

Лабораторная работа №6

Мандатное разграничение прав в Linux

Лебедева Ольга Андреевна

Содержание

Цель работы	4
Теоретическое введение	5
Выполнение лабораторной работы	6
Заключение	15
Библиографическая справка	16

Список иллюстраций

1	Скачивание httpd	6
2	ServerName	6
3	Вход в систему	7
4	Обращение к веб-серверу	7
5	Контекст безопасности	8
6	Состояние переключателей SELinux	8
7	Статистика по политике	9
8	Определение типа файлов и директорий	9
9	Создание файла test.html	10
10	Просмотр файла через веб-сервер	10
11	Контексты файлов для httpd	11
12	Изменение контекста	11
13	Ошибка доступа	12
14	Просмотр log-файлов	12
15	Замена порта на 81	12
16	Лог-файлы	13
17	Проверка порта	13
18	Запуск сервера	13
19	Возврат контекста	13
20	Удаление привязки к 81 порту	14
21	Удаление файла	14

Цель работы

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

Теоретическое введение

Мандатное управление доступом (Mandatory Access Control, MAC) предназначено для обеспечения большего уровня безопасности и контроля над доступом к ресурсам системы.

Мандатное разграничение доступа применяется в совокупности с дискреционным разграничением доступа. Оно определяет правила доступа на основе атрибутов объектов и субъектов, которые затем при проверке определяют разрешен ли доступ. Объект в данной модели – это то, над чем совершаются какие-либо действия, а субъект – исполнитель этого действия. Значение уровня доступа субъекта или объекта называется меткой. Метка может быть символьной или числовой. Проверка полномочий определяется при помощи сопоставления меток объекта и субъекта. Пользователи системы не могут самостоятельно определять доступ субъектов к объектам. Управление доступом субъектов к объектам осуществляют только администраторы[1].

Выполнение лабораторной работы

Перед выполнением лабораторной работы подготовим рабочее пространство и скачаем httpd: См. рис. 1

```
[oalebedeva@oalebedeva ~]$ su
Password:
[root@oalebedeva oalebedeva]# yum install httpd
Last metadata expiration check: 0:23:38 ago on Fri 11 Oct 2024 05:28:44 MSK.
Dependencies resolved.
=====
Package                Arch      Version              Repository           Size
=====
Installing:
httpd                  x86_64    2.4.57-11.el9_4.1   appstream            44 k
Installing dependencies:
apr                    x86_64    1.7.0-12.el9_3      appstream            122 k
apr-util               x86_64    1.6.1-23.el9        appstream            94 k
apr-util-bdb           x86_64    1.6.1-23.el9        appstream            12 k
httpd-core              x86_64    2.4.57-11.el9_4.1   appstream            1.4 M
httpd-filesystem        noarch    2.4.57-11.el9_4.1   appstream            11 k
=====
```

Рис. 1: Скачивание httpd

В конфигурационном файле задаем ServerName и отключаем пакетный фильтр: См. рис. 2

```
Complete!
[root@oalebedeva oalebedeva]# cd /etc/httpd
[root@oalebedeva httpd]# echo "ServerName test.ru" >> httpd.conf
[root@oalebedeva httpd]# iptables -F
[root@oalebedeva httpd]# iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT
Bad argument `iptables'
Try `iptables -h' or 'iptables --help' for more information.
[root@oalebedeva httpd]# iptables -P INPUT ACCEPT
[root@oalebedeva httpd]# iptables -P OUTPUT ACCEPT
```

Рис. 2: ServerName

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted: См. рис. 3

```
[root@oalebedeva httpd]# 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
```

Рис. 3: Вход в систему

2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает: См. рис. 4

```
[root@oalebedeva httpd]# service httpd status
Redirecting to /bin/systemctl status httpd.service
o httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset:
   Active: inactive (dead)
         Docs: man:httpd.service(8)
lines 1-4/4 (END)
[root@oalebedeva httpd]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@oalebedeva httpd]# 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:
   Active: active (running) since Fri 2024-10-11 06:01:21 MSK; 17s ago
         Docs: man:httpd.service(8)
    Main PID: 42528 (httpd)
   Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Byt
Tasks: 177 (limit: 23039)
Memory: 22.2M
CPU: 50ms
```

Рис. 4: Обращение к веб-серверу

3. Найдите веб-сервер Араче в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт: См. рис. 5

```
[root@oalebedeva httpd]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 42528 0.0 0.3 20364 11332 ?
Ss 06:01 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 42529 0.0 0.1 22096 7384 ?
S 06:01 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 42533 0.0 0.2 981648 11204 ?
Sl 06:01 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 42534 0.0 0.3 1112784 13536 ?
Sl 06:01 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 42535 0.0 0.2 981648 11216 ?
Sl 06:01 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 43256 0.0 0.0 221
796 2304 pts/0 S+ 06:05 0:00 grep --color=auto httpd
```

Рис. 5: Контекст безопасности

4. Посмотрите текущее состояние переключателей SELinux для Apache: См. рис. 6

```
[root@oalebedeva httpd]# 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
```

Рис. 6: Состояние переключателей SELinux

Посмотрите статистику по политике с помощью команды `seinfo`, также определите множество пользователей, ролей, типов: См. рис. 7


```
[root@oalebedeva httpd]# 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:        5145    Attributes:      259
Users:        8       Roles:          15
Booleans:     356     Cond. Expr.:    388
Allow:        65504   Neverallow:     0
Auditallow:   176     Dontaudit:      8682
Type_trans:   271770  Type_change:    94
Type_member:  37      Range_trans:    5931
Role_allow:   40      Role_trans:     417
Constraints:  70      Validatetrans:  0
MLS Constrai: 72      MLS Val. Tran:  0
Permissives:  4       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:        665
Netifcon:     0       Nodecon:        0
```

Рис. 7: Статистика по политике

6. Определите тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды `ls -lZ /var/www`
7. Определите тип файлов, находящихся в директории /var/www/html: `ls -lZ /var/www/html`: См. рис. 8

```
[root@oalebedeva httpd]# ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Aug 8
19:30 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Aug 8
19:30 html
[root@oalebedeva httpd]# ls -lZ /var/www/html
total 0
```

Рис. 8: Определение типа файлов и директорий

8. Определите круг пользователей, которым разрешено создание файлов в директории /var/www/html.

9. Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл /var/www/html/test.html.
10. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html.: См. рис. 9

```
[root@oalebedeva httpd]# touch /var/www/html/test.html
[root@oalebedeva httpd]# cd ..
[root@oalebedeva etc]# cd
[root@oalebedeva ~]# ls
anaconda-ks.cfg
[root@oalebedeva ~]# echo '<html>' >> /var/www/html/test.html
[root@oalebedeva ~]# echo '<body>test<body>' >> /var/www/html/test.html
[root@oalebedeva ~]# echo '</html>' >> /var/www/html/test.html
[root@oalebedeva ~]# ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 32 Oct 1
1 06:13 /var/www/html/test.html
```

Рис. 9: Создание файла test.html

11. Обратитесь к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Убедитесь, что файл был успешно отображён: См. рис. 10

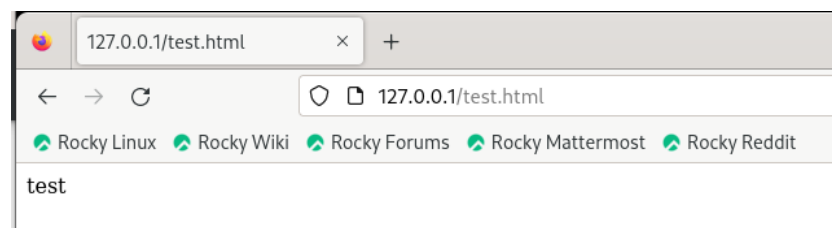
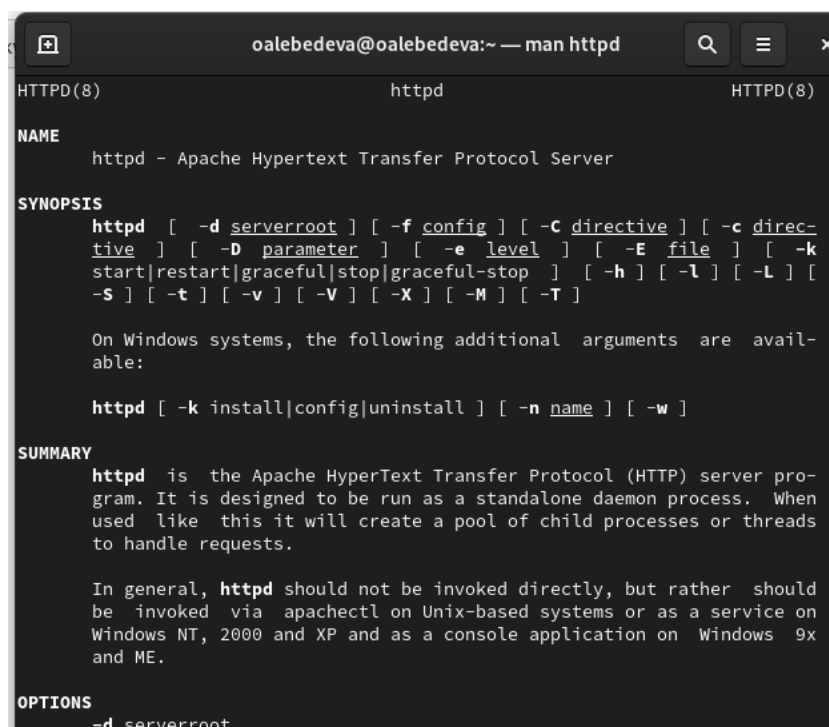


Рис. 10: Просмотр файла через веб-сервер

12. Изучите справку `man httpd_selinux` и выясните, какие контексты файлов определены для httpd: См. рис. 11



```
oalebedeva@oalebedeva:~ — man httpd
HTTPD(8)                                httpd                                HTTPD(8)

NAME
    httpd - Apache Hypertext Transfer Protocol Server

SYNOPSIS
    httpd [ -d serverroot ] [ -f config ] [ -C directive ] [ -c directive ] [ -D parameter ] [ -e level ] [ -E file ] [ -k start|restart|graceful|stop|graceful-stop ] [ -h ] [ -l ] [ -L ] [ -s ] [ -t ] [ -v ] [ -V ] [ -X ] [ -M ] [ -T ]

    On Windows systems, the following additional arguments are available:

    httpd [ -k install|config|uninstall ] [ -n name ] [ -w ]

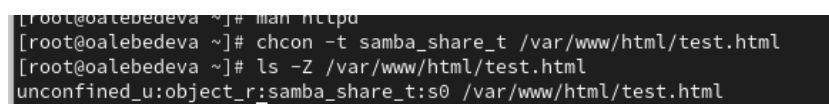
SUMMARY
    httpd is the Apache HyperText Transfer Protocol (HTTP) server program. It is designed to be run as a standalone daemon process. When used like this it will create a pool of child processes or threads to handle requests.

    In general, httpd should not be invoked directly, but rather should be invoked via apachectl on Unix-based systems or as a service on Windows NT, 2000 and XP and as a console application on Windows 9x and ME.

OPTIONS
    -d serverroot
```

Рис. 11: Контексты файлов для httpd

13. Измените контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на любой другой, к которому процесс `httpd` не должен иметь доступа, например, на `samba_share_t`. См. рис. 12



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

Рис. 12: Изменение контекста

14. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Вы должны получить сообщение об ошибке: См. рис. 13

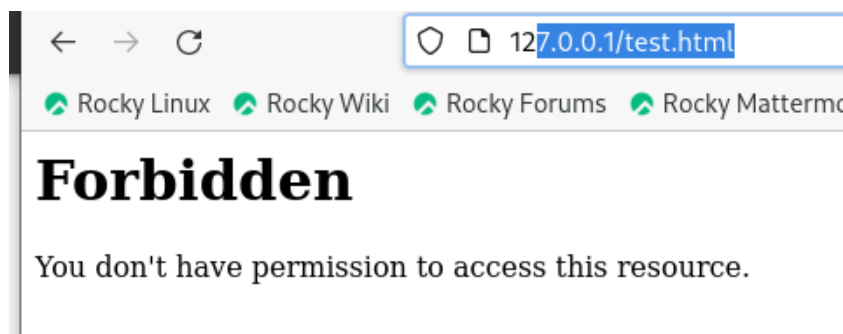


Рис. 13: Ошибка доступа

15. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл. См. рис. 14

```
[root@oalebedeva ~]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 32 Oct 11 06:13 /var/www/html/test.html
[root@oalebedeva ~]# tail /var/log/messages
Oct 11 06:21:29 oalebedeva systemd[1]: Starting SETroubleshoot daemon for processing new SELinux denial logs...
Oct 11 06:21:29 oalebedeva systemd[1]: Started SETroubleshoot daemon for processing new SELinux denial logs.
Oct 11 06:21:29 oalebedeva setroubleshoot[44203]: failed to retrieve rpm info
```

Рис. 14: Просмотр log-файлов

16. Попробуйте 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 найдите строчку Listen 80 и замените её на Listen 81. См. рис. 15

```
#Listen 12.34.56.78:80
Listen 81
```

Рис. 15: Замена порта на 81

17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему?

18. Проанализируйте лог-файлы. См. рис. 16

```
tail: invalid number of lines.
[root@oalebedeva ~]# tail -n1 /var/log/messages
Oct 11 06:37:31 oalebedeva gnome-shell[3162]: Source ID 12955 was not found when attempting to remove it
```

Рис. 16: Лог-файлы

19. Выполните команду `semanage port -a -t http_port_t tcp 81`. После этого проверьте список портов. Убедитесь, что порт 81 появился в списке. См. рис. 17

```
[root@oalebedeva ~]# semanage port -a -t http_port_t tcp 81
usage: semanage [-h] (--import,--export,--login,--user,--port,--ibpkey,--ibendport,--interface,--module,--node,--fcontext,--boolean,--permissive,--dontaudit) ...
semanage: error: unrecognized arguments: 81
[root@oalebedeva ~]# 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. 17: Проверка порта

20. Попробуйте запустить веб-сервер Apache ещё раз. См. рис. 18

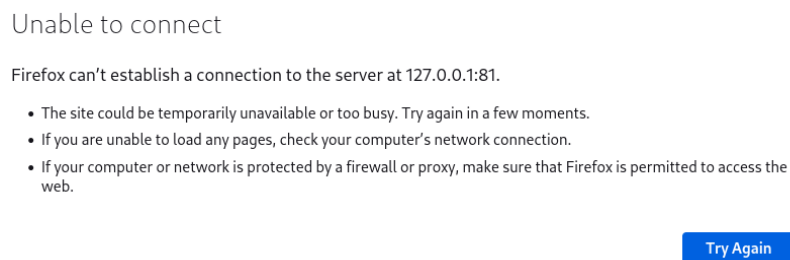


Рис. 18: Запуск сервера

21. Верните контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html`: `chcon -t httpd_sys_content_t /var/www/html/test.html`. После этого попробуйте получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`. Вы должны увидеть содержимое файла — слово «test». См. рис. 19

```
[lines 1-41]
[root@oalebedeva ~]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@oalebedeva ~]#
```

Рис. 19: Возврат контекста

22. Исправьте обратно конфигурационный файл apache, вернув Listen 80.
23. Удалите привязку http_port_t к 81 порту и проверьте, что порт 81 удалён. См. рис. 20

```
[root@oalebedeva conf]# nano httpd.conf  
[root@oalebedeva conf]# semanage port -d -t http_port_t -p tcp 81
```

Рис. 20: Удаление привязки к 81 порту

24. Удалите файл /var/www/html/test.html: См. рис. 21

```
rm: cannot remove '/var/www/html/test.html': No such file or dir  
[root@oalebedeva oalebedeva]# rm /var/www/html/test.html  
rm: remove regular file '/var/www/html/test.html'? y
```

Рис. 21: Удаление файла

Заключение

Развили навыки администрирования ОС Linux. Получили первое практическое знакомство с технологией SELinux¹. Проверили работу SELinx на практике совместно с веб-сервером Apache.

Библиографическая справка

[1] Мандатное управление: <https://itcloud-edu.ru/info/articles/upravlenie-dostupom-v-gnu-linux/>