

Отчет по лабораторной работе №5

Дискреционное разграничение прав в Linux. Расширенные атрибуты

Низамова Альфия Айдаровна

Содержание

1	Цель работы	5
2	Выполнение	6
2.1	Создание программы	6
3	Вывод	13

Список иллюстраций

Рис. 1	6
Рис. 2	7
Рис. 3	7
Рис. 4	8
Рис. 5	8
Рис. 8	9
Рис. 9	9
Рис. 10	10
Рис. 11	10
Рис. 12	10
Рис. 13	10
Рис. 14	11
Рис. 15	11
Рис. 16	12
Рис. 17	12

Список таблиц

1 Цель работы

Развить навыки администрирования ОС Linux. Получить первое практическое знакомство с технологией SELinux1 .

Проверить работу SELinx на практике совместно с веб-сервером Apache.

2 Выполнение

2.1 Создание программы

1. Войдем в систему с полученными учётными данными и убедимся, что SELinux работает в режиме enforcing политики targeted (рис.1)
2. Обратимся с помощью браузера к веб-серверу, запущенному на компьютере, и убедимся, что последний работает (рис.1)

```
[root@aanizamovalocaldomain ~]# getenforce
Enforcing
[root@aanizamovalocaldomain ~]# 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@aanizamovalocaldomain ~]# 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: d>
   Active: inactive (dead)
   Docs: man:httpd.service(8)
lines 1-4/4 (END)
```

Рис. 1

3. Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности. (рис.2)

```
[root@aanizamovalocaldomain ~]# ps auxZ | grep httpd
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 40044 0.0  0.1 221664
2300 pts/0 S+ 18:15   0:00 grep --color=auto httpd
```

Рис. 2

4. Посмотрим текущее состояние переключателей SELinux для Apache. Многие из них находятся в положении «off» (рис.3)

```
[root@aanizamovalocaldomain ~]# 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
```

Рис. 3

5. Посмотрим статистику по политике, также определим множество пользователей, ролей, типов (рис.4)

```
[root@aanizamovalocaldomain ~]# 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 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

6. Определим тип файлов и поддиректорий, находящихся в директории */var/www* (рис.5)

```
[root@aanizamovalocaldomain ~]# 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 20:21 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0      6 May 16 20:21 html
```

Рис. 5

7. Определим тип файлов, находящихся в директории */var/www/html* (рис.6)
8. Определим круг пользователей, которым разрешено создание файлов в директории */var/www/html* - только суперпользователь (рис.6)

9. Создадим от имени суперпользователя html-файл `/var/www/html/test.html` (рис.6-7)

```
[root@aanizamovalocaldomain ~]# ls -lZ /var/www/html
total 0
[root@aanizamovalocaldomain ~]# echo "<html>\n<body>test</body>\n</html>" > /var/www/html/test.html
[root@aanizamovalocaldomain ~]# cat /var/www/html/test.html
<html>\n<body>test</body>\n</html>
[root@aanizamovalocaldomain ~]# gedit /var/www/html/test.html
```

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

10. Проверим контекст созданного файла. (рис.8)

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

Рис. 8

11. Обратимся к файлу через веб-сервер (рис.9)

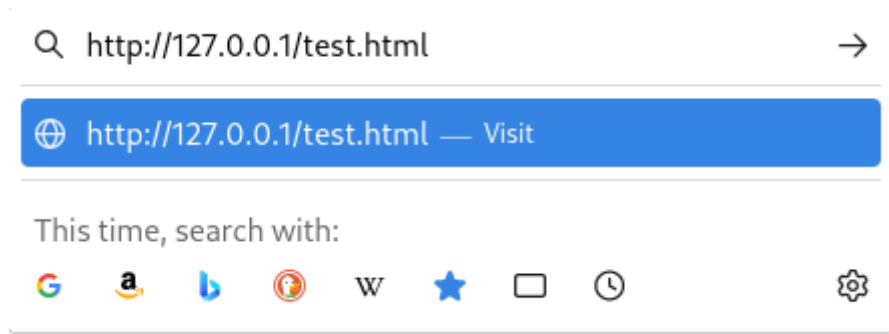


Рис. 9

12. Изучим справку `man httpd_selinux` и выясним, какие контексты файлов определены для `httpd`. Выдало, что справки нет (рис.10)
13. Изменим контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на `samba_share_t`.

После этого проверим, что контекст поменялся (рис.10)

```
[root@aanizamovalocaldomain ~]# man httpd_selinux
No manual entry for httpd_selinux
[root@aanizamovalocaldomain ~]# chcon -t samba_share_t /var/www/html/test.html
[root@aanizamovalocaldomain ~]# ls -Z /var/www/html/test.html
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Рис. 10

14. Попробуем ещё раз получить доступ к файлу через веб-сервер, мы должны получить сообщение об ошибке (рис.11)

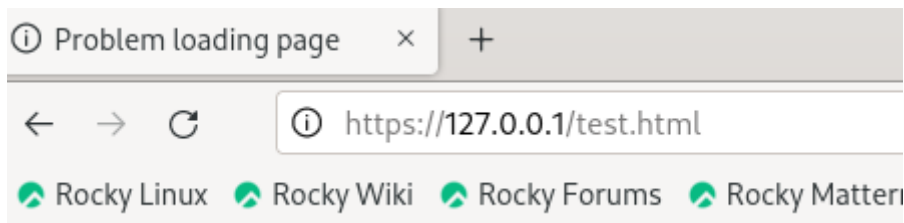


Рис. 11

15. Просмотрим log-файлы веб-сервера Apache. Также посмотрим системный лог-файл (рис.12)

```
[root@aanizamovalocaldomain ~]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct 12 18:30 /var/www/html/test.html
[root@aanizamovalocaldomain ~]# tail /var/log/messages
Oct 12 18:36:26 aanizamovalocaldomain firefox.desktop[40596]: Missing chrome or
resource URL: resource://gre/modules/UpdateListener.jsm
Oct 12 18:36:26 aanizamovalocaldomain firefox.desktop[40596]: Missing chrome or
resource URL: resource://gre/modules/UpdateListener.sys.mis
```

Рис. 12

16. Попробуем 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 заменим строчку *Listen 80* на *Listen 81* (рис.13)

```
44 # page for more information.
45 #
46 #Listen 12.34.56.78:80
47 Listen 81
48
49 #
```

Рис. 13

17. Выполним перезапуск веб-сервера Apache. (рис.14)
18. Проанализируем лог-файлы. Просмотрим файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и выясним, в каких файлах появились записи. (рис.14)

```
[root@aanizamovalocaldomain ~]# apache
bash: apache: command not found...
[root@aanizamovalocaldomain ~]# tail -nl /var/log/messages
tail: invalid number of lines: 'l'
[root@aanizamovalocaldomain ~]# tail -l /var/log/messages
Oct 12 18:36:26 aanizamovalocaldomain firefox.desktop[40596]: Missing chrome or
resource URL: resource://gre/modules/UpdateListener.jsm
Oct 12 18:36:26 aanizamovalocaldomain firefox.desktop[40596]: Missing chrome or
```

Рис. 14

19. Выполним заданную в методичке команду.
После этого проверим список портов. Убедимся, что порт 81 появился в списке (рис.15)
20. Попробуем запустить веб-сервер Apache ещё раз (рис.15)

```
[root@aanizamovalocaldomain ~]# semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[root@aanizamovalocaldomain ~]# 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@aanizamovalocaldomain ~]# apache
bash: apache: command not found...
```

Рис. 15

21. Вернем контекст `*httpd_sys_content_t*` к файлу `/var/www/html/ test.html` (рис.16)
22. Исправим обратно конфигурационный файл apache, вернув `Listen 80` (рис.16)

```
[root@aanizamovalocaldomain ~]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@aanizamovalocaldomain ~]# gedit /etc/httpd/conf/httpd.conf
```

Рис. 16

23. Удалим привязку *http_port_t* к 81 порту и проверим, что порт 81 удалён (рис.17)

24. Удалим файл */var/www/html/test.html* (рис.17)

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

Рис. 17

3 Вывод

Мы развили навыки администрирования ОС Linux, а также получили первое практическое знакомство с технологией SELinux1.

Проверили работу SELinx на практике совместно с веб-сервером Apache.