

Лабораторная работа №6

Информационная безопасность

Банникова Екатерина Алексеевна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
4	Выводы	14

Список иллюстраций

3.1	getenforce и sestatus	7
3.2	Работающий сервер	8
3.3	Контекст безопасности Apache	9
3.4	Состояние переключателей	10
3.5	Статистика	11
3.6	Данные директорий	11
3.7	Тип файлов	11
3.8	Круг пользователей	12
3.9	Просмотр файла в веб-браузере	12
3.10	Сообщение об ошибке	12
3.11	Прослушивание	13
3.12	Порт	13
3.13	Повторный просмотр файла	13

Список таблиц

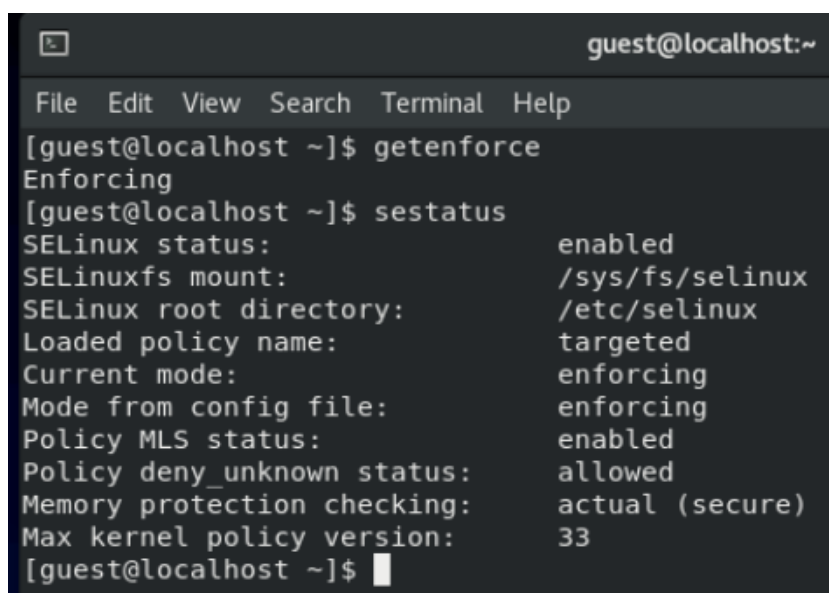
1 Цель работы

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

2 Теоретическое введение

3 Выполнение лабораторной работы

Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`.

A screenshot of a terminal window with a dark background. The title bar at the top right says 'guest@localhost:~'. The terminal has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The user has entered the command `getenforce`, which returns 'Enforcing'. Then, the user enters `sestatus`, which displays the following status information:

```
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
```

The prompt `[guest@localhost ~]$` is visible at the bottom.

Рис. 3.1: `getenforce` и `sestatus`

Обратилась с помощью браузера к веб-серверу, запущенному на компьютере, и убедилась, что последний работает: `service httpd status` или `/etc/rc.d/init.d/httpd status`

```
guest@localhost:~  
File Edit View Search Terminal Help  
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  
[guest@localhost ~]$ service httpd status  
Redirecting to /bin/systemctl status httpd.service  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; vendor preset: enabled)  
   Active: active (running) since Fri 2021-11-26 14:00:00 MSK; 1min 1s ago  
     Docs: man:httpd.service(8)  
  Main PID: 1205 (httpd)  
    Status: "Running, listening on: port 80"  
   Tasks: 213 (limit: 4808)  
  Memory: 10.7M  
    CGroup: /system.slice/httpd.service  
            └─1205 /usr/sbin/httpd -DFOREGROUND  
              └─1281 /usr/sbin/httpd -DFOREGROUND  
                └─1283 /usr/sbin/httpd -DFOREGROUND  
                  └─1284 /usr/sbin/httpd -DFOREGROUND  
                    └─1285 /usr/sbin/httpd -DFOREGROUND  
lines 1-14/14 (END)
```

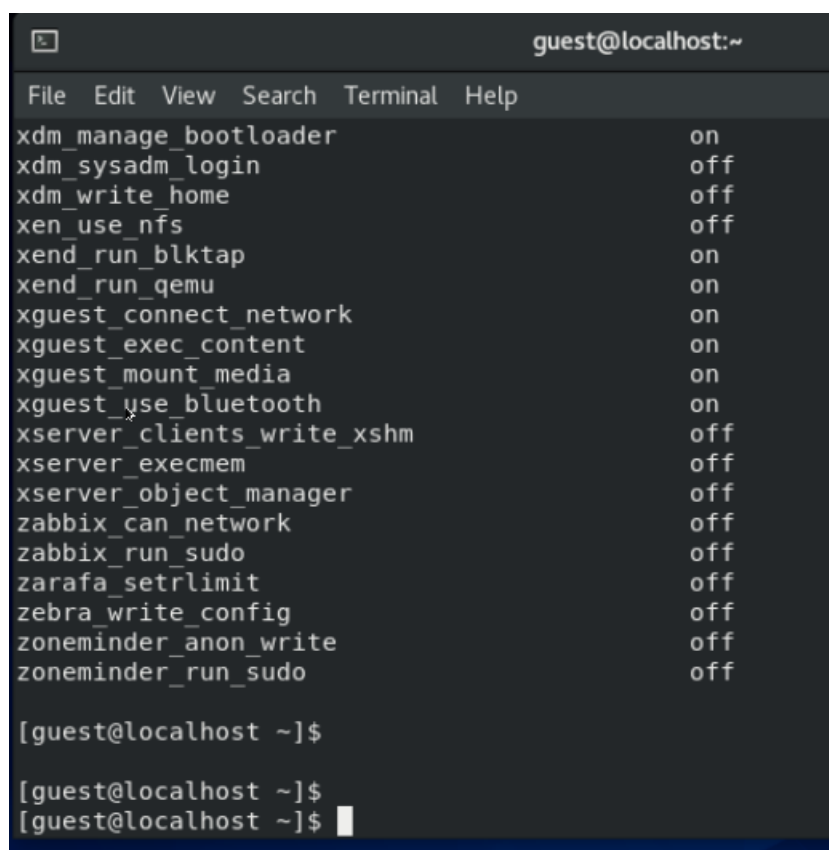
Рис. 3.2: Работающий сервер

Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности и занесла эту информацию в отчёт.


```
guest@localhost:~  
File Edit View Search Terminal Help  
~  
~  
~  
~  
[guest@localhost ~]$ ps auxZ | grep httpd  
system_u:system_r:httpd_t:s0 root 1205 0.0 0.3 275984 3260 ?  
Ss 14:50 0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 1281 0.0 0.2 289868 2256 ?  
S 14:50 0:00 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 1283 0.1 0.3 1347668 2752 ?  
Sl 14:50 0:01 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 1284 0.1 0.3 1478796 2580 ?  
Sl 14:50 0:01 /usr/sbin/httpd -DFOREGROUND  
system_u:system_r:httpd_t:s0 apache 1285 0.1 0.3 1347668 2756 ?  
Sl 14:50 0:01 /usr/sbin/httpd -DFOREGROUND  
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 guest 3451 0.0 0.1 12  
1196 pts/0 S+ 15:10 0:00 grep --color=auto httpd  
[guest@localhost ~]$ ps -eZ | grep httpd  
system_u:system_r:httpd_t:s0 1205 ? 00:00:00 httpd  
system_u:system_r:httpd_t:s0 1281 ? 00:00:00 httpd  
system_u:system_r:httpd_t:s0 1283 ? 00:00:01 httpd  
system_u:system_r:httpd_t:s0 1284 ? 00:00:01 httpd  
system_u:system_r:httpd_t:s0 1285 ? 00:00:01 httpd  
[guest@localhost ~]$
```

Рис. 3.3: Контекст безопасности Apache

Посмотрела текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd`



The screenshot shows a terminal window titled "guest@localhost:~". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The main content is a list of system switches and their states, displayed in a table-like format. The switches are listed on the left, and their states (on or off) are listed on the right. The switches include xdm_manage_bootloader, xdm_sysadm_login, xdm_write_home, xen_use_nfs, xend_run_blktp, xend_run_qemu, xguest_connect_network, xguest_exec_content, xguest_mount_media, xguest_use_bluetooth, xserver_clients_write_xshm, xserver_execmem, xserver_object_manager, zabbix_can_network, zabbix_run_sudo, zarafa_setrlimit, zebra_write_config, zoneminder_anon_write, and zoneminder_run_sudo. The states are either "on" or "off". At the bottom of the terminal, there are three lines of text: "[guest@localhost ~]\$", "[guest@localhost ~]\$", and "[guest@localhost ~]\$".

Switch	State
xdm_manage_bootloader	on
xdm_sysadm_login	off
xdm_write_home	off
xen_use_nfs	off
xend_run_blktp	on
xend_run_qemu	on
xguest_connect_network	on
xguest_exec_content	on
xguest_mount_media	on
xguest_use_bluetooth	on
xserver_clients_write_xshm	off
xserver_execmem	off
xserver_object_manager	off
zabbix_can_network	off
zabbix_run_sudo	off
zarafa_setrlimit	off
zebra_write_config	off
zoneminder_anon_write	off
zoneminder_run_sudo	off

Рис. 3.4: Состояние переключателей

Посмотрела статистику по политике с помощью команды seinfo, также определила множество пользователей, ролей, типов.

```

guest@localhost:~
File Edit View Search Terminal Help
Policy Version:          31 (MLS enabled)
Target Policy:           selinux
Handle unknown classes:  allow
Classes:                 132
Sensitivities:           1
Types:                   4961
Users:                   8
Booleans:                338
Allow:                   112594
Auditallow:              166
Type_trans:              252747
Type_member:             35
Role_allow:              38
Constraints:             72
MLS Constrain:           72
Permissives:             0
Defaults:                7
Allowxperm:              0
Auditallowxperm:         0
Ibendportcon:            0
Initial SIDs:            27
Genfscon:                107
Netifcon:                0
Permissions:             464
Categories:             1024
Attributes:              255
Roles:                   14
Cond. Expr.:             386
Neverallow:              0
Dontaudit:               10358
Type_change:             87
Range_trans:             5781
Role_trans:              421
Validatetrans:           0
MLS Val. Tran:           0
Polcap:                  5
Typebounds:              0
Neverallowxperm:         0
Dontauditxperm:          0
Ibpkeycon:               0
Fs_use:                  34
Portcon:                 642
Nodecon:                 0
[guest@localhost ~]$

```

Рис. 3.5: Статистика

Определила тип файлов и поддиректорий, находящихся в директории /var/www, с помощью команды `ls -lZ /var/www`

```

[guest@localhost ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 Nov 12 07
:58 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Nov 12 07
:58 html
[guest@localhost ~]$

```

Рис. 3.6: Данные директорий

Определила тип файлов, находящихся в директории /var/www/html: `ls -lZ /var/www/html`

```

[guest@localhost ~]$ ls -lZ /var/www/html
total 0

```

Рис. 3.7: Тип файлов

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

```
[guest@localhost html]$ ls -l /var/www/html
total 0
```

Рис. 3.8: Круг пользователей

Создала от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл. Проверила контекст созданного файла.

Обратилась к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Убедилась, что файл был успешно отображён

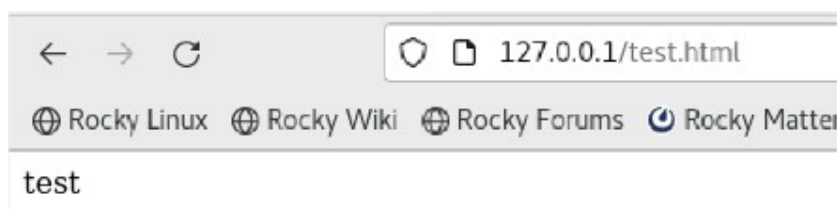


Рис. 3.9: Просмотр файла в веб-браузере

Изучила справку `man httpd_selinux` и выяснила, какие контексты файлов определены для `httpd`. Сопоставила их с типом файла `test.html`. Изменила контекст файла `/var/www/html/test.html`

Попробовала ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес <http://127.0.0.1/test.html>. Получила сообщение об ошибке:

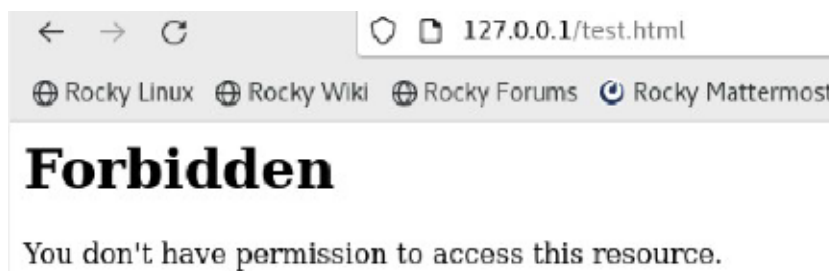


Рис. 3.10: Сообщение об ошибке

Проанализировала ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? Просмотрела log-файлы веб-сервера Apache.

Установила веб-сервер Apache на прослушивание TCP-порта 81, изменяя строку Listen

```
#  
# 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 specific IP addresses as shown below to  
# prevent Apache from glomming onto all bound IP addresses.  
#  
#Listen 12.34.56.78:80  
Listen 01
```

Рис. 3.11: Прослушивание

Перезапустила сервер и посмотрела данные log-файлов веб-сервера Apache. Установила для веб-сервера Apache порт TCP-81 и проверила его наличие

http_port_t	tcp	80, 81, 443, 488, 8088, 8089, 8443, 9000
pegasus_http_port_t	tcp	5988

Рис. 3.12: Порт

Попробовала запустить веб-сервер Apache ещё раз.



Рис. 3.13: Повторный просмотр файла

Удалила привязку http_port_t к 81 порту. Удалила файл test.html

4 Выводы

Развила навыки администрирования ОС Linux. Получила первое практическое знакомство с технологией SELinux¹. Проверила работу SELinx на практике совместно с веб-сервером Apache.