

Презентация по лабораторной работе №6

SELinux. Apache

Вакутайпа М.

01 мая 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Вактайпа Милдред
- НКАбд-02-23
- Факультет физико-математических и естественных наук
- Российский университет дружбы народов
- 1032239009@rudn.ru
- <https://wakutaipa.github.io>

Цель работы

Развить навыки администрирования ОС linux. Получить практическое знакомство с SELinux1.
Проверить работу SELinux на практике совместно с веб-сервером Apache.

Выполнение лабораторной работы

проверка режима работы SELinux

Вошла в систему под своей учетной записью. Убедилась, что SELinux работает в режиме enforcing политики targeted с помощью getenforce и sestatus

```
[mwakutaipa@mwakutaipa ~]$ 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
```

Рис. 1: проверка режима работы SELinux

Запускаю сервер apache, далее обращаюсь с помощью браузера к веб-серверу, запущенному на компьютере, он работает, что видно из вывода команды `service httpd status`

```
[mwakutaipa@mwakutaipa ~]$ sudo systemctl start httpd
[mwakutaipa@mwakutaipa ~]$ sudo systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr
/lib/systemd/system/httpd.service.
[mwakutaipa@mwakutaipa ~]$ service httpd status
```

Рис. 2: Проверка работы Apache

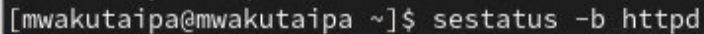
Контекст безопасности Apache

С помощью команды `ps auxZ | grep httpd` нашла веб-сервер Apache в списке процессов. Его контекст безопасности - `httpd_t`

```
[mwakutaipa@mwakutaipa ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0  root    103681  0.4  0.6  21232 11512 ?
Ss  16:07  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache  103682  0.0  0.4  22964  7544 ?
S   16:07  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache  103683  0.2  0.8 1441268 15300 ?
Sl  16:07  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache  103684  0.3  1.0 1572404 19768 ?
Sl  16:07  0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache  103685  0.2  0.9 1441268 17384 ?
Sl  16:07  0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 mwakuta+ 103915 0.0  0.1 2
21660 2304 pts/0 S+ 16:09  0:00 grep --color=auto httpd
[mwakutaipa@mwakutaipa ~]$
```

Рис. 3: Контекст безопасности Apache

Просмотрела текущее состояние переключателей SELinux для Apache

A terminal window with a dark background. The prompt is [mwakutaipa@mwakutaipa ~]\$. The command sestatus -b httpd is entered and highlighted in blue.

```
[mwakutaipa@mwakutaipa ~]$ sestatus -b httpd
```

Рис. 4: Состояние переключателей SELinux

Просмотрела статистику по политике с помощью команды seinfo. Множество пользователей - 8, ролей - 39, типов - 5135.

```
Classes: 135 Permissions: 457
Sensitivities: 1 Categories: 1024
Types: 5169 Attributes: 259
Users: 8 Roles: 15
Booleans: 358 Cond. Expr.: 390
Allow: 65633 Neverallow: 0
Auditallow: 176 Dontaudit: 8703
Type_trans: 271851 Type_change: 94
Type_member: 37 Range_trans: 5931
Role_allow: 40 Role_trans: 417
Constraints: 70 Validatetrans: 0
MLS Constrains: 72 MLS Val. Tran: 0
Permissives: 1 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
Notifcon: 0 Modcon: 0
```

Типы поддиректорий, находящихся в директории /var/www, владелец - root, права на изменения только у владельца. Файлов в директории нет

```
[mwakutaipa@mwakutaipa ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Jan 22 03
:25 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0      6 Jan 22 03
:25 html
```

Рис. 6: Типы поддиректорий

В директории /var/www/html нет файлов.

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

Рис. 7: Типы файлов

Создать файл может только суперпользователь, поэтому от его имени создаем файл touch.html со следующим содержанием:

```
<html>  
<body>test</body>  
</html>
```

Проверяю контекст созданного файла. По умолчанию это httpd_sys_content_t.

```
[mwakutaipa@mwakutaipa ~]$ sudo cat /var/www/html/test.html  
<html>  
<body>Test text ))</body>  
</html>
```

Рис. 8: Контекст файла

Отображение файла

Обращаюсь к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Файл был успешно отображён.

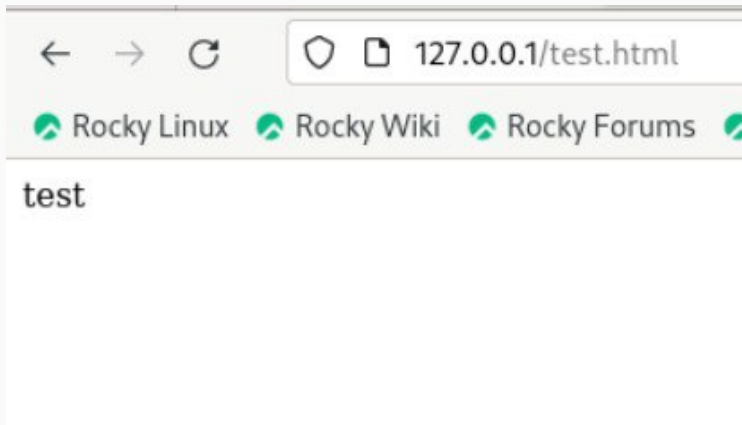


Рис. 9: Отображение файла

Изучила справку `man httpd_selinux`.

```
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|grace-
    ful|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 re-
    quests.

    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
    Manual page httpd(8) line 1 (press h for help or q to quit)
```

Рис. 10: Изучение справки по команде

Изменяю контекст файла test.html с httpd_sys_content_t на любой другой, к которому процесс httpd не должен иметь доступа. Контекст действительно поменялся.

```
[mwakutaipa@mwakutaipa ~]$ ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[mwakutaipa@mwakutaipa ~]$ chcon -t samba_share_t /var/www/html/test.html
chcon: failed to change context of '/var/www/html/test.html' to 'unconfined_u:object_r:samba_share_t:s0': Operation not permitted
```

Рис. 11: Изменение контекста

При попытке отображения файла в браузере получаем сообщение об ошибке.

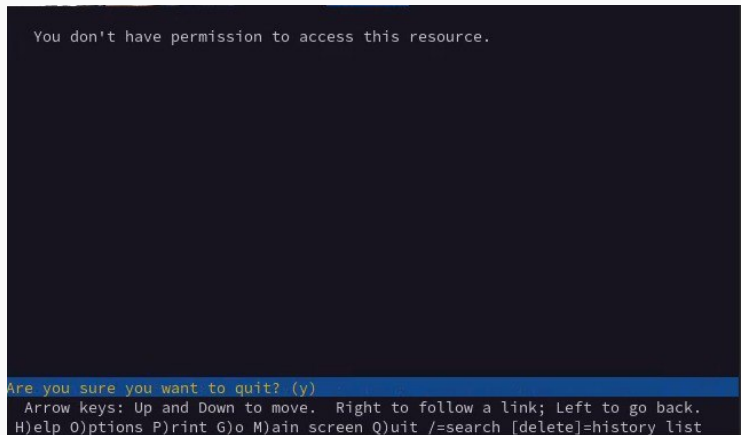
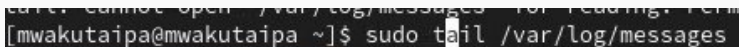


Рис. 12: Отображение файла

файл не был отображён, хотя права доступа позволяют читать этот файл любому пользователю, потому что установлен контекст, к которому процесс httpd не должен иметь доступа.

A terminal window showing a command and its output. The prompt is [mwakutaipa@mwakutaipa ~]. The command is sudo tail /var/log/messages. The output is cat: cannot open '/var/log/messages' for reading: Permission denied.

```
cat: cannot open '/var/log/messages' for reading: Permission denied  
[mwakutaipa@mwakutaipa ~]$ sudo tail /var/log/messages
```

Рис. 13: Попытка прочесть лог-файл

Чтобы запустить веб-сервер Apache на прослушивание TCP-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services) открываю файл /etc/httpd/httpd.conf для изменения. Нахожу строчку Listen 80 и заменяю её на Listen 81.

```
# 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. 14: Изменение файла

Попытка прослушивания другого порта

Выполняю перезапуск веб-сервера Apache. Произошёл сбой, потому что порт 80 для локальной сети, а 81 нет.

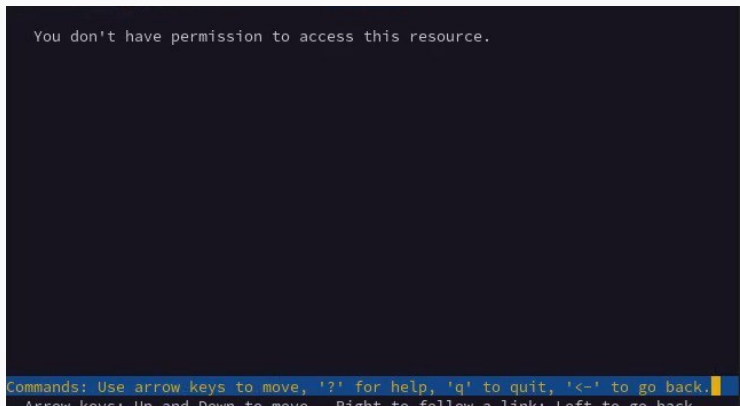


Рис. 15: Попытка прослушивания другого порта

Проанализировала лог-файлы: `tail -nl /var/log/messages`

```
[mwakutaipa@mwakutaipa ~]$ sudo tail -nl /var/log/messages
May  1 16:34:10 mwakutaipa systemd[1]: dbus-:1.1-org.fedoraproject.Setroubleshoo
tPrivileged@1.service: Consumed 12.766s CPU time.
[mwakutaipa@mwakutaipa ~]$
```

Рис. 16: Проверка лог-файлов

Запись появилась в файлу error_log.

```
May  1 16:34:10 mwakutaipa systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@1.service: Consumed 12.766s CPU time.
[mwakutaipa@mwakutaipa ~]$ sudo cat /var/log/http/error_log
cat: /var/log/http/error_log: No such file or directory
[mwakutaipa@mwakutaipa ~]$ sudo cat /var/log/httpd/error_log
[Thu May 01 16:07:06.560501 2025] [core:notice] [pid 103681:tid 103681] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Thu May 01 16:07:06.571606 2025] [suexec:notice] [pid 103681:tid 103681] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using fe80::a00:27ff:feab:92ff%enp0s3. Set the 'ServerName' directive globally to suppress this message
[Thu May 01 16:07:06.865719 2025] [lbmethod_heartbeat:notice] [pid 103681:tid 103681] AH02282: No slotmem from mod_heartbeat
[Thu May 01 16:07:06.979500 2025] [mpm_event:notice] [pid 103681:tid 103681] AH00489: Apache/2.4.62 (Rocky Linux) configured -- resuming normal operations
[Thu May 01 16:07:06.983864 2025] [core:notice] [pid 103681:tid 103681] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[Thu May 01 16:19:05.212933 2025] [autoindex:error] [pid 103683:tid 103779] [client ::1:49644] AH01276: Cannot serve directory /var/www/html/: No matching DirectoryIndex (index.html) found, and server-generated directory index forbidden by Options directive
```

Рис. 17: Проверка лог-файлов

Выполняю команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверяю список портов командой `semanage port -l | grep http_port_t`. Порт 81 появился в списке

```
[mwakutaipa@mwakutaipa ~]$ sudo semanage port -a -t http_port_t -p tcp 81
Port tcp/81 already defined, modifying instead
[mwakutaipa@mwakutaipa ~]$ sudo semanage port -l | grep http_port_t
http_port_t                tcp      81, 80, 81, 443, 488, 8008, 8009, 8443,
9000
pegasus_http_port_t        tcp      5988
[mwakutaipa@mwakutaipa ~]$
```

Рис. 18: Проверка портов

Перезапускаю сервер Apache

```
[mwakutaipa@mwakutaipa ~]$ sudo chcon -t httpd_sys_content_t /var/www/html/test.html  
[mwakutaipa@mwakutaipa ~]$ sudo systemctl restart httpd  
[mwakutaipa@mwakutaipa ~]$ lynx http://127.0.0.1:81/test.html
```

Рис. 19: Перезапуск сервера

Теперь он работает, ведь мы внесли порт 81 в список портов `httpd_port_t`. Возвращаю в файле `/etc/httpd/httpd.conf` порт 80, вместо 81. Проверяю, что порт 81 удален, это правда.

```
[mwakutaipa@mwakutaipa ~]$ sudo semanage port -d -t http_port_t -p tcp 81
[mwakutaipa@mwakutaipa ~]$ 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
[mwakutaipa@mwakutaipa ~]$
```

Рис. 20: Проверка порта 81

Далее удаляю файл test.html, проверяю, что он удален

```
[mwakutaipa@mwakutaipa ~]$ sudo rm /var/www/html/test.html  
[mwakutaipa@mwakutaipa ~]$
```

Рис. 21: Удаление файла

Выводы

При выполнении данной лабораторной работы были развиты навыки администрирования ОС Linux, получено первое практическое знакомство с технологией SELinux и проверена работа SELinux на практике совместно с веб-сервером Apache.