

# **Отчет по лабораторной работе №6**

**по дисциплине: Информационная безопасность**

Го Чаопэн

# Содержание

<b>1</b>	<b>Цели работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>5</b>	<b>Выводы</b>	<b>17</b>
<b>6</b>	<b>Список литературы</b>	<b>18</b>

## Список иллюстраций

4.1	Конфигурация SELinux . . . . .	7
4.2	Обращение к веб-серверу . . . . .	8
4.3	Контекст безопасности веб-сервера Apache . . . . .	8
4.4	Текущее состояние переключателей SELinux для Apache . . . . .	9
4.5	Статистика по политике . . . . .	10
4.6	Тип файлов и поддиректорий, находящихся в директории /var/www . . . . .	11
4.7	Создание файла /var/www/html/test.html . . . . .	11
4.8	Файл test.html в браузере . . . . .	12
4.9	Вызов справки . . . . .	12
4.10	Изменение контекста . . . . .	12
4.11	Файл test.html в браузере после изменения контекста . . . . .	13
4.12	Содержимое логов . . . . .	13
4.13	Изменение содержимого файла /etc/httpd/httpd.conf . . . . .	14
4.14	Лог-файл tail -nl /var/log/messages . . . . .	14
4.15	Попытка добавления порта 81 в список и вывод списка допустимых портов . . . . .	15
4.16	Повторный запуск веб-сервера . . . . .	15
4.17	Попытка удаления привязки к порту 81 . . . . .	16

# 1 Цели работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux<sup>1</sup>. Проверить работу SELinx на практике совместно с веб-сервером Apache.

## 2 Задание

1. Настроить и запустить сервер Apache.
2. Исследовать влияние параметров сервера на его работу.

### 3 Теоретическое введение

- Операционная система — это комплекс программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем [1].
- Права доступа определяют, какие действия конкретный пользователь может или не может совершать с определенным файлами и каталогами. С помощью разрешений можно создать надежную среду — такую, в которой никто не может поменять содержимое ваших документов или повредить системные файлы. [2].

## 4 Выполнение лабораторной работы

1. Войдем в систему с полученными учётными данными и убедимся, что SELinux работает в режиме enforcing политики targeted (4.1).

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

Рис. 4.1: Конфигурация SELinux

2. Обратимся с помощью браузера к веб-серверу, запущенному на нашем компьютере, и убедимся, что последний работает (4.2).

```
[guo@guo ~]$ service httpd start
Redirecting to /bin/systemctl start httpd.service
[guo@guo ~]$ 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: disabled)
   Active: active (running) since Fri 2023-10-13 20:15:13 MSK; 4s ago
     Docs: man:httpd.service(8)
  Main PID: 70859 (httpd)
    Status: "Started, listening on: port 80"
    Tasks: 213 (limit: 12196)
   Memory: 39.7M
      CPU: 139ms
   CGroup: /system.slice/httpd.service
           └─70859 /usr/sbin/httpd -DFOREGROUND
             └─70867 /usr/sbin/httpd -DFOREGROUND
               └─70868 /usr/sbin/httpd -DFOREGROUND
                 └─70869 /usr/sbin/httpd -DFOREGROUND
                   └─70870 /usr/sbin/httpd -DFOREGROUND

окт 13 20:15:12 guo.localdomain systemd[1]: Starting The Apache HTTP Server...
окт 13 20:15:13 guo.localdomain systemd[1]: Started The Apache HTTP Server.
окт 13 20:15:13 guo.localdomain httpd[70859]: Server configured, listening on: port 80
```

Рис. 4.2: Обращение к веб-серверу

3. Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности (4.3).

```
[guo@guo ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root      70859  0.1  0.5 20328 11632 ?        Ss   20:15   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  70867  0.0  0.3 21664  7440 ?        S    20:15   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  70868  0.0  0.8 2521332 17264 ?      Sl   20:15   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  70869  0.0  0.8 2324660 17264 ?      Sl   20:15   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  70870  0.0  0.8 2324660 17264 ?      Sl   20:15   0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 guo 71130  0.0  0.1 221820 2404 pts/0 S+   20:16   0:00 grep --color=auto httpd
[guo@guo ~]$
```

Рис. 4.3: Контекст безопасности веб-сервера Apache

4. Посмотрим текущее состояние переключателей SELinux для Apache (4.4).



```
[guo@guo ~]$ 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
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_opencryptoki off
httpd_use_openstack off
httpd_use_sasl off
httpd_verify_dns off
[guo@guo ~]$
```

Рис. 4.4: Текущее состояние переключателей SELinux для Apache

5. Посмотрим статистику по политике с помощью команды seinfo (4.5).

```
[guo@guo ~]$ 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:                   5100     Attributes:              258
Users:                   8         Roles:                   14
Booleans:                353     Cond. Expr.:            384
Allow:                   65008    Neverallow:              0
Auditallow:              170     Dontaudit:              8572
Type_trans:              265344   Type_change:             87
Type_member:              35      Range_trans:            6164
Role allow:              38       Role_trans:             420
Constraints:             70      Validatetrans:           0
MLS Constrain:           72      MLS Val. Tran:           0
Permissives:             2        Polcap:                  6
Defaults:                7       Typebounds:              0
Allowxperm:              0        Neverallowxperm:         0
Auditallowxperm:         0       Dontauditxperm:          0
Ibendportcon:            0       Ibpkeycon:               0
Initial SIDs:            27      Fs_use:                  35
Genfscon:                109     Portcon:                 660
Netifcon:                0       Nodecon:                 0
```

Рис. 4.5: Статистика по политике

6. Определим тип файлов и поддиректорий, находящихся в директориях /var/www и /var/www/html. Определим круг пользователей, которым разрешено создание файлов в директории /var/www/html(4.6).

```
[guo@guo ~]$ ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 мая 16 23:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 мая 16 23:21 html
[guo@guo ~]$ ls -lZ /var/www/html
итого 0
[guo@guo ~]$ ll /var/www/html
итого 0
```

Рис. 4.6: Тип файлов и поддиректорий, находящихся в директории /var/www

7. Создадим от имени суперпользователя html-файл /var/www/html/test.html.  
Проверим контекст созданного нами файла (4.7).

```
[root@guo guo]# cd /var/www/html
[root@guo html]# ls
[root@guo html]# nano
[root@guo html]# ls -lZ /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@guo html]# █
```

Рис. 4.7: Создание файла /var/www/html/test.html

Заполним его следующим содержимым:

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

Как видим по умолчанию присваивается контекст *unconfined\_u:object\_r:httpd\_sys\_content\_t:s0*

8. Обратимся к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>.  
Убедимся, что файл был успешно отображён (4.8).

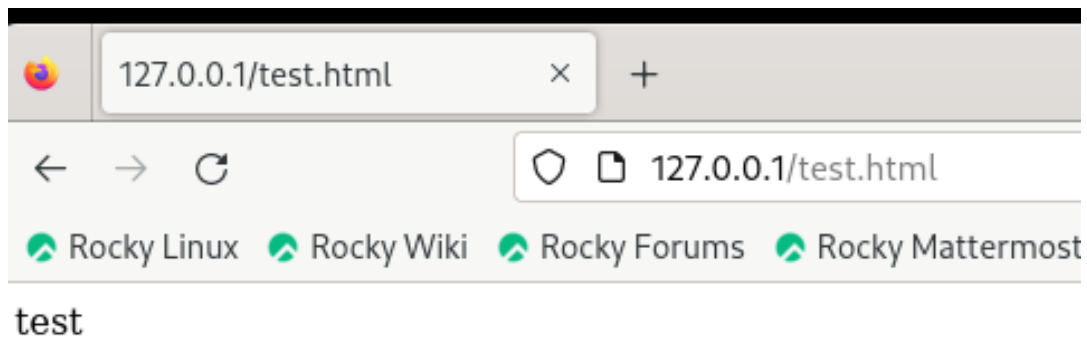


Рис. 4.8: Файл test.html в браузере

9. Изучим справку man httpd\_selinux и выясним, какие контексты файлов определены для httpd. Сопоставим их с типом файла test.html (4.9).

```
[guo@guo ~]$ man selinux
[guo@guo ~]$ ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[guo@guo ~]$ chcon -t samba_share_t /var/www/html/test.html
```

Рис. 4.9: Вызов справки

10. Изменим контекст файла /var/www/html/test.html с httpd\_sys\_content\_t на samba\_share\_t (4.10).

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

Рис. 4.10: Изменение контекста

11. Попробуем ещё раз получить доступ к файлу через веб-сервер (4.11).



```

[root@guo guo]# service httpd stop
Redirecting to /bin/systemctl stop httpd.service
[root@guo guo]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@guo guo]# 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: disabled)
   Active: active (running) since Fri 2023-10-13 20:45:23 MSK; 5s ago
     Docs: man:httpd.service(8)
    Main PID: 3506 (httpd)
   Status: "Started, listening on: port 80"
     Tasks: 213 (limit: 12132)
    Memory: 47.8M
       CPU: 107ms
    CGroup: /system.slice/httpd.service
            └─3506 /usr/sbin/httpd -DFOREGROUND
              └─3507 /usr/sbin/httpd -DFOREGROUND
                └─3508 /usr/sbin/httpd -DFOREGROUND
                  └─3509 /usr/sbin/httpd -DFOREGROUND
                    └─3511 /usr/sbin/httpd -DFOREGROUND

окт 13 20:45:23 guo.localdomain systemd[1]: Starting The Apache HTTP Server...
окт 13 20:45:23 guo.localdomain httpd[3506]: Server configured, listening on: port 80
окт 13 20:45:23 guo.localdomain systemd[1]: Started The Apache HTTP Server.
[root@guo guo]#

```

Рис. 4.13: Изменение содержимого файла /etc/httpd/httpd.conf

#### 14. Проанализируем лог-файлы (4.14).

```

[root@guo guo]# tail /var/log/messages
Oct 13 20:43:49 guo systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@1.service: Deactivated successfully.
Oct 13 20:43:49 guo systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@1.service: Consumed 1.389s CPU time.
Oct 13 20:43:49 guo systemd[1]: setroubleshootd.service: Deactivated successfully.
Oct 13 20:45:08 guo systemd[1]: Stopping The Apache HTTP Server...
Oct 13 20:45:09 guo systemd[1]: httpd.service: Deactivated successfully.
Oct 13 20:45:09 guo systemd[1]: Stopped The Apache HTTP Server.
Oct 13 20:45:09 guo systemd[1]: httpd.service: Consumed 1.182s CPU time.
Oct 13 20:45:23 guo systemd[1]: Starting The Apache HTTP Server...
Oct 13 20:45:23 guo httpd[3506]: Server configured, listening on: port 80
Oct 13 20:45:23 guo systemd[1]: Started The Apache HTTP Server.
[root@guo guo]#

```

Рис. 4.14: Лог-файл tail -nl /var/log/messages

#### 15. 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 Убедимся, что порт 81 есть в списке. (4.15).

```
[root@guo guo]# semanage port -a -t http_port_t -p tcp 81
usage: semanage [-h] {import,export,login,user,port,ibpkey,ibendport,interface,mod
semanage: error: unrecognized arguments: -p 81
[root@guo guo]# 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@guo guo]#
```

Рис. 4.15: Попытка добавления порта 81 в список и вывод списка допустимых портов

16. Попробуем запустить веб-сервер Apache ещё раз. Вернем контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html`. Попробуем получить доступ к файлу через веб-сервер (4.16).

```
[root@guo guo]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@guo guo]# 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: disabled)
   Active: active (running) since Fri 2023-10-13 20:49:50 MSK; 7s ago
     Docs: man:httpd.service(8)
  Main PID: 3816 (httpd)
    Status: "Started, listening on: port 80"
    Tasks: 213 (limit: 12132)
   Memory: 41.3M
      CPU: 147ms
   CGroup: /system.slice/httpd.service
           └─3816 /usr/sbin/httpd -DFOREGROUND
             3817 /usr/sbin/httpd -DFOREGROUND
             3818 /usr/sbin/httpd -DFOREGROUND
             3819 /usr/sbin/httpd -DFOREGROUND
             3829 /usr/sbin/httpd -DFOREGROUND

окт 13 20:49:49 guo.localdomain systemd[1]: Starting The Apache HTTP Server...
окт 13 20:49:50 guo.localdomain httpd[3816]: Server configured, listening on: port 80
окт 13 20:49:50 guo.localdomain systemd[1]: Started The Apache HTTP Server.
[root@guo guo]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@guo guo]#
```

Рис. 4.16: Повторный запуск веб-сервера

17. Исправим обратно конфигурационный файл apache, вернув `Listen 80`. Попробуем удалить привязку `http_port_t` к 81. Удалим файл `/var/www/html/test.html` (4.17).

```
[root@guo guo]# nano /etc/httpd/httpd.conf
[root@guo guo]# semanage port -d -t http_port_t -p tcp 81
ValueError: Порт tcp/81 определен на уровне политики и не может быть удален
[root@guo guo]# sudo semanage port -d -t http_port_t -p tcp 81
ValueError: Порт tcp/81 определен на уровне политики и не может быть удален
[root@guo guo]# rm /var/www/html/test.html
rm: удалить обычный файл '/var/www/html/test.html'? y
[root@guo guo]#
```

Рис. 4.17: Попытка удаления привязки к порту 81



## 5 Выводы

В рамках данной лабораторной работы были развиты навыки администрирования ОС Linux. Получено первое практическое знакомство с технологией SELinux<sup>1</sup>. Проверена работа SELinux на практике совместно с веб-сервером Apache.

## 6 Список литературы

1. Операционные системы [Электронный ресурс]. URL: <https://softline.tm/solutions/programmnoe-obespechenie/operating-system>.
2. Права доступа [Электронный ресурс]. URL: <https://w.wiki/7UBB>.