

Отчет по лабораторной работе №6

Информационная безопасность

Нгуен Фыюк Дат

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	16
	Список литературы	17

Список таблиц

Список иллюстраций

4.1	Проверка режима работы	8
4.2	Проверка работы веб-сервера	8
4.3	Запуск веб-сервера	9
4.4	Определение контекста безопасности	9
4.5	Просмотр переключателей	10
4.6	Статистика по политике	11
4.7	Определение типов поддиректорий	11
4.8	Создание файла test.html	12
4.9	Проверка контекста test.html	12
4.10	Просмотр файла в браузере	12
4.11	Смена контекста	12
4.12	Просмотр файла в браузере	13
4.13	Чтение лог-файлов	13
4.14	Смена порта	13
4.15	Перезапуск веб-сервера	14
4.16	Проверка лог-файлов	14
4.17	Просмотр списка портов	14
4.18	Смена контекста	15
4.19	Просмотр файла	15
4.20	Изменение порта	15
4.21	Попытка удаления порта	15

1 Цель работы

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

2 Задание

- Поиск информации про веб-сервер
- Работа с Html-файлами
- Просмотр лог-файлов

3 Теоретическое введение

SELinux представляет собой систему маркировки, каждый процесс файл, каталог, пользователь, устройство, порт и так далее имеет метку. SELinux определяет правила доступа процесса к объектам с определенными метками. Это называется политикой.

Владелец файла не имеет полной свободы действий над атрибутами безопасности. Стандартные атрибуты контроля доступа, такие как группа и владелец ничего не значат для SELinux. Полностью все управляется метками. Значения атрибутов могут быть установлены и без прав root, но на это нужно иметь специальные полномочия SELinux.

SELinux может работать в трех режимах — отключен, система полностью отключена и не работает, режим ограничений Enforcing — программа активирована и блокирует все не соответствующие политикам действия и третий режим Permissive — только фиксировать нарушения.

Политики SELinux бывают тоже нескольких типов. Политика targeted относится к типу Type Enforcement (TE) политик, в которых управление доступом к файлам осуществляется на основе ролей. Сюда же относится политика strict. Есть ещё политики Multi-Level Security (MLS), в которых добавлены дополнительные категории.

Более подробно о см. в [1,2].

4 Выполнение лабораторной работы

В качестве первого шага лабораторной работы мы проверили режим работы SELinux с помощью команд `getenforce` и `sestatus` (рис. 4.1).

```
[root@phuocdat ~]# getenforce
Enforcing
[root@phuocdat ~]# 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@phuocdat ~]# 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)
```

Рис. 4.1: Проверка режима работы

Далее мы проверили, работает ли веб-сервер (рис. 4.2), и запустили его, так как он не работал (рис. 4.3).

```
[root@phuocdat ~]# 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)
```

Рис. 4.2: Проверка работы веб-сервера


```
[root@phuocdat ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@phuocdat ~]# 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 Sat 2023-10-14 18:29:29 MSK; 2s ago
     Docs: man:httpd.service(8)
  Main PID: 40127 (httpd)
    Status: "Started, listening on: port 80"
    Tasks: 213 (limit: 5768)
   Memory: 24.8M
      CPU: 155ms
   CGroup: /system.slice/httpd.service
           └─40127 /usr/sbin/httpd -DFOREGROUND
             └─40128 /usr/sbin/httpd -DFOREGROUND
               └─40132 /usr/sbin/httpd -DFOREGROUND
                 └─40133 /usr/sbin/httpd -DFOREGROUND
                   └─40134 /usr/sbin/httpd -DFOREGROUND
```

Рис. 4.3: Запуск веб-сервера

Определили контекст безопасности процесса веб-сервера (рис. 4.4). Главной информацией для нас стал тип процесса — `httpd_t`.

```
[root@phuocdat ~]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 40127 0.1 1.1 20116 11504 ? Ss 18:29 0:
00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 40128 0.0 0.7 21600 7384 ? S 18:29 0:
00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 40132 0.0 1.1 1079372 11024 ? Sl 18:29 0:
00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 40133 0.0 1.3 1210508 13072 ? Sl 18:29 0:
00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 40134 0.0 1.1 1079372 11024 ? Sl 18:29 0:
00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 40371 0.0 0.2 221664 2372 pts/0 S+ 1
8:31 0:00 grep --color=auto httpd
```

Рис. 4.4: Определение контекста безопасности

Посмотрели текущее положение переключателей SELinux, большинство из них находятся в выключенном состоянии (рис. 4.5).

```
[root@phuocdat ~]# 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
```

Рис. 4.5: Просмотр переключателей

Посмотрели статистику по политике с помощью `seinfo` (рис. 4.6). Определили, что множество пользователей имеет размер 8, множество ролей — 14, а множество типов — 5100.

```
[root@phuocdat ~]# 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:                   65009    Neverallow:              0
Auditallow:              170     Dontaudit:               8572
Type_trans:              265337   Type_change:             87
Type_member:              35      Range_trans:             6164
Role allow:              38       Role_trans:              420
Constraints:              70     Validatetrans:           0
MLS Constrains:          72      MLS Val. Tran:           0
Permissives:              2       Polcap:                  6
Defaults:                 7      Typebounds:              0
Allowxperm:               0       Neverallowxperm:         0
Auditallowxperm:          0      Dontauditxperm:          0
```

Рис. 4.6: Статистика по политике

Определили тип файлов и поддиректорий директории /var/www (рис. 4.7). Поддиректория cgi-bin имеет тип httpd_sys_script_exec_t, а html — httpd_sys_content_t. Только пользователь-владелец имеет право создавать файлы в папке html.

```
[root@phuocdat ~]# 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 23:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0      6 May 16 23:21 html
[root@phuocdat ~]# ls -lZ /var/www/html
total 0
```

Рис. 4.7: Определение типов поддиректорий

Создали файл test.html в папке html от лица суперпользователя (рис. 4.8).

```
[root@phuocdat ~]# su
[root@phuocdat ~]# touch /var/www/html/test.html
[root@phuocdat ~]# echo '<html>' > /var/www/html/test.html
[root@phuocdat ~]# echo '<body>test</body>' >> /var/www/html/test.html
[root@phuocdat ~]# echo '</html>' >> /var/www/html/test.html
```

Рис. 4.8: Создание файла test.html

Проверили его контекст (рис. 4.9). Вновь созданным файлам в папке html по умолчанию присваивается тип `httpd_sys_content_t`.

```
[root@phuocdat ~]# ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 Oct 14 18:43 /var/www/html/test.html
```

Рис. 4.9: Проверка контекста test.html

Обратились к файлу через веб-сервер и увидели его содержимое (рис. 4.10).

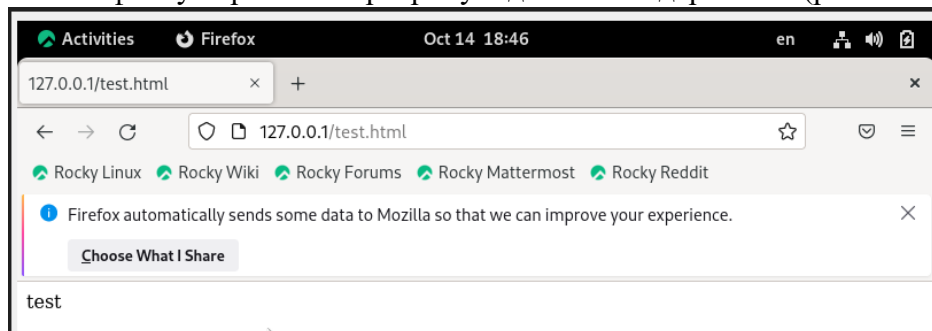


Рис. 4.10: Просмотр файла в браузере

Снова проверили контекст файла и поменяли его на другой (рис. 4.11). Новый контекст файла не позволяет процессу `httpd` получить доступ к файлу при обращении к нему через браузер.

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

Рис. 4.11: Смена контекста

Попробовали открыть файл в браузере (рис. 4.12). Возникла ошибка из-за нового контекста.

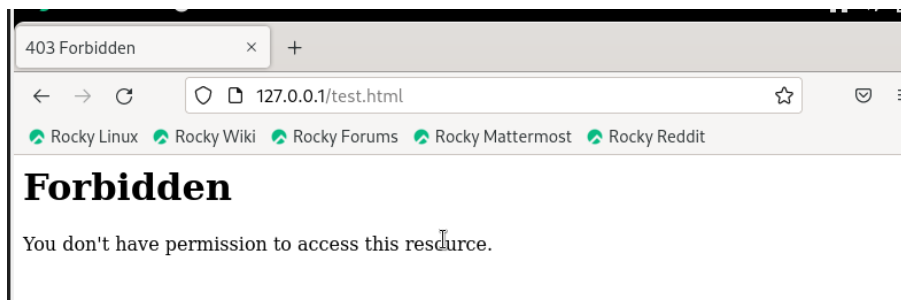


Рис. 4.12: Просмотр файла в браузере

Посмотрели лог-файлы (рис. 4.13). Увидели запись о неудачной попытке браузера получить доступ к файлу (ошибка 403).

```
[root@phucdat ~]# tail /var/log/messages
Oct 14 18:51:55 phucdat gnome-shell[1654]: libinput error: event3 - ImExPS/2 Generic Explorer
Mouse: client bug: event processing lagging behind by 15ms, your system is too slow
Oct 14 18:51:56 phucdat firefox.desktop[41393]: Missing chrome or resource URL: resource://gre/
modules/UpdateListener.jsm
Oct 14 18:51:57 phucdat firefox.desktop[41393]: Missing chrome or resource URL: resource://gre/
```

Рис. 4.13: Чтение лог-файлов

Изменили порт в конфигурационном файле httpd.conf с 80 на 81 (рис. 4.14).

```
httpd.conf  [-M--]  0 L: [ 29+19  48/360]  *(2026/12024b) 0010 0x00A  (*)
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used.  If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default.  See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts.  See the httpd.service(8) man
# page for more information.
#Listen 12.34.56.78:80
Listen 81
```

Рис. 4.14: Смена порта

Перезапустили веб-сервер, получили сообщение о том, что он запущен на прослушивание 81 порта (рис. 4.15).

```

[root@phuocdat conf]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[root@phuocdat conf]# cd ~
[root@phuocdat ~]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[root@phuocdat ~]# 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 Sat 2023-10-14 19:00:38 MSK; 19s ago
     Docs: man:httpd.service(8)
   Main PID: 42104 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0"
      Tasks: 213 (limit: 5768)
    Memory: 24.7M
       CPU: 115ms
    CGroup: /system.slice/httpd.service
            └─42104 /usr/sbin/httpd -DFOREGROUND
              └─42105 /usr/sbin/httpd -DFOREGROUND
                └─42109 /usr/sbin/httpd -DFOREGROUND
                  └─42110 /usr/sbin/httpd -DFOREGROUND

```

Рис. 4.15: Перезапуск веб-сервера

Проверили лог-файлы и нашли информацию о переключении веб-сервера на прослушивание нового порта (рис. 4.16).

```

[root@phuocdat ~]# tail -n1 /var/log/messages
Oct 14 19:03:41 phuocdat cupsd[832]: REQUEST localhost - - "POST / HTTP/1.1" 200 183 Renew-Subscription successful-ok
[root@phuocdat ~]# tail -n1 /var/log/httpd/error_log
tail: cannot open '/var/log/httpd/error_log' for reading: No such file or directory
[root@phuocdat ~]# tail -n1 /etc/log/httpd/error_log
tail: cannot open '/etc/log/httpd/error_log' for reading: No such file or directory
[root@phuocdat ~]# tail -n1 /etc/httpd/error_log
tail: cannot open '/etc/httpd/error_log' for reading: No such file or directory
[root@phuocdat ~]# tail -n1 /etc/httpd/logs/error_log
[Sat Oct 14 19:00:38.691029 2023] [core:notice] [pid 42104:tid 42104] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[root@phuocdat ~]# tail -n1 /etc/httpd/logs/access_log
127.0.0.1 - - [14/Oct/2023:18:51:44 +0300] "GET /test.html HTTP/1.1" 403 199 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0"
[root@phuocdat ~]# tail -n1 /var/log/audit/audit.log
type=SERVICE_START msg=audit(1697299238.674:219): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=httpd comm="systemd" exe="/usr/lib/systemd/systemd" host=

```

Рис. 4.16: Проверка лог-файлов

Посмотрели список портов веб-сервера, нашли там указанный нами порт (рис. 4.17).

```

[root@phuocdat ~]# semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[root@phuocdat ~]# 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

```

Рис. 4.17: Просмотр списка портов

Вернули файлу test.html старый контекст (рис. 4.18).

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

Рис. 4.18: Смена контекста

Получили доступ к файлу через веб-сервер в браузере (рис. 4.19).

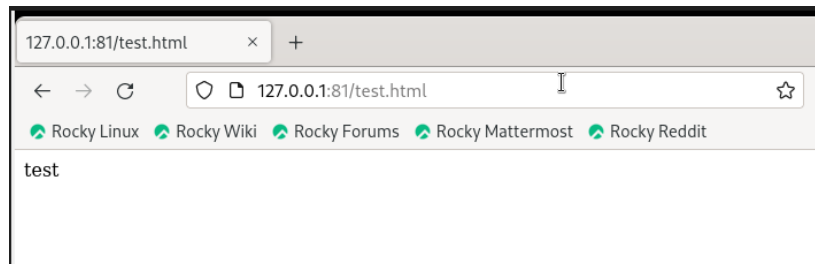


Рис. 4.19: Просмотр файла

Вернули порт 80 в конфигурационном файле (рис. 4.20).

```
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts. See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 80
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# functions can be found at runtime.
```

Рис. 4.20: Изменение порта

Попытались удалить 81 порт, но столкнулись с ошибкой, что он определен на уровне политики и не может быть удален (рис. 4.21). После этого удалили файл test.html.

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

Рис. 4.21: Попытка удаления порта

5 Выводы

В результате лабораторной работы я получила базовые навыки администрирования ОС Linux, познакомилась с технологией SELinux и проверила ее работу на практике совместно с веб-сервером Apache.

Список литературы

1. Мандатное разграничение прав в Linux [Электронный ресурс]. URL: https://esystem.rudn.ru/pluginfile.php/2090282/mod_resource/content/2/006-lab_selinux.pdf.
2. Настройка SELinux [Электронный ресурс]. URL: <https://losst.pro/nastrojka-selinux>.