Лабораторная работа №6

Мандатное разграничение прав в Linux

Го Чаопэн

13 октября 2023

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

- Го Чаопэн
- студент НФИбд-02-20
- Российский университет дружбы народов
- 1032194919@pfur.ru
- https://github.com/LIONUCKY

Вводная часть

Актуальность

Логические объекты файловой системы (файлы) являются носителями своеобразных меток, которые привычно называют правами доступа. Некоторые метки действительно означают право выполнения определенного действия пользователя над этим объектом. Важно изучить их для дальнейшего применения на практике.

Объект и предмет исследования

- Атрибуты файлов
- Дистрибутив Rocky
- Дискреционное разграничение доступа

Цели и задачи

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

Войдем в систему с полученными учётными данными и убедимся, что SELinux работает в режиме enforcing политики targeted

```
[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. 1: Конфигурация SELinux

Выполнение работы

Обратимся с помощью браузера к веб-серверу, запущенному на нашем компьютере, и убедимся, что последний работает

```
[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. 2: Обращение к веб-серверу

Выполнение работы

Найдем веб-сервер Apache в списке процессов, определим его контекст безопасности

```
[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. 3: Контекст безопасности веб-сервера Apache

Выполнение работы

Посмотрим текущее состояние переключателей SELinux для Apache

```
[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
```

Выполнение работы

Посмотрим статистику по политике с помощью команды seinfo

```
[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
```

Выполнение работы

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

```
[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. 6: Тип файлов и поддиректорий, находящихся в директории /var/www

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

```
[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. 7: Создание файла /var/www/html/test.html

Как видим по умолчанию присваивается контекст *unconfined_u:object_r:httpd_sys_content_t:s0*

Обратимся к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html.

Убедимся, что файл был успешно отображён

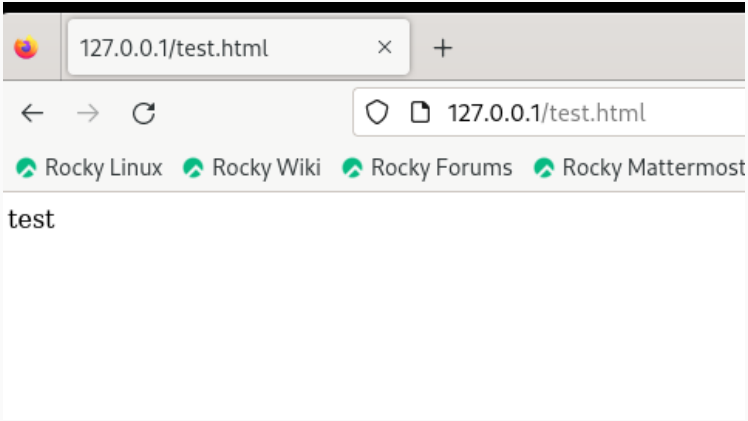


Рис. 8: Файл test.html в браузере

Изучим справку man httpd_selinux и выясним, какие контексты файлов определены для httpd. Сопоставим их с типом файла test.html

```
[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. 9: Вызов справки

Изменим контекст файла /var/www/html/test.html с httpd_sys_content_t на samba_share_t

```
[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. 10: Изменение контекста

Попробуем ещё раз получить доступ к файлу через веб-сервер

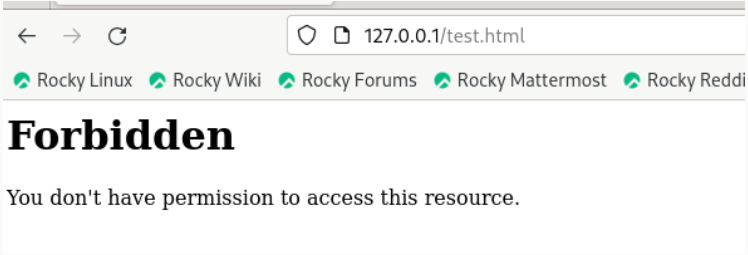


Рис. 11: Файл test.html в браузере после изменения контекста

Выполнение работы

Просмотрим log-файлы веб-сервера Apache и системный лог-файл

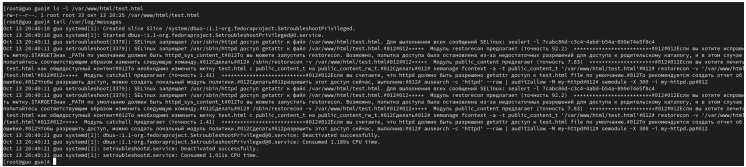
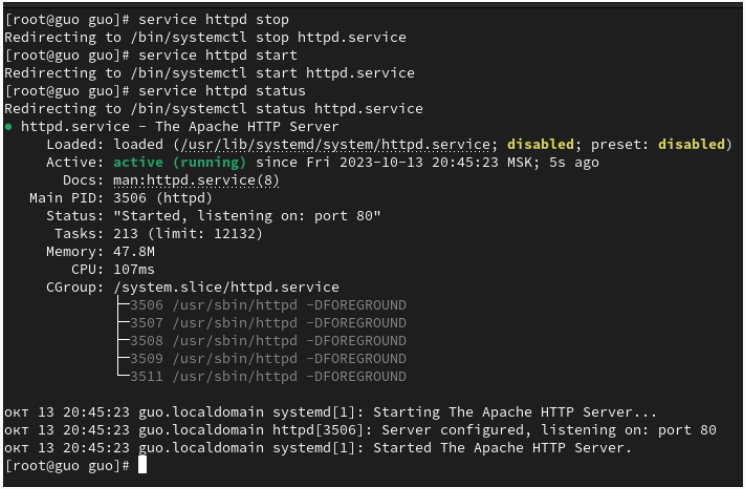


Рис. 12: Содержимое логов

Как видим, нам не удалось получить доступ к файлу как раз из-за измененного контекста.

Выполнение работы

Попробуем запустить веб-сервер Apache на прослушивание TCP-порта 81. Выполним перезапуск веб-сервера. Сбоя не произошло



Проанализируем лог-файлы

```
[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. 14: Лог-файл tail -nl /var/log/messages

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.

```
[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. 15: Попытка добавления порта 81 в список и вывод списка допустимых портов

Выполнение работы

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

```
[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]#
```

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

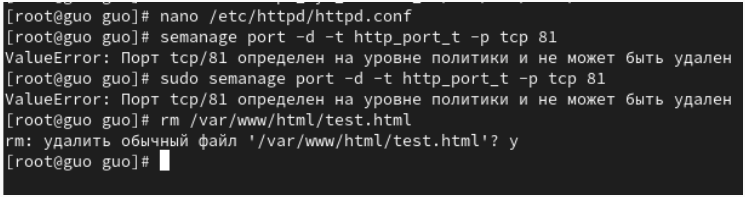


Рис. 17: Попытка удаления привязки к порту 81

Вывод

- В рамках данной лабораторной работы были развиты навыки администрирования ОС Linux
- Получено первое практическое знакомство с технологией SELinux1. Проверена работа SELinx на практике совместно с веб-сервером Apache