

# **Отчёт по лабораторной работе №6**

**Мандатное разграничение прав в Linux**

Голощапова Ирина Борисовна

# Содержание

<b>1</b>	<b>Цели и задачи лабораторной работы</b>	<b>5</b>
1.1	Цели и задачи работы . . . . .	5
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Подготовка к выполнению работы . . . . .	6
2.2	Выполнение основной части работы . . . . .	7
<b>3</b>	<b>Выводы</b>	<b>13</b>
<b>4</b>	<b>Библиография</b>	<b>14</b>

# Список иллюстраций

2.1	Установка httpd . . . . .	6
2.2	параметр ServerName . . . . .	6
2.3	Отключение пакетного фильтра . . . . .	7
2.4	Вход в систему . . . . .	7
2.5	Обращение к веб-браузеру . . . . .	7
2.6	Apach . . . . .	8
2.7	переключатели SELinux . . . . .	8
2.8	seinfo . . . . .	9
2.9	тип файлов и поддиректорий . . . . .	9
2.10	html-файл . . . . .	9
2.11	Обращение к файлу через веб-сервер . . . . .	10
2.12	контексты файлов для httpd . . . . .	10
2.13	Изменение контекст файла . . . . .	10
2.14	log-файлы веб-сервера Apache . . . . .	10
2.15	log-файлы веб-сервера Apache_2 . . . . .	11
2.16	Замена на Listen 81 . . . . .	11
2.17	Список портов . . . . .	11
2.18	Возврат контекст . . . . .	12
2.19	Удаление привязки и файла . . . . .	12

## **Список таблиц**

# **1 Цели и задачи лабораторной работы**

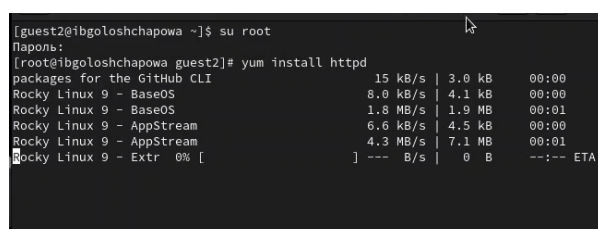
## **1.1 Цели и задачи работы**

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 Выполнение лабораторной работы

### 2.1 Подготовка к выполнению работы

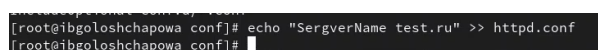
1. Установила от имени администратора httpd (рис. 2.1)



```
[guest2@ibgoloshchapowa ~]$ su root
Пароль:
[root@ibgoloshchapowa guest2]# yum install httpd
packages for the GitHub CLI
Rocky Linux 9 - BaseOS      15 kB/s | 3.0 kB      00:00
Rocky Linux 9 - BaseOS      8.0 kB/s | 4.1 kB      00:00
Rocky Linux 9 - BaseOS      1.8 MB/s | 1.9 MB      00:01
Rocky Linux 9 - AppStream    6.6 kB/s | 4.5 kB      00:00
Rocky Linux 9 - AppStream    4.3 MB/s | 7.1 MB      00:01
Rocky Linux 9 - Extr  0% [ ] --- B/s | 0 B      --:-- ETA
```

Рис. 2.1: Установка httpd

2. В конфигурационном файле /etc/httpd/httpd.conf задала параметр ServerName (рис. 2.2):



```
[root@ibgoloshchapowa conf]# echo "ServerName test.ru" >> httpd.conf
[root@ibgoloshchapowa conf]#
```

Рис. 2.2: параметр ServerName

3. Также проследила, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp (рис. 2.3):

```

[guest2@ibgoloshchapowa ~]$ cd
[root@ibgoloshchapowa ~]# iptables -F
[root@ibgoloshchapowa ~]# iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT
Bad argument 'iptables'
Try 'iptables -h' or 'iptables --help' for more information.
[root@ibgoloshchapowa ~]# iptables -P INPUT -p tcp --dport 80 -j ACCEPT
iptables v1.8.8 (nf_tables): -P requires a chain and a policy
Try 'iptables -h' or 'iptables --help' for more information.
[root@ibgoloshchapowa ~]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
[root@ibgoloshchapowa ~]# iptables -I INPUT -p tcp --dport 81 -j ACCEPT
[root@ibgoloshchapowa ~]# iptables -I OUTPUT -p tcp --sport 80 -j ACCEPT
[root@ibgoloshchapowa ~]# iptables -I OUTPUT -p tcp --sport 81 -j ACCEPT
[root@ibgoloshchapowa ~]#

```

Рис. 2.3: Отключение пакетного фильтра

## 2.2 Выполнение основной части работы

1. Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted (рис. 2.4):

```

[guest2@ibgoloshchapowa ~]$ getenforce
Enforcing
[guest2@ibgoloshchapowa ~]$ setstatus
bash: setstatus: command not found...
[guest2@ibgoloshchapowa ~]$ 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
[guest2@ibgoloshchapowa ~]$

```

Рис. 2.4: Вход в систему

2. Обратилась с помощью браузера к веб-серверу, запущенному на компьютере, и убедилась, что последний работает (рис. 2.5):

```

[root@ibgoloshchapowa guest2]# 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: inactive (dead)
     Docs: man:httpd.service(8)
[lines 1-4/4 (END)]

```

Рис. 2.5: Обращение к веб-браузеру

3. Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности (рис. 2.6):





```

httpd_verify_dns      off
[root@ibgoloshchapowa guest2]# seinfo
Statistics for policy file: /sys/fs/selinux/policy
Policy Version:        33 (MLS enabled)
Target Policy:          selinux
Handle unknown classes: allow
Classes:                135
Sensitivities:          1
Types:                  5100
Users:                  8
Booleans:               353
Allow:                  65000
Auditallow:             170
Type_trans:             265341
Type_member:            35
Role allow:             38
Constraints:            70
MLS Constrains:         72
Permissives:            2
Defaults:               7
Allowxperm:             0
Auditallowxperm:        0
Ibendportcon:           0
Initial SIDs:           27
Genfscon:               109
Netifcon:               0
Permissions:            457
Categories:            1024
Attributes:             258
Roles:                  14
Cond. Expr.:            384
Neverallow:             0
Dontaudit:              8572
Type_change:            87
Range_trans:            6164
Role_trans:             420
Validatetrans:          0
MLS Val. Tran:          0
Polcap:                 6
Typebounds:             0
Neverallowxperm:        0
Dontauditxperm:         0
Ibpkeycon:              0
Fs_use:                 35
Portcon:                660
Nodecon:                0
[root@ibgoloshchapowa guest2]#

```

Рис. 2.8: seinfo

- Определила тип файлов и поддиректорий, находящихся в директории /var/www (рис. 2.9):

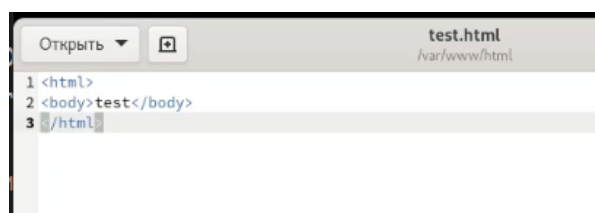
```

[root@ibgoloshchapowa guest2]# ls -lZ /var/www
lrwxr-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
[root@ibgoloshchapowa guest2]#

```

Рис. 2.9: тип файлов и поддиректорий

- Создала от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html следующего содержания (рис. 2.10):



```

test.html
/var/www/html
1 <html>
2 <body>test</body>
3 </html>

```

Рис. 2.10: html-файл

8. Обратилась к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html` (рис. 2.11):

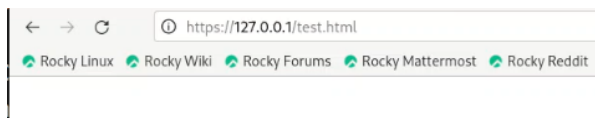


Рис. 2.11: Обращение к файлу через веб-сервер

9. Изучила справку `man httpd_selinux` и выяснила, какие контексты файлов определены для `httpd`. Сопоставила их с типом файла `test.html`. Проверила контекст файла командой `ls -Z` (рис. 2.12):

```
[root@ibgoloshchapowa html]# man httpd_selinux
Нет справочной страницы для httpd_selinux
[root@ibgoloshchapowa html]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@ibgoloshchapowa html]#
```

Рис. 2.12: контексты файлов для `httpd`

10. Изменила контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на другой, к которому процесс `httpd` не должен иметь доступа, например, на `samba_share_t` (рис. 2.13):

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

Рис. 2.13: Изменение контекст файла

11. Проанализировала ситуацию. Просмотрела `log`-файлы веб-сервера `Apache`. Также просмотрела системный `лог`-файл (рис. 2.14) (рис. 2.15):

```
[root@ibgoloshchapowa html]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 окт 13 19:21 /var/www/html/test.html
[root@ibgoloshchapowa html]# tail /var/log/messages
```

Рис. 2.14: `log`-файлы веб-сервера `Apache`

```

Oct 13 19:22:21 ibgoloshchapowa gnome-shell[1893]: libinput error: event5 - VirtualBox mouse integration: client bug: event processing lagging behind by 14ms, your system is too slow
Oct 13 19:22:28 ibgoloshchapowa firefox.desktop[42157]: Missing chrome or resource URL: resource://gre/modules/UpdateListener.jsm
Oct 13 19:22:28 ibgoloshchapowa firefox.desktop[42157]: Missing chrome or resource URL: resource://gre/modules/UpdateListener.sys.mjs
Oct 13 19:24:00 ibgoloshchapowa firefox.desktop[42157]: Missing chrome or resource URL: resource://gre/modules/UpdateListener.jsm
Oct 13 19:24:00 ibgoloshchapowa firefox.desktop[42157]: Missing chrome or resource URL: resource://gre/modules/UpdateListener.sys.mjs
Oct 13 19:24:02 ibgoloshchapowa systemd[1775]: app-gnome-firefox-42157.scope: Consumed 39.354s CPU time.
Oct 13 19:24:03 ibgoloshchapowa systemd[1775]: Started Application launched by gnome-shell.
Oct 13 19:24:05 ibgoloshchapowa rtkit-daemon[725]: Successfully made thread 42658 of process 42545 (/usr/lib64/firefox/firefox) owned by '1003' RT at priority 10.
Oct 13 19:24:11 ibgoloshchapowa firefox.desktop[42545]: Missing chrome or resource URL: resource://gre/modules/UpdateListener.jsm
Oct 13 19:24:11 ibgoloshchapowa firefox.desktop[42545]: Missing chrome or resource URL: resource://gre/modules/UpdateListener.sys.mjs
[root@ibgoloshchapowa html]#

```

Рис. 2.15: log-файлы веб-сервера Apache\_2

12. Попробовала 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 нашла строчку Listen 80 и заменила её на Listen 81 (рис. 2.16):

```

40 #
41 # Change this to Listen on a specific IP address, but note that if
42 # httpd.service is enabled to run at boot time, the address may not be
43 # available when the service starts. See the httpd.service(8) man
44 # page for more information.
45 #
46 #Listen 12.34.56.78:80
47 Listen 81
48
49 #
50 # Dynamic Shared Object (DSO) Support
51 #
52 # To be able to use the functionality of a module which was built as a DSO you
53 # have to place corresponding 'LoadModule' lines at this location so the

```

Рис. 2.16: Замена на Listen 81

13. Выполнила команду и после этого проверила список портов(рис. 2.17):

```

[root@ibgoloshchapowa ~]# semanage port -a -t http_port_t -p tcp 81
ValueError: Порт tcp/81 уже определен
[root@ibgoloshchapowa ~]# semanage port -l | grep http_port_t
http_port_t                tcp      80, 81, 443, 488, 8008, 8009, 8443, 900
0
pegasus_http_port_t        tcp      5988
[root@ibgoloshchapowa ~]#

```

Рис. 2.17: Список портов

14. Вернула контекст httpd\_sys\_content\_t к файлу /var/www/html/ test.html и после этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html (рис. 2.18):

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

Рис. 2.18: Возврат контекст

15. Исправила обратно конфигурационный файл apache, вернув Listen 80. Удалила привязку http\_port\_t к 81 порту и проверила, что порт 81 удалён. Затем удалила файл /var/www/html/test.html (рис. 2.19):

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

Рис. 2.19: Удаление привязки и файла

## 3 Выводы

В ходе лабораторной работы мне удалось:

- Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1;
- Проверить работу SELinx на практике совместно с веб-сервером Apache/

## 4 Библиография

1. Git - система контроля версий
2. Rocky Linux