

Отчет по лабораторной работе №6

Основы информационной безопасности

Разанацуа Сара Естэлл, НКАбд-02-23

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	18
	Список литературы	19

Список иллюстраций

2.1	проверка режима работы SELinux	6
2.2	Проверка работы Apache	7
2.3	Контекст безопасности Apache	7
2.4	Состояние переключателей SELinux	8
2.5	Статистика по политике	9
2.6	Типы поддиректорий	9
2.7	Типы файлов	9
2.8	Создание файла	10
2.9	Контекст файла	10
2.10	Отображение файла	10
2.11	Изучение справки по команде	11
2.12	Изменение контекста	12
2.13	Отображение файла	12
2.14	Попытка прочесть лог-файл	13
2.15	Изменение файла	13
2.16	Изменение порта	14
2.17	Попытка прослушивания другого порта	15
2.18	Проверка лог-файлов	15
2.19	Проверка лог-файлов	16
2.20	Проверка портов	16
2.21	Перезапуск сервера	17
2.22	Проверка сервера	17
2.23	Проверка порта 81	17

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. [**course?**]

2 Выполнение лабораторной работы

- Вошла в систему под своей учетной записью. Убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (рис. 2.1).

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

Рис. 2.1: проверка режима работы SELinux

- Запускаю сервер apache, далее обращаюсь с помощью браузера к веб-серверу, запущенному на компьютере, он работает, что видно из вывода команды `service httpd status` (рис. 2.2).

```
[serazanacua@serazanacua ~]$ service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: >
   Active: active (running) since Thu 2025-05-01 18:04:59 MSK; 1min 20s ago
     Docs: man:httpd.service(8)
   Main PID: 35565 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Byt>
   Tasks: 177 (limit: 12175)
  Memory: 30.4M
     CPU: 377ms
   CGroup: /system.slice/httpd.service
           └─35565 /usr/sbin/httpd -DFOREGROUND
             └─35566 /usr/sbin/httpd -DFOREGROUND
               └─35567 /usr/sbin/httpd -DFOREGROUND
                 └─35568 /usr/sbin/httpd -DFOREGROUND
                   └─35569 /usr/sbin/httpd -DFOREGROUND

mai 01 18:04:59 serazanacua systemd[1]: Starting The Apache HTTP Server...
mai 01 18:04:59 serazanacua httpd[35565]: AH00558: httpd: Could not reliably >
mai 01 18:04:59 serazanacua httpd[35565]: Server configured, listening on: po>
mai 01 18:04:59 serazanacua systemd[1]: Started The Apache HTTP Server.
lines 1-20/20 (END)
```

Рис. 2.2: Проверка работы Apache

- С помощью команды `ps auxZ | grep httpd` нашла веб-сервер Apache в списке процессов. Его контекст безопасности - `httpd_t` (рис. 2.3).

```
[serazanacua@serazanacua ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 35565 0.0 0.5 21232 11444 ?
Ss 18:04 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35566 0.0 0.3 22964 7404 ?
S 18:04 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35567 0.1 0.7 2358704 15424 ?
Sl 18:04 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35568 0.1 0.8 2162032 17576 ?
Sl 18:04 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 35569 0.1 0.5 2162032 11064 ?
Sl 18:04 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 serazan+ 35851 0.0 0.1
221676 2432 pts/0 S+ 18:09 0:00 grep --color=auto httpd
[serazanacua@serazanacua ~]$
```

Рис. 2.3: Контекст безопасности Apache

- Просмотрела текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd` (рис. 2.4).

```
[serazanacua@serazanacua ~]$ sestatus -bigrep httpd
sestatus: invalid option -- 'i'

Usage: sestatus [OPTION]

  -v  Verbose check of process and file contexts.
  -b  Display current state of booleans.

Without options, show SELinux status.
[serazanacua@serazanacua ~]$ sestatus -b httpd
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

Policy booleans:
abrt_anon_write                off
abrt_handle_event              off
abrt_upload_watch_anon_write   on
antivirus_can_scan_system      off
antivirus_use_jit              off
auditadm_exec_content          on
authlogin_nsswitch_use_ldap    off
authlogin_radius               off
authlogin_yubikey              off
awstats_purge_apache_log_files off
boinc_execmem                  on
cdrecord_read_content          off
cluster_can_network_connect    off
```

Рис. 2.4: Состояние переключателей SELinux

- Просмотрела статистику по политике с помощью команды `seinfo`. Множество пользователей - 8, ролей - 39, типов - 5135. (рис. 2.5).


```

Statistics for policy file: /sys/fs/selinux/policy
Policy Version:          33 (MLS enabled)
Target Policy:           selinux
Handle unknown classes:  allow
Classes:                 135
Sensitivities:           1
Types:                   5169
Users:                   8
Booleans:                358
Allow:                   65633
Auditallow:              176
Type_trans:              271851
Type_member:             37
Role allow:              40
Constraints:             70
MLS Constrains:          72
Permissives:             1
Defaults:                7
Allowxperm:              0
Auditallowxperm:         0
Ibendportcon:            0
Initial SIDs:            27
Genfscon:                109
Netifcon:                0
Permissions:             457
Categories:              1024
Attributes:              259
Roles:                   15
Cond. Expr.:             390
Neverallow:              0
Dontaudit:               8703
Type_change:             94
Range_trans:             5931
Role_trans:              417
Validatetrans:           0
MLS Val. Tran:           0
Polcap:                  6
Typebounds:              0
Neverallowxperm:         0
Dontauditxperm:          0
Ibpkeycon:               0
Fs_use:                  35
Portcon:                 665
Nodecon:                 0

```

Рис. 2.5: Статистика по политике

- Типы поддиректорий, находящихся в директории /var/www, с помощью команды `ls -lZ /var/www` следующие: владелец - root, права на изменения только у владельца. Файлов в директории нет (рис. 2.6).

```

[serazanacua@serazanacua ~]$ ls -lZ /var/www
total 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 22 janv
. 03:25 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 22 janv
. 03:25 html
[serazanacua@serazanacua ~]$

```

Рис. 2.6: Типы поддиректорий

В директории /var/www/html нет файлов. (рис. 2.7).

```

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

```

Рис. 2.7: Типы файлов

- Создать файл может только суперпользователь, поэтому от его имени создаем файл `touch.html` со следующим содержанием:

```
<html>
<body>test</body>
</html>
```

(рис. 2.8).

```
[serazanacua@serazanacua ~]$ sudo touch /var/www/html/test.html
[serazanacua@serazanacua ~]$ sudo nano /var/www/html/test.html
[serazanacua@serazanacua ~]$ sudo cat /var/www/html/test.html
<html>
<body>test</body>
</html>
[serazanacua@serazanacua ~]$
```

Рис. 2.8: Создание файла

- Проверяю контекст созданного файла. По умолчанию это httpd_sys_content_t (рис. 2.9).

```
[serazanacua@serazanacua ~]$ ls -lZ /var/www/html
total 4
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 1 mai
18:22 test.html
[serazanacua@serazanacua ~]$
```

Рис. 2.9: Контекст файла

- Обращаюсь к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Файл был успешно отображён (рис. 2.10).

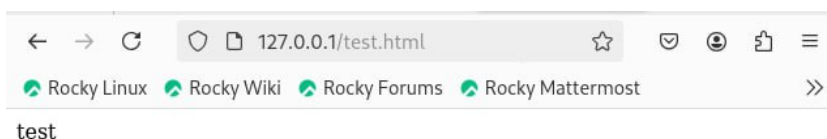


Рис. 2.10: Отображение файла

Изучила справку `man httpd_selinux`. Рассмотрим полученный контекст детально. Так как по умолчанию пользователи CentOS являются свободными от типа (`unconfined` в переводе с англ. означает свободный), созданному нами файлу `test.html` был сопоставлен SELinux, пользователь `unconfined_u`. Это первая часть

контекста. Далее политика ролевого разделения доступа RBAC используется процессами, но не файлами, поэтому роли не имеют никакого значения для файлов. Роль `object_r` используется по умолчанию для файлов на «постоянных» носителях и на сетевых файловых системах. (В директории `/ргос` файлы, относящиеся к процессам, могут иметь роль `system_r`. Если активна политика MLS, то могут использоваться и другие роли, например, `secadm_r`. Данный случай мы рассматривать не будем, как и предназначение `:s0`). Тип `httpd_sys_content_t` позволяет процессу `httpd` получить доступ к файлу. Благодаря наличию последнего типа мы получили доступ к файлу при обращении к нему через браузер. (рис. 2.11).

```

serazanacua@serazanacua:~ — sudo man httpd selinux
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
        Set the initial value for the ServerRoot directive to serverroot. This can be overridden by the ServerRoot directive in the configuration file. The default is /etc/httpd.

    -f config
        Uses the directives in the file config on startup. If config does not begin with a /, then it is taken to be a path relative to the ServerRoot. The default is conf/httpd.conf.

    -k start|restart|graceful|stop|graceful-stop
        Signals httpd to start, restart, or stop. See Stopping Apache httpd for more information.

Manual page httpd(8) line 1 (press h for help or q to quit)

```

Рис. 2.11: Изучение справки по команде

Изменяю 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`: `chcon -t samba_share_t /var/www/html/test.html ls -Z /var/www/html/test.html` Контекст действительно поменялся (рис. 2.12).

```
[serazanacua@serazanacua ~]$ sudo chcon -t samba_share_t /var/www/html/test.html
[serazanacua@serazanacua ~]$ ls -lZ /var/www/html
total 4
-rw-r--r--. 1 root root unconfined_u:object_r:samba_share_t:s0 33 1 mai 18:
22 test.html
[serazanacua@serazanacua ~]$
```

Рис. 2.12: Изменение контекста

- При попытке отображения файла в браузере получаем сообщение об ошибке (рис. 2.13).

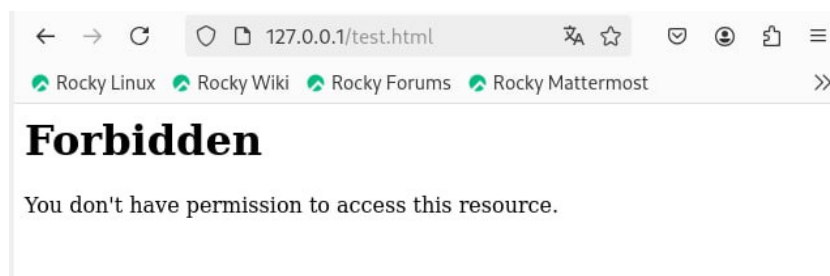


Рис. 2.13: Отображение файла

файл не был отображён, хотя права доступа позволяют читать этот файл любому пользователю, потому что установлен контекст, к которому процесс `httpd` не должен иметь доступа.

Просматриваю `log`-файлы веб-сервера `Apache` и системный `log`-файл: `tail /var/log/messages`. Если в системе окажутся запущенными процессы `setroubleshootd` и `audtd`, то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле `/var/log/audit/audit.log`. (рис. 2.14).

```

[serazanacua@serazanacua ~]$ ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 1 mai 18:22 /var/www/html/test.html
[serazanacua@serazanacua ~]$ tail /var/log/messages
tail: impossible d'ouvrir '/var/log/messages' en lecture: Permission non accordée
[serazanacua@serazanacua ~]$ sudo tail /var/log/messages
May 1 18:42:13 serazanacua systemd[1]: Started SETroubleshoot daemon for processing new SELinux denial logs.
May 1 18:42:13 serazanacua setroubleshoot[36807]: failed to retrieve rpm info for path '/var/www/html/test.html':
May 1 18:42:13 serazanacua systemd[1]: Created slice Slice /system/dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged.
May 1 18:42:13 serazanacua systemd[1]: Started dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@0.service.
May 1 18:42:15 serazanacua setroubleshoot[36807]: SELinux interdit à /usr/sbin/httpd d'utiliser l'accès getattr sur le fichier /var/www/html/test.html. Pour des messages SELinux exhaustifs, lancez sealert -l e7f2aa8e-a658-49e7-9d46-6af64e944208
May 1 18:42:15 serazanacua setroubleshoot[36807]: SELinux interdit à /usr/sbin/httpd d'utiliser l'accès getattr sur le fichier /var/www/html/test.html.#012#012***** Le greffon restorecon (92.2 de confiance) suggère *****
*****#012#012Si vous souhaitez corriger l'étiquette. #012L'étiquette par défaut de /var/www/html/test.html devrait être httpd_sys_content_t.#012Alors vous pouvez lancer restorecon. La tentative d'accès pourrait avoir été stoppée due à des permissions insuffisantes d'accès au dossier parent, auquel cas essayez de changer la commande suivante en conséquence.#012Faire#012# /sbin/restorecon -v /var/www/html/test.html#012#012***** Le greffon public_content (7.83 de confiance) suggère *****#012#012Si vous souhaitez considérer test.html comme contenu public#012Alors vous devez modifier l'étiquette de test.html en public_content_t ou public_content_rw_t.#012Faire#012# semanage fcontext -a -t public_content_t '/var/www/html/test.html' #012# restorecon -v '/var/www/html/test.html' #012#012***** Le greffon catchall (1.41 de confiance) suggère *
*****#012#012Si vous pensez que httpd devrait être autorisé à accéder getattr sur test.html file par défaut.#012Alors vous devriez rapporter ceci en tant qu'anomalie.#012Vous pouvez générer un module de stratégie local pour autoriser cet accès.#012Faire#012autoriser cet accès pour le moment en exécutant :#012# ausearch -c "httpd" --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
May 1 18:42:15 serazanacua setroubleshoot[36807]: SELinux interdit à /usr/sbin/httpd d'utiliser l'accès getattr sur le fichier /var/www/html/test.html. Pour des messages SELinux exhaustifs, lancez sealert -l e7f2aa8e-a658-49e7-9d46-6af64e944208

```

Рис. 2.14: Попытка прочесть лог-файл

- Чтобы запустить веб-сервер Apache на прослушивание TCP-порта 81 (а не 80, как рекомендует IANA и прописано в /etc/services) открываю файл /etc/httpd/httpd.conf для изменения. (рис. 2.15).

```

[serazanacua@serazanacua ~]$ sudo nano /etc/httpd/conf/httpd.conf
[serazanacua@serazanacua ~]$ █

```

Рис. 2.15: Изменение файла

- Нахожу строчку Listen 80 и заменяю её на Listen 81. (рис. 2.16).

```
GNU nano 5.6.1 /etc/httpd/conf/httpd.conf Modifié
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path.  If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so 'log/access_log'
# with ServerRoot set to '/www' will be interpreted by the
# server as '/www/log/access_log', where as '/log/access_log' will be
# interpreted as '/log/access_log'.
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path.  If you point
# ServerRoot at a non-local disk, be sure to specify a local disk on the
# Mutex directive, if file-based mutexes are used.  If you wish to share the
# same ServerRoot for multiple httpd daemons, you will need to change at
# least PidFile.
#
ServerRoot "/etc/httpd"
#
# 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 a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts. See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 81
#
# Dynamic Shared Object (DSO) Support
#
```

Рис. 2.16: Изменение порта

- Выполняя перезапуск веб-сервера Apache. Произошёл сбой, потому что порт 80 для локальной сети, а 81 нет (рис. 2.17).

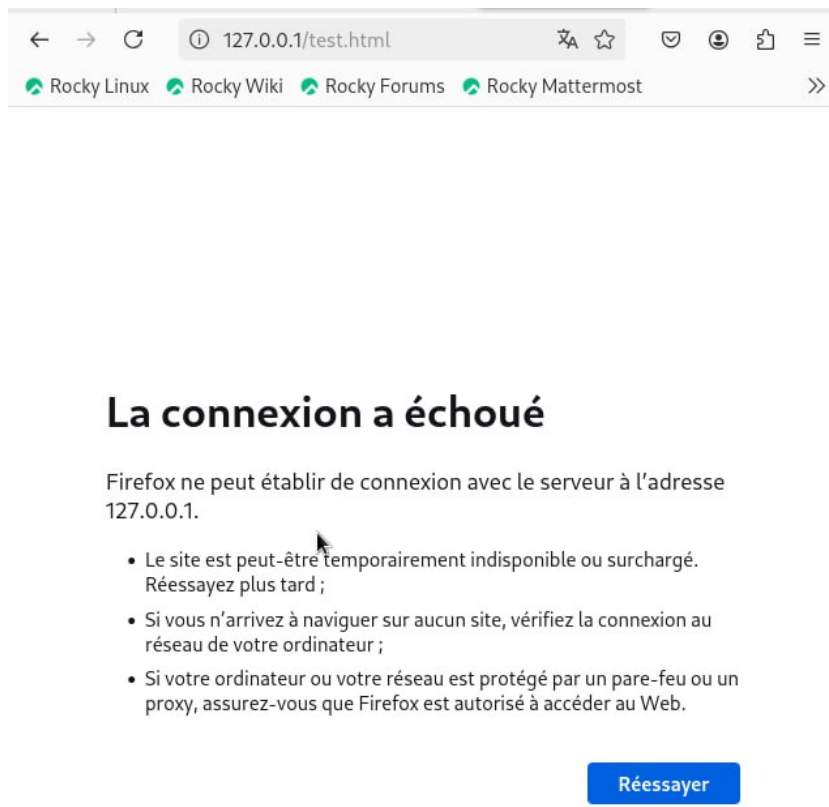


Рис. 2.17: Попытка прослушивания другого порта

- Проанализируйте лог-файлы: `tail -nl /var/log/messages` (рис. 2.18).

```
[serazanacua@serazanacua ~]$ sudo tail -nl /var/log/messages
May  1 19:00:49 serazanacua systemd[1]: Started The Apache HTTP Server.
```

Рис. 2.18: Проверка лог-файлов

Просмотрите файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и выясните, в каких файлах появились записи. Запись появилась в файлу `error_log` (рис. 2.19).


```
[serazanacua@serazanacua ~]$ sudo cat /var/log/httpd/error_log
[Thu May 01 18:04:59.537030 2025] [core:notice] [pid 35565:tid 35565] SELinux
policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Thu May 01 18:04:59.537645 2025] [suexec:notice] [pid 35565:tid 35565] AH0123
2: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain
name, using fe80::a00:27ff:fedc:f96e%enp0s3. Set the 'ServerName' directive
globally to suppress this message
[Thu May 01 18:04:59.560575 2025] [lbmethod_heartbeat:notice] [pid 35565:tid 3
5565] AH02282: No slotmem from mod_heartbeat
[Thu May 01 18:04:59.567662 2025] [mpm_event:notice] [pid 35565:tid 35565] AH0
0489: Apache/2.4.62 (Rocky Linux) configured -- resuming normal operations
[Thu May 01 18:04:59.567700 2025] [core:notice] [pid 35565:tid 35565] AH00094:
Command line: '/usr/sbin/httpd -D FOREGROUND'
[Thu May 01 18:42:12.966109 2025] [core:error] [pid 35567:tid 35717] (13)Permi
ssion denied: [client 127.0.0.1:44322] AH00035: access to /test.html denied (f
ilesystem path '/var/www/html/test.html') because search permissions are missi
ng on a component of the path
[Thu May 01 18:59:44.111297 2025] [core:error] [pid 35569:tid 35686] (13)Permi
ssion denied: [client 127.0.0.1:60544] AH00035: access to /test.html denied (f
ilesystem path '/var/www/html/test.html') because search permissions are missi
ng on a component of the path
[Thu May 01 18:59:46.301031 2025] [core:error] [pid 35569:tid 35689] (13)Permi
ssion denied: [client 127.0.0.1:60544] AH00035: access to /test.html denied (f
ilesystem path '/var/www/html/test.html') because search permissions are missi
ng on a component of the path
[Thu May 01 19:00:48.330435 2025] [mpm_event:notice] [pid 35565:tid 35565] AH0
0492: caught SIGWINCH, shutting down gracefully
[Thu May 01 19:00:49.500676 2025] [core:notice] [pid 37164:tid 37164] SELinux
policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Thu May 01 19:00:49.502450 2025] [suexec:notice] [pid 37164:tid 37164] AH0123
2: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain
name, using fe80::a00:27ff:fedc:f96e%enp0s3. Set the 'ServerName' directive
globally to suppress this message
[Thu May 01 19:00:49.528829 2025] [lbmethod_heartbeat:notice] [pid 37164:tid 3
7164] AH02282: No slotmem from mod_heartbeat
[Thu May 01 19:00:49.537114 2025] [mpm_event:notice] [pid 37164:tid 37164] AH0
0489: Apache/2.4.62 (Rocky Linux) configured -- resuming normal operations
[Thu May 01 19:00:49.537172 2025] [core:notice] [pid 37164:tid 37164] AH00094:
Command line: '/usr/sbin/httpd -D FOREGROUND'
[serazanacua@serazanacua ~]$
```

Рис. 2.19: Проверка лог-файлов

Выполняю команду `semanage port -a -t http_port_t -p tcp 81` После этого проверяю список портов командой `semanage port -l | grep http_port_t` Порт 81 появился в списке (рис. 2.20).

```
[serazanacua@serazanacua ~]$ 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
[serazanacua@serazanacua ~]$ semanage port -l | grep http_port_t
ValueError: La stratégie SELinux n'est pas gérée ou la base n'est pas accessible.
[serazanacua@serazanacua ~]$ sudo 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
[serazanacua@serazanacua ~]$
```

Рис. 2.20: Проверка портов

- Перезапускаю сервер Apache (рис. 2.21).

```
[serazanacua@serazanacua ~]$ sudo systemctl restart httpd
[sudo] Mot de passe de serazanacua :
[serazanacua@serazanacua ~]$ sudo chcon -t httpd_sys_content_t /var/www/html/test.html
[serazanacua@serazanacua ~]$ sudo systemctl restart httpd
[serazanacua@serazanacua ~]$
```

Рис. 2.21: Перезапуск сервера

- Теперь он работает, ведь мы внесли порт 81 в список портов httpd_port_t (рис. 2.22).

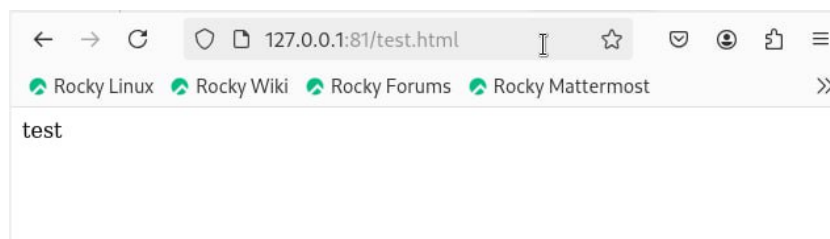


Рис. 2.22: Проверка сервера

- Возвращаю в файле /etc/httpd/httpd.conf порт 80, вместо 81. Проверяю, что порт 81 удален, это правда. (рис. 2.23).

```
[serazanacua@serazanacua ~]$ sudo nano /etc/httpd/conf/httpd.conf
[serazanacua@serazanacua ~]$ sudo semanage port -d -t http_port_t -p tcp 81
ValueError: Le port tcp/81 est défini dans la politique, il ne peut être supprimé
[serazanacua@serazanacua ~]$ semanage port -d -t http_port_t -p tcp 81
ValueError: La stratégie SELinux n'est pas gérée ou la base n'est pas accessible.
[serazanacua@serazanacua ~]$ sudo semanage port -d -t http_port_t -p tcp 81
ValueError: Le port tcp/81 est défini dans la politique, il ne peut être supprimé
[serazanacua@serazanacua ~]$
```

Рис. 2.23: Проверка порта 81

3 Выводы

- В ходе выполнения данной лабораторной работы были развиты навыки администрирования ОС Linux, получено первое практическое знакомство с технологией SELinux и проверена работа SELinux на практике совместно с веб-сервером Apache.

Список литературы