L ] [ -S ] [ -t ] [ -v ] [ -V ]
    [ -X ] [ -M ] [ -T ]

    On Windows systems, the following additional arguments are available:

    httpd [ -k install|config|uninstall ] [ -n name ] [ -w ]

SUMMARY
    httpd is the Apache HyperText Transfer Protocol (HTTP) server program.
    It is designed to be run as a standalone daemon process. When used like
    this it will create a pool of child processes or threads to handle
    requests.

    In general, httpd should not be invoked directly, but rather should be
    invoked via apachectl on Unix-based systems or as a service on Windows
    NT, 2000 and XP and as a console application on Windows 9x and ME.
```

Рис. 15: `man`

15. Изменила контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на любой другой, к которому процесс `httpd` не должен иметь доступа.

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

Рис. 16: Изменение контекста

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

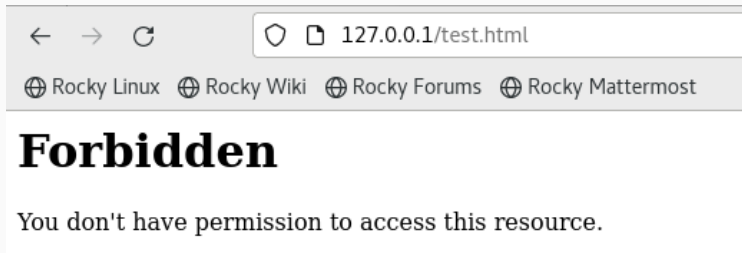


Рис. 17: Ошибка веб-сервера

17. Проанализировала ситуацию. Также просмотрела лог-файл.

```
[root@galeginkikh ~]# ls -l /var/www/html/test.html  
-rw-r--r--. 1 root root 33 Sep 20 14:26 /var/www/html/test.html
```

Рис. 18: Права

```
[root@galeginkikh ~]# tail /var/log/messages
Sep 20 14:33:19 galeginkikh dbus-daemon[799]: [system] Activating service name='org.fedoraproject.SetroubleshootPrivileged' requested by ':1.468' (uid=980 pid=43640 comm="/usr/libexec/platform-python -Es /usr/sbin/setroub" label="system_u:system_r:setroubleshootd_t:s0") (using servicehelper)
Sep 20 14:33:19 galeginkikh dbus-daemon[799]: [system] Successfully activated service 'org.fedoraproject.SetroubleshootPrivileged'
Sep 20 14:33:20 galeginkikh setroubleshoot[43640]: SELinux is preventing httpd from getattr access on the file /var/www/html/test.html. For complete SELinux message s run: sealert -l 0b5c2438-50ed-4f9a-9470-19169920a0b0
Sep 20 14:33:20 galeginkikh setroubleshoot[43640]: SELinux is preventing httpd from
```

Рис. 19: Лог-файл



18. Нашла строчку Listen 80 и заменила её на Listen 81.

```
#Listen 12.34.56.78:80  
Listen 81
```

Рис. 20: Listen 81

19. Проанализировала лог-файлы.

```
[root@galeginkikh ~]# tail -n1 /var/log/messages  
Sep 20 14:44:44 galeginkikh systemd[1]: setroubleshootd.service: Succeeded.
```

Рис. 21: Лог-файлы\_2

20. Проверила список портов командой. Убедитесь, что порт 81 появился в списке.

```
[root@galeginkikh ~]# semanage port -a -t http_port_t -p tcp 81
usage: semanage [-h]
               {import,export,login,user,port,ibpkey,ibendport,interface,module,n
ode,fcontext,boolean,permissive,dontaudit}
               ...
semanage: error: unrecognized arguments: -p 81
[root@galeginkikh ~]# 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@galeginkikh ~]#
```

Рис. 22: Список портов

21. Попробовала запустить веб-сервер Apache ещё раз.

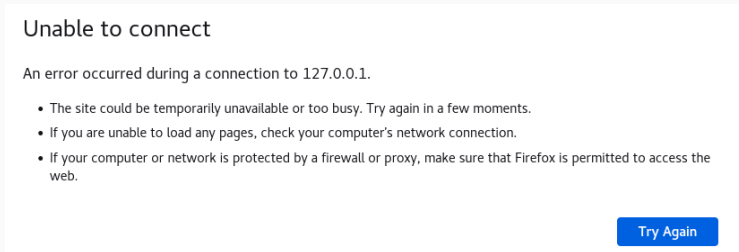


Рис. 23: Unable to connect

22. Вернула контекст `httpd_sys_content__t` к файлу `/var/www/html/ test.html`. После этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`.

```
[root@galeginkikh ~]# chcon -t httpd_sys_content_t /var/www/html/test.html  
[root@galeginkikh ~]#
```

Рис. 24: chcon

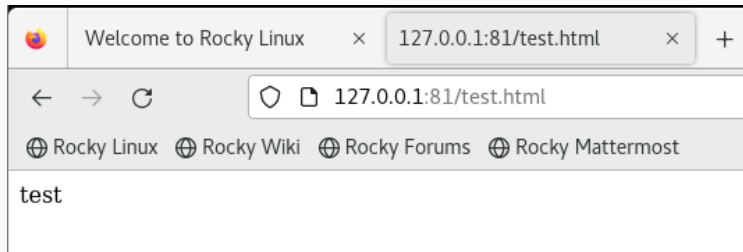


Рис. 25: Веб-сервер\_81

23. Исправила обратно конфигурационный файл apache, вернув Listen 80.

24. Удалила привязку http\_port\_t к 81 порту и проверьте, что порт 81 удалён.

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

Рис. 26: Порт 81



24. Удалила файл /var/www/html/test.html.

```
[root@galeginkikh ~]# semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
[root@galeginkikh ~]# rm /var/www/html/test.html
rm: remove regular file '/var/www/html/test.html'?
[root@galeginkikh ~]# rm /var/www/html/test.html
rm: remove regular file '/var/www/html/test.html'? y
[root@galeginkikh ~]#
```

Рис. 27: rm test.html

## Вывод

---

Развила навыки администрирования ОС Linux. Получила первое практическое знакомство с технологией SELinux. Проверила работу SELinux на практике совместно с веб-сервером Apache.