

Лабораторная работа 6

Попов Дмитрий Павлович, НФИбд-01-19

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ПРЕЗЕНТАЦИЯ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

дисциплина: Информационная безопасность

Преподаватель: Кулябов Дмитрий Сергеевич

Студент: Попов Дмитрий Павлович

Группа: НФИбд-01-19

МОСКВА

2022 г.

Цель работы

Цель работы

Целью данной лабораторной работы является развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1. Проверить работу SELinx на практике совместно с веб-сервером Apache.

Выполнение лабораторной работы

1. Входим в систему с полученными учётными данными. Проверили, что SELinux работает в режиме enforcing политики targeted с помощью команд **getenforce** и **sestatus**.

1. Входим в систему с полученными учётными данными. Проверили, что SELinux работает в режиме enforcing политики targeted с помощью команд **getenforce** и **sestatus**.

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

Figure 1: Выполнение команд getenforce и sestatus

2. Запустили веб-сервер и обратились к нему с помощью команды `service httpd status`

2. Запустили веб-сервер и обратились к нему с помощью команды `service httpd status`

```
[dpopov@dpopov ~]$ 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; vendor pre>
   Active: active (running) since Sat 2022-10-15 17:43:03 MSK; 43min ago
     Docs: man:httpd.service(8)
  Main PID: 942 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes>
     Tasks: 213 (limit: 18645)
    Memory: 29.5M
       CPU: 580ms
    CGroup: /system.slice/httpd.service
            └─942 /usr/sbin/httpd -DFOREGROUND
              └─971 /usr/sbin/httpd -DFOREGROUND
                └─973 /usr/sbin/httpd -DFOREGROUND
                  └─974 /usr/sbin/httpd -DFOREGROUND
                    └─975 /usr/sbin/httpd -DFOREGROUND

Oct 15 17:43:03 dpopov.localdomain systemd[1]: Starting The Apache HTTP Server.>
Oct 15 17:43:03 dpopov.localdomain systemd[1]: Started The Apache HTTP Server.
Oct 15 17:43:03 dpopov.localdomain httpd[942]: Server configured, listening on:>
```

Figure 2: Выполнение команды `status`

3. Найшли веб-сервер Apache в списке процессов с помощью команды **ps auxZ | grep httpd**. Контекст безопасности - **unconfined_u:unconfined_r:unconfined_t**.

3. Найшли веб-сервер Apache в списке процессов с помощью команды **ps auxZ | grep httpd**. Контекст безопасности - **unconfined_u:unconfined_r:unconfined_t**.

```
[dpopov@dpopov ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0  root          942  0.0  0.3  20064 11552 ?        Ss
   18:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache        971  0.0  0.2   2156   7300 ?        S
   18:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache        973  0.0  0.3 1079216 11044 ?        Sl
   18:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache        974  0.0  0.4 1210352 13092 ?        Sl
   18:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0  apache        975  0.0  0.4 1079216 13088 ?        Sl
   18:17   0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 dpopov 2466 0.0  0.0 221800 2
236 pts/0 S+  18:26   0:00 grep --color=auto httpd
```

Figure 3: Выполнение команды **ps auxZ | grep httpd**

4. Посмотрели текущее состояние переключателей SELinux для Apache с помощью команды .

4. Посмотрели текущее состояние переключателей SELinux для Apache с помощью команды .

```
[dpopov@dpopov ~]$ 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
```

5. Определили тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды **ls -lZ /var/www**.

5. Определили тип файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды **`ls -lZ /var/www`**.

```
[dpopov@dpopov ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 May 16 15:10
  cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0      6 May 16 15:10
  html
```

Figure 5: Выполнение команды `ls -lZ /var/www`

6. Создали от имени суперпользователя
html-файл /var/www/html/test.html
следующего содержания:

6. Создали от имени суперпользователя html-файл /var/www/html/test.html следующего содержания:

A screenshot of a code editor window with a light gray background. It displays three lines of HTML code, each preceded by a line number in a light gray margin. Line 1 is '<html>', line 2 is '<body>test</body>', and line 3 is '</html>'. The code is in a blue monospace font. A small dark gray cursor is visible at the end of the third line.

```
1 <html>
2 <body>test</body>
3 </html>
```

Figure 6: Содержимое файла test.html

7. Проверили контекст созданного файла -
httpd_sys_content_t.

7. Проверили контекст созданного файла - httpd_sys_content_t.

```
[dpopov@dpopov ~]$ ls -lZ /var/www/html/test.html  
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 15 18:3  
0 /var/www/html/test.html
```

Figure 7: Контекст файла test.html

8. Обратитесь к файлу через веб-сервер,
введя в браузере адрес
`http://127.0.0.1/test.html` и убедитесь, что
файл был успешно отображён.

8. Обратитесь к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html` и убедитесь, что файл был успешно отображён.

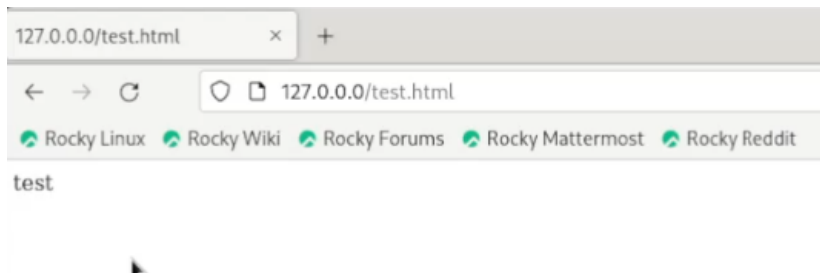


Figure 8: Обращение к файлу test.html через веб-сервер

9. Изменили контекст файла И проверили,
что контекст поменялся.

9. Изменили контекст файла И проверили, что контекст поменялся.

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

Figure 9: Изменение контекста файла /var/www/html/test.html

10. Пробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. В результате получили ошибку.

10. Пробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. В результате получили ошибку.

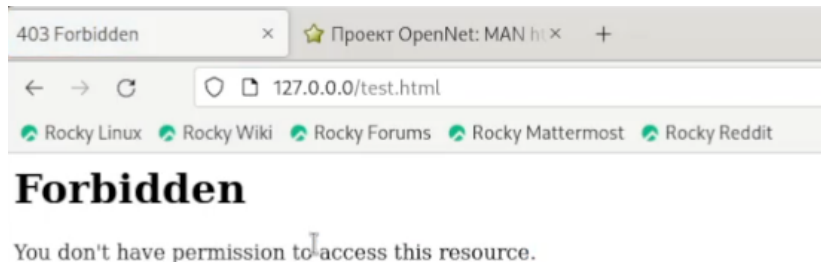


Figure 10: Обращение к файлу `test.html` через веб-сервер после изменения контекста

11. Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81. Для этого в файле `/etc/httpd/httpd.conf` находим строчку `Listen 80` и заменяем её на `Listen 81`.

11. Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81. Для этого в файле `/etc/httpd/httpd.conf` находим строчку `Listen 80` и заменяем её на `Listen 81`.

```
[dpopov@dpopov ~]$ sudo tail -n 10 /var/log/messages
Oct 15 19:07:13 dpopov systemd[1]: dbus-:1.10-org.fedoraproject.Setroubleshootd@0.s
ervice: Failed with result 'signal'.
Oct 15 19:09:20 dpopov journal[1690]: Source ID 12611 was not found when attempting
to remove it
Oct 15 19:09:20 dpopov gnome-shell[1690]: Window manager warning: Buggy client sent
a _NET_ACTIVE_WINDOW message with a timestamp of 0 for 0x8000f5
Oct 15 19:10:06 dpopov systemd[1]: Stopping The Apache HTTP Server...
Oct 15 19:10:07 dpopov systemd[1]: httpd.service: Deactivated successfully.
Oct 15 19:10:07 dpopov systemd[1]: Stopped The Apache HTTP Server.
Oct 15 19:10:07 dpopov systemd[1]: httpd.service: Consumed 2.803s CPU time.
Oct 15 19:10:07 dpopov systemd[1]: Starting The Apache HTTP Server...
Oct 15 19:10:07 dpopov systemd[1]: Started The Apache HTTP Server.
Oct 15 19:10:07 dpopov httpd[3642]: Server configured, listening on: port 81
```

Figure 11: Запуск веб-сервера Apache на прослушивание TCP-порта 81

12(13). Выполним перезапуск веб-сервера Apache. Произошёл сбой? Нет. Выполним команду **semanage port -a -t http_port_t -p tcp 81**. Вылетает ValueError в связи с тем, что порт уже определен. После этого проверим список портов командой **semanage port -l | grep http_port_t** и убедимся, что порт 81 появился в списке.

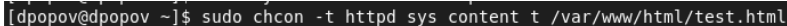
12(13). Выполним перезапуск веб-сервера Apache. Произошёл сбой? Нет. Выполним команду **semanage port -a -t http_port_t -p tcp 81**. Вылетает `ValueError` в связи с тем, что порт уже определен. После этого проверим список портов командой **semanage port -l | grep http_port_t** и убедимся, что порт 81 появился в списке.

```
[dpopov@dpopov ~]$ sudo semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[dpopov@dpopov ~]$ 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
```

Figure 12: Проверка установления 81 порта tcp

14. Вернули контекст `httpd_sys_content_t` к
файлу `/var/www/html/test.html`: **chcon -t**
httpd_sys_content_t
/var/www/html/test.html

14. Вернули контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html`: **`chcon -t httpd_sys_content_t /var/www/html/test.html`**

A terminal window with a dark background. The prompt is [dpopov@dpopov ~]\$. The command entered is sudo chcon -t httpd_sys_content_t /var/www/html/test.html.

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

Figure 13: Возвращение контекста `httpd_sys_content_t` к файлу `test.html`

После этого пробуем получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`. В результате увидели содержимое файла — слово «test».

После этого пробуем получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`. В результате увидели содержимое файла — слово «test».

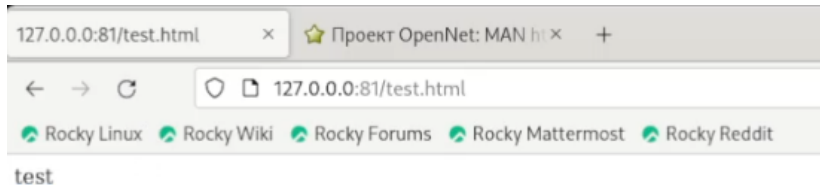


Figure 14: Обращение к файлу `test.html` через веб-сервер

15. Исправим обратно конфигурационный файл apache, вернув Listen 80.

15. Исправим обратно конфигурационный файл apache, вернув Listen 80.

```
46 #Listen 12.34.56.78:80  
47 Listen 80  
48
```

Figure 15: Исправление конфигурационного файла apache

16. Удалим привязку http_port_t к 81 порту: **semanage port -d -t http_port_t -p tcp 81** и проверим, что порт 81 удалён.
Данная команда не была выполнена.

16. Удалим привязку `http_port_t` к 81 порту:
`semanage port -d -t http_port_t -p tcp 81` и
проверим, что порт 81 удалён. Данная команда не
была выполнена.

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

Figure 16: Удаление привязки `http_port_t` к 81 порту

17. Удалим файл /var/www/html/test.html:
rm /var/www/html/test.html.

17. Удалим файл /var/www/html/test.html: **rm /var/www/html/test.html.**

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

Figure 17: Удаление файла test.html

Вывод

Вывод

В ходе выполнения лабораторной работы мы развили навыки администрирования ОС Linux. Получили первое практическое знакомство с технологией SELinux1. Проверили работу SELinux на практике совместно с веб-сервером Apache.

