

# Лабораторная работа №6

Мандатное разграничение прав в Linux

---

Ван И

13 октября 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Ван И
- студент НФИбд-02-20
- Российский университет дружбы народов
- 1032198069@pfur.ru
- <https://github.com/WangYi157>

## Вводная часть

---

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.

- Атрибуты файлов
- Дистрибутив Rocky
- Дискреционное разграничение доступа

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.

## Ход работы

---



## Выполнение работы

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

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

## Выполнение работы

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

```
[yi@yi ~]$ service httpd start
Redirecting to /bin/systemctl start httpd.service
[yi@yi ~]$ 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 22:35:13 MSK; 4s ago
     Docs: man:httpd.service(8)
   Main PID: 41345 (httpd)
    Status: "Started, listening on: port 80"
     Tasks: 213 (limit: 12196)
    Memory: 32.1M
       CPU: 132ms
    CGroup: /system.slice/httpd.service
            └─41345 /usr/sbin/httpd -DFOREGROUND
              └─41353 /usr/sbin/httpd -DFOREGROUND
                └─41357 /usr/sbin/httpd -DFOREGROUND
                  └─41360 /usr/sbin/httpd -DFOREGROUND
                    └─41378 /usr/sbin/httpd -DFOREGROUND

окт 13 22:35:12 yi.localdomain systemd[1]: Starting The Apache HTTP Server...
окт 13 22:35:13 yi.localdomain systemd[1]: Started The Apache HTTP Server.
окт 13 22:35:13 yi.localdomain httpd[41345]: Server configured, listening on: port 80
[yi@yi ~]$
```

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

```
[yi@yi ~]$ ps auxZ | grep httpd
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 40699 0.0 0.4 238316 9036 pts/0 T 22:28 0:00 sudo vi /etc/httpd/conf/httpd.conf
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 40701 0.0 0.4 230180 8720 pts/0 T 22:28 0:00 /usr/bin/vim /etc/httpd/conf/httpd.conf
system_u:system_r:httpd_t:s0 root 41345 0.3 0.5 20328 11396 ? Ss 22:35 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41353 0.0 0.3 21664 7384 ? S 22:35 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41357 0.0 0.8 2521332 17176 ? Sl 22:35 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41360 0.0 0.5 2324660 11040 ? Sl 22:35 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 41378 0.0 0.5 2324660 11064 ? Sl 22:35 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 yi 41608 0.0 0.1 221820 2380 pts/0 S+ 22:35 0:00 grep --color=auto httpd
```

Рис. 3: Контекст безопасности веб-сервера Apache

Посмотрим текущее состояние переключателей SELinux для Apache

```
[yi@yi ~]$ sestatus -b | grep httpd
httpd_anon_write off
httpd_builtinscripting 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
```

Посмотрим статистику по политике с помощью команды seinfo

```
# Installing packages...
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:                    65000    Neverallow:              0
Auditallow:               170      Dontaudit:               8572
Type_trans:               265341   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
Ibendportcon:             0       Ibpkeycon:               0
Initial SIDe:             27      Fs_use:                  25
```

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`

```
[yi@yi ~]$ 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 yi yi system_u:object_r:httpd_sys_content_t:s0 6 мая 16 23:21 html
drwxr-xr-x. 2 root root unconfined_u:object_r:httpd_sys_content_t:s0 6 окт 13 22:26 log
[yi@yi ~]$ ls -lZ /var/www/html
итого 0
[yi@yi ~]$ ll /var/www/html/
итого 0
[yi@yi ~]$
```

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

```
[yi@yi ~]$ sudo su
[sudo] пароль для yi:
[root@yi yi]# nano /var/www/html/test.html
[root@yi yi]# ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.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

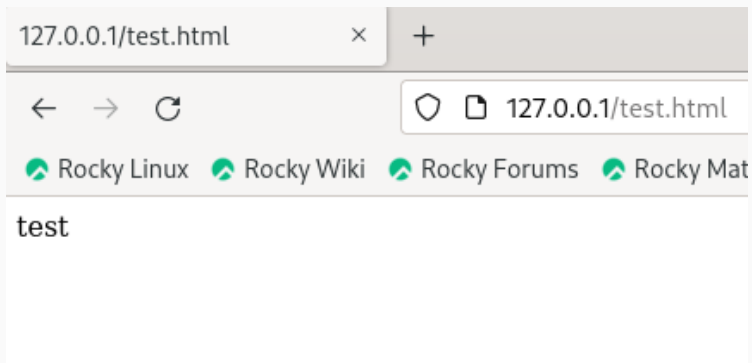


Рис. 8: Файл test.html в браузере



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

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

Рис. 10: Изменение контекста

Попробуем ещё раз получить доступ к файлу через веб-сервер

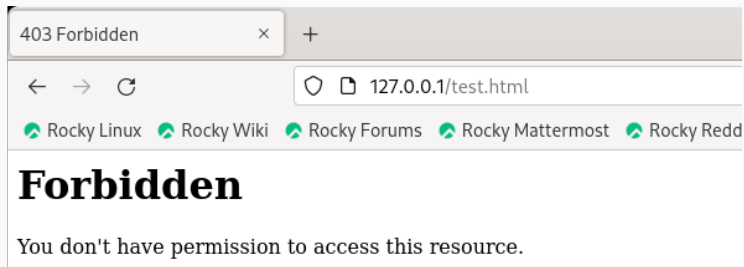
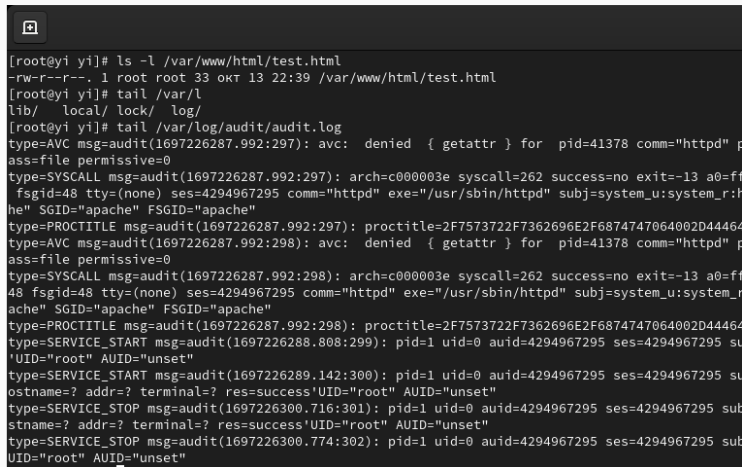


Рис. 11: Файл test.html в браузере после изменения контекста

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

A terminal window with a dark background and a window icon in the top-left corner. It displays the output of several commands: 'ls -l /var/www/html/test.html', 'tail /var/lib/local/lock/log', and 'tail /var/log/audit/audit.log'. The audit log output shows various system events, including denied file access attempts by httpd and service start/stop events for root.

```
[root@yi yi]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 окт 13 22:39 /var/www/html/test.html
[root@yi yi]# tail /var/lib/local/lock/log
[root@yi yi]# tail /var/log/audit/audit.log
type=AVC msg=audit(1697226287.992:297): avc: denied { getattr } for pid=41378 comm="httpd" p
ass=file permissive=0
type=SYSCALL msg=audit(1697226287.992:297): arch=c000003e syscall=262 success=no exit=-13 a0=ff
fsgid=48 tty=(none) ses=4294967295 comm="httpd" exe="/usr/sbin/httpd" subj=system_u:system_r:h
he" SGID="apache" FSGID="apache"
type=PROCTITLE msg=audit(1697226287.992:297): proctitle=2F7573722F7362696E2F6874747064002D44464
type=AVC msg=audit(1697226287.992:298): avc: denied { getattr } for pid=41378 comm="httpd" p
ass=file permissive=0
type=SYSCALL msg=audit(1697226287.992:298): arch=c000003e syscall=262 success=no exit=-13 a0=ff
48 fsgid=48 tty=(none) ses=4294967295 comm="httpd" exe="/usr/sbin/httpd" subj=system_u:system_r
ache" SGID="apache" FSGID="apache"
type=PROCTITLE msg=audit(1697226287.992:298): proctitle=2F7573722F7362696E2F6874747064002D44464
type=SERVICE_START msg=audit(1697226288.808:299): pid=1 uid=0 auid=4294967295 ses=4294967295 su
'UID="root" AUID="unset"
type=SERVICE_START msg=audit(1697226289.142:300): pid=1 uid=0 auid=4294967295 ses=4294967295 su
ostname=? addr=? terminal=? res=success'UID="root" AUID="unset"
type=SERVICE_STOP msg=audit(1697226300.716:301): pid=1 uid=0 auid=4294967295 ses=4294967295 sub
stname=? addr=? terminal=? res=success'UID="root" AUID="unset"
type=SERVICE_STOP msg=audit(1697226300.774:302): pid=1 uid=0 auid=4294967295 ses=4294967295 sub
UID="root" AUID="unset"
```

Рис. 12: Содержимое логов

## Выполнение работы

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

```
[root@yi yi]# nano /etc/h
host.conf hostname hosts hp/ httpd/
[root@yi yi]# nano /etc/httpd/conf/httpd.conf
[root@yi yi]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[root@yi yi]# 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 22:48:12 MSK; 7s ago
     Docs: man:httpd.service(8)
   Main PID: 42333 (httpd)
   Status: "Started, listening on: port 81"
    Tasks: 213 (limit: 12196)
   Memory: 43.7M
      CPU: 100ms
   CGroup: /system.slice/httpd.service
           └─42333 /usr/sbin/httpd -DFOREGROUND
             └─42335 /usr/sbin/httpd -DFOREGROUND
               └─42336 /usr/sbin/httpd -DFOREGROUND
                 └─42337 /usr/sbin/httpd -DFOREGROUND
                   └─42364 /usr/sbin/httpd -DFOREGROUND

окт 13 22:48:12 yi.localdomain systemd[1]: Starting The Apache HTTP Server...
окт 13 22:48:12 yi.localdomain httpd[42333]: Server configured, listening on: port 81
окт 13 22:48:12 yi.localdomain systemd[1]: Started The Apache HTTP Server.
[root@yi yi]#
```

Проанализируем лог-файлы

```
[root@yi yi]# tail /var/log/messages
Oct 13 22:45:00 yi systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged
Oct 13 22:45:00 yi systemd[1]: setroubleshootd.service: Deactivated successfully.
Oct 13 22:45:00 yi systemd[1]: setroubleshootd.service: Consumed 1.090s CPU time.
Oct 13 22:48:11 yi systemd[1]: Stopping The Apache HTTP Server...
Oct 13 22:48:12 yi systemd[1]: httpd.service: Deactivated successfully.
Oct 13 22:48:12 yi systemd[1]: Stopped The Apache HTTP Server.
Oct 13 22:48:12 yi systemd[1]: httpd.service: Consumed 1.412s CPU time.
Oct 13 22:48:12 yi systemd[1]: Starting The Apache HTTP Server...
Oct 13 22:48:12 yi httpd[42333]: Server configured, listening on: port 81
Oct 13 22:48:12 yi systemd[1]: Started The Apache HTTP Server.
```

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@yi yi]# semanage port -a -t http_port_t -p tcp 81
usage: semanage [-h] {import,export,login,user,port,ibpkey,ibendport,interface,module,node,fcontext,boolean,permissive,dontaudit}
semanage: error: unrecognized arguments: -p 81
[root@yi yi]# 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@yi yi]#
```

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

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@yi yi]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
[root@yi yi]# 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 22:59:36 MSK; 9s ago
     Docs: man:httpd.service(8)
  Main PID: 42768 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
   Tasks: 213 (limit: 12196)
  Memory: 41.5M
    CPU: 137ms
  CGroup: /system.slice/httpd.service
          └─42768 /usr/sbin/httpd -DFOREGROUND
            └─42770 /usr/sbin/httpd -DFOREGROUND
              └─42771 /usr/sbin/httpd -DFOREGROUND
                └─42772 /usr/sbin/httpd -DFOREGROUND
                  └─42773 /usr/sbin/httpd -DFOREGROUND

окт 13 22:59:35 yi.localdomain systemd[1]: Starting The Apache HTTP Server...
окт 13 22:59:36 yi.localdomain httpd[42768]: Server configured, listening on: port 81
окт 13 22:59:36 yi.localdomain systemd[1]: Started The Apache HTTP Server.
[root@yi yi]#
```

Рис. 16: Повторный запуск веб-сервера



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

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

Рис. 17: Попытка удаления привязки к порту 81

- 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