

Отчёта по лабораторной работе №6

**Мандатное разграничение прав в Linux (SELinux) с использованием
веб-сервера Apache**

Акондзо Жордани Лади Гаэл

Содержание

1	Цель работы	5
2	Теоретическое введение	6
2.1	Подготовка стенда	6
3	Выполнение лабораторной работы	7
4	Выводы	22

Список иллюстраций

3.1	Проверка статуса SELinux	7
3.2	Настройка веб-сервера ApacheПроверка статуса SELinux	8
3.3	Проверка контекста SELinux для Apache	8
3.4	Проверка текущих настроек SELinux для Apache	9
3.5	Анализ политик и типов SELinux	10
3.6	Анализ политик и типов SELinux	10
3.7	Определение типа файлов и поддиректорий	11
3.8	Определение типа файлов и поддиректорий	11
3.9	Определение круга пользователей	12
3.10	Создание тестового файла и проверка доступа	12
3.11	Проверка статуса SELinux	12
3.12	Проверка контекста созданного файла	13
3.13	Тестирование работы веб-сервера	13
3.14	Анализ контекста файлов	14
3.15	Изменение контекста безопасности	14
3.16	Проверка блокировки доступа	15
3.17	Анализ ситуаций	15
3.18	Замена порта 80 на 81 для Apache в SELinux	16
3.19	Перезапуск веб-сервера Apache	17
3.20	Анализ лог-файлов	17
3.21	Добавление порта 81 для Apache в SELinux	18
3.22	Проверка списка портов	18
3.23	Перезапуск Apache	19
3.24	Вернул контекст httpd_sys_content_t	20
3.25	Удаление привязки порта 81 к Apache	20
3.26	Удаление тестового файла	21

List of Tables

1 Цель работы

Цель данной лабораторной работы — развить навыки администрирования в операционной системе Linux с акцентом на использование технологии SELinux. Основное внимание уделено настройке SELinux для работы с веб-сервером Apache, что позволяет на практике проверить ограничения прав доступа.

2 Теоретическое введение

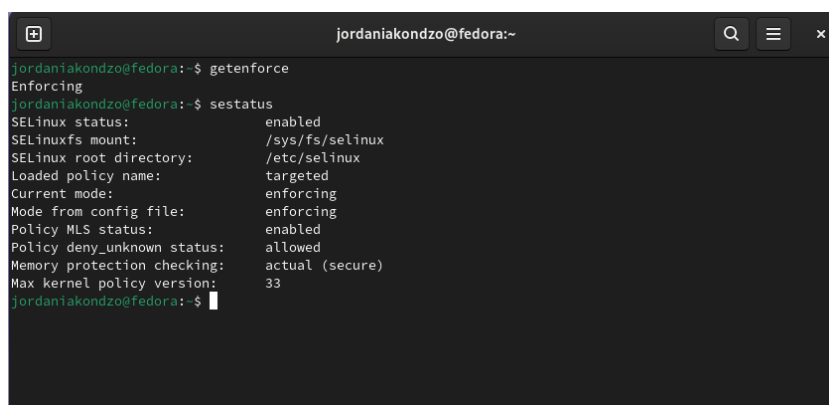
2.1 Подготовка стенда

Для выполнения лабораторной работы использовался дистрибутив Linux с включённой политикой SELinux targeted и режимом enforcing. В качестве веб-сервера использовался Apache, который был настроен для работы на портах 80 и 81. Важно было убедиться, что iptables настроен корректно и не блокирует доступ к данным портам.

3 Выполнение лабораторной работы

1. Проверка статуса SELinux

- В первую очередь, я проверил, что SELinux работает в режиме enforcing с использованием следующей команды: (рис. 3.1)

A screenshot of a terminal window titled 'jordaniakondzo@fedora:~'. The terminal shows the command 'getenforce' being executed, which returns 'Enforcing'. Then, the command 'sestatus' is executed, displaying the following status information:

```
jordaniakondzo@fedora:~$ getenforce
Enforcing
jordaniakondzo@fedora:~$ 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
jordaniakondzo@fedora:~$
```

Рис. 3.1: Проверка статуса SELinux

- Результат показал, что система настроена корректно для выполнения лабораторной работы.

2. Настройка веб-сервера Apache

- После этого я убедился, что сервер Apache работает, используя команду: (рис. 3.2)

```
jordaniakondzo@fedora:~ — /bin/systemctl status httpd.service
jordaniakondzo@fedora:~$ 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)
   Drop-In: /usr/lib/systemd/system/service.d
            └─10-timeout-abort.conf
   Active: active (running) since Thu 2024-10-10 23:15:42 MSK; 13h ago
     Docs: man:httpd.service(8)
   Main PID: 8579 (httpd)
   Status: "Total requests: 5; Idle/Busy workers 100/0; Requests/sec: 0.000104; Bytes served/sec: 0"
     Tasks: 230 (limit: 4645)
   Memory: 17.2M (peak: 20.1M swap: 5.8M swap peak: 5.8M)
     CPU: 11.116s
   CGroup: /system.slice/httpd.service
           └─8579 /usr/sbin/httpd -DFOREGROUND
             └─8586 /usr/sbin/httpd -DFOREGROUND
               └─8588 /usr/sbin/httpd -DFOREGROUND
                 └─8589 /usr/sbin/httpd -DFOREGROUND
                   └─8635 /usr/sbin/httpd -DFOREGROUND
                     └─11458 /usr/sbin/httpd -DFOREGROUND

oct. 10 23:15:34 fedora systemd[1]: Starting httpd.service - The Apache HTTP Server...
oct. 10 23:15:36 fedora httpd[8579]: AH00558: httpd: Could not reliably determine the server's fully q
oct. 10 23:15:42 fedora httpd[8579]: Server configured, listening on: port 80
oct. 10 23:15:42 fedora systemd[1]: Started httpd.service - The Apache HTTP Server.
lines 1-24/24 (END)
```

Рис. 3.2: Настройка веб-сервера ApacheПроверка статуса SELinux

3. Проверка контекста SELinux для Apache

- Я проверил контексты безопасности для процессов Apache, используя команду: (рис. 3.3)

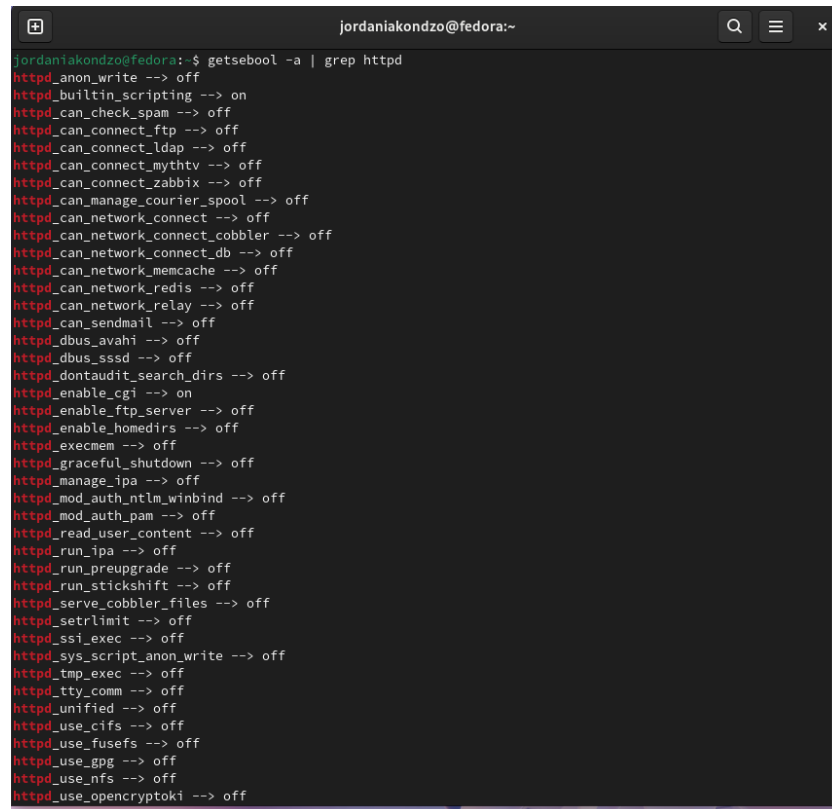
```
jordaniakondzo@fedora:~$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root      8579  0.0  0.2 20020  8392 ?        Ss   10:50   0:01 /usr
/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  8586  0.0  0.0 19732  2576 ?        S    10:50   0:00 /usr
/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  8588  0.0  0.1 1569012 4360 ?        Sl   10:50   0:03 /usr
/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  8589  0.0  0.1 1437908 6012 ?        Sl   10:50   0:02 /usr
/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  8635  0.0  0.1 1437908 4496 ?        Sl   10:50   0:02 /usr
/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 11458  0.0  0.1 1437908 6956 ?        Sl   11:53   0:01 /usr
/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 jordani+ 13740  0.0  0.0 227808 2400 pts/0 S+   12:4
1   0:00 grep --color=auto httpd
jordaniakondzo@fedora:~$ ps -eZ | grep httpd
system_u:system_r:httpd_t:s0      8579 ?           00:00:01 httpd
system_u:system_r:httpd_t:s0      8586 ?           00:00:00 httpd
system_u:system_r:httpd_t:s0      8588 ?           00:00:03 httpd
system_u:system_r:httpd_t:s0      8589 ?           00:00:02 httpd
system_u:system_r:httpd_t:s0      8635 ?           00:00:02 httpd
system_u:system_r:httpd_t:s0     11458 ?           00:00:01 httpd
jordaniakondzo@fedora:~$
```

Рис. 3.3: Проверка контекста SELinux для Apache

- Эта команда показала, что процессы Apache работают с контекстом httpd_t, что позволяет серверу правильно обрабатывать веб-запросы.

4. Проверка текущих настроек SELinux для Apache

- Далее, я использовал команду `sestatus` для проверки настроек SELinux, связанных с веб-сервером Apache: (рис. 3.4)



```
jordaniakondzo@fedora:~  
jordaniakondzo@fedora:~$ getsebool -a | grep httpd  
httpd_anon_write --> off  
httpd_built_in_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_redis --> off  
httpd_can_network_relay --> off  
httpd_can_sendmail --> off  
httpd_dbus_avaahi --> off  
httpd_dbus_sss --> 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  
httpd_use_openssl --> off
```

Рис. 3.4: Проверка текущих настроек SELinux для Apache

- Это позволило мне увидеть, какие функции включены для Apache.

5. Анализ политик и типов SELinux

- Я также изучил статистику политик и список пользователей, ролей и типов, используя команду: (рис. 3.5) и (рис. 3.6)

```
jordaniakondzo@fedora:~  
jordaniakondzo@fedora:~$ seinfo  
Statistics for policy file: /sys/fs/selinux/policy  
Policy Version: 33 (MLS enabled)  
Target Policy: selinux  
Handle unknown classes: allow  
Classes: 134 Permissions: 460  
Sensitivities: 1 Categories: 1024  
Types: 5262 Attributes: 264  
Users: 8 Roles: 15  
Booleans: 365 Cond. Expr.: 398  
Allow: 68036 Neverallow: 0  
Auditallow: 181 Dontaudit: 8829  
Type_trans: 284188 Type_change: 94  
Type_member: 37 Range_trans: 6164  
Role_allow: 40 Role_trans: 419  
Constraints: 70 Validatetrans: 0  
MLS Constrain: 72 MLS Val. Tran: 0  
Permissives: 9 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: 110 Portcon: 665  
Netifcon: 0 Nodecon: 0  
jordaniakondzo@fedora:~$
```

Рис. 3.5: Анализ политик и типов SELinux

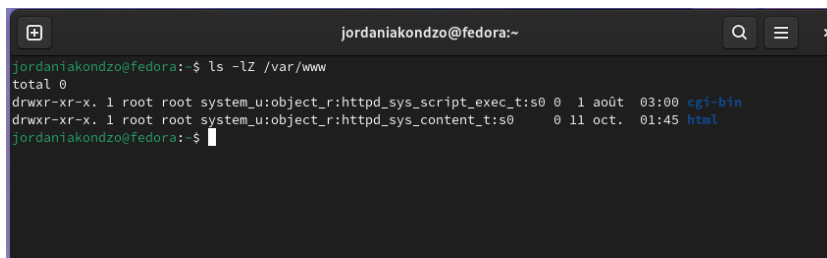
```
jordaniakondzo@fedora:~  
jordaniakondzo@fedora:~$ seinfo -u  
Users: 8  
guest_u  
root  
staff_u  
sysadm_u  
system_u  
unconfined_u  
user_u  
xguest_u  
jordaniakondzo@fedora:~$ seinfo -r  
Roles: 15  
auditadm_r  
container_user_r  
dbadm_r  
guest_r  
logadm_r  
nx_server_r  
object_r  
secadm_r  
staff_r  
sysadm_r  
system_r  
unconfined_r  
user_r  
webadm_r  
xguest_r  
jordaniakondzo@fedora:~$ seinfo -t  
Types: 5262  
NetworkManager_dispatcher_chronyc_script_t  
NetworkManager_dispatcher_chronyc_t  
NetworkManager_dispatcher_cloud_script_t  
NetworkManager_dispatcher_cloud_t  
NetworkManager_dispatcher_console_script_t  
NetworkManager_dispatcher_console_t  
NetworkManager_dispatcher_console_var_run_t  
NetworkManager_dispatcher_custom_t  
NetworkManager_dispatcher_ddclient_script_t  
NetworkManager_dispatcher_ddclient_t  
NetworkManager_dispatcher_dhclient_script_t
```

Рис. 3.6: Анализ политик и типов SELinux

- Это позволило мне получить полную картину того, как SELinux управляет

доступом для Apache.

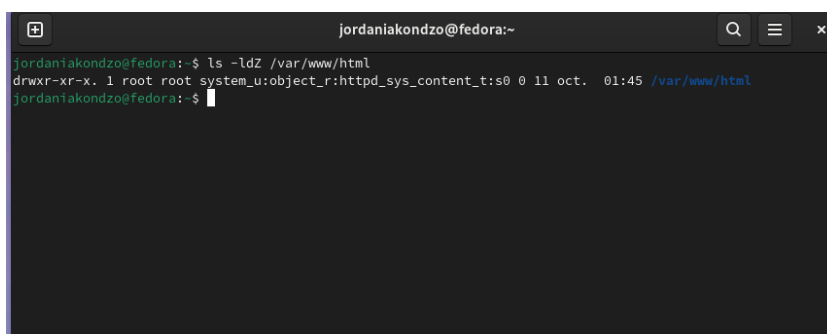
6. Определение типа файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды: (рис. 3.7)



```
jordaniakondzo@fedora:~  
jordaniakondzo@fedora:~$ ls -lZ /var/www  
total 0  
drwxr-xr-x. 1 root root system_u:object_r:httpd_sys_script_exec_t:s0 0 1 août 03:00 cgi-bin  
drwxr-xr-x. 1 root root system_u:object_r:httpd_sys_content_t:s0 0 11 oct. 01:45 html  
jordaniakondzo@fedora:~$
```

Рис. 3.7: Определение типа файлов и поддиректорий

7. Определение типа файлов, находящихся в директории `/var/www/html`: (рис. 3.8)

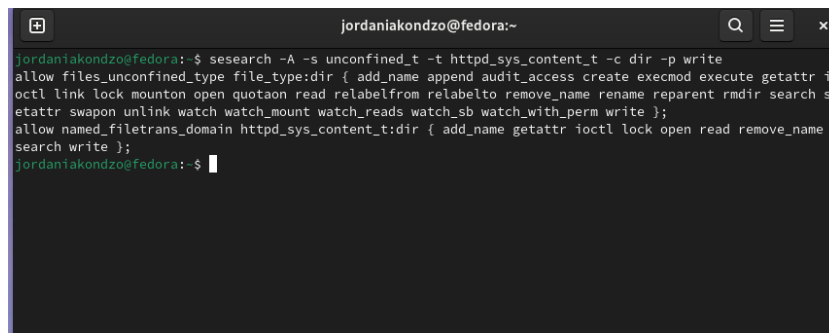


```
jordaniakondzo@fedora:~  
jordaniakondzo@fedora:~$ ls -ldZ /var/www/html  
drwxr-xr-x. 1 root root system_u:object_r:httpd_sys_content_t:s0 0 11 oct. 01:45 /var/www/html  
jordaniakondzo@fedora:~$
```

Рис. 3.8: Определение типа файлов и поддиректорий

8. Определение круга пользователей

- Я определил круг пользователей, которым разрешено создание файлов в директории `/var/www/html`. (рис. 3.9)



```
jordaniakondzo@fedora:~  
jordaniakondzo@fedora:~$ sesearch -A -s unconfined_t -t httpd_sys_content_t -c dir -p write  
allow files_unconfined_type file_type:dir { add_name append audit_access create execmod execute getattr i  
octl link lock mounton open quotaon read relabelfrom relabelto remove_name rename reparent rmdir search s  
etattr swapon unlink watch watch_mount watch_reads watch_sb watch_with_perm write };  
allow named_filetrans_domain httpd_sys_content_t:dir { add_name getattr ioctl lock open read remove_name  
search write };  
jordaniakondzo@fedora:~$
```

Рис. 3.9: Определение круга пользователей

9. Создание тестового файла и проверка доступа

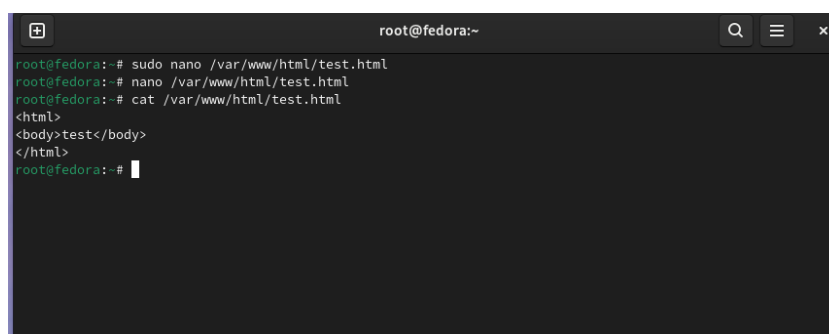
- Я создал тестовый файл test.html в каталоге /var/www/html, используя ко-манду: (рис. 3.10)



```
root@fedora:~  
GNU nano 7.2 /var/www/html/test.html  
<html>  
<body>test</body>  
</html>
```

Рис. 3.10: Создание тестового файла и проверка доступа

- Содержание файла: (рис. 3.11)

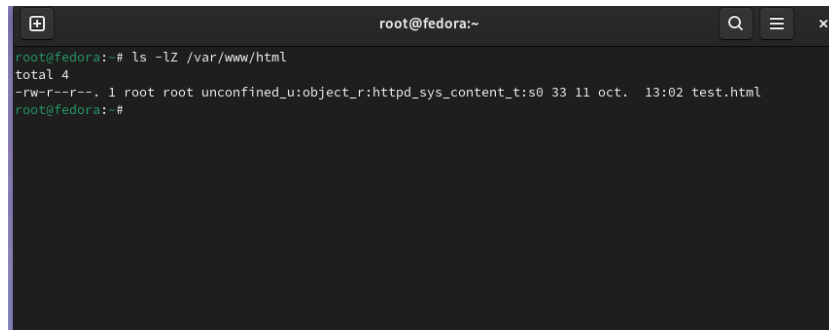


```
root@fedora:~  
root@fedora:~# sudo nano /var/www/html/test.html  
root@fedora:~# nano /var/www/html/test.html  
root@fedora:~# cat /var/www/html/test.html  
<html>  
<body>test</body>  
</html>  
root@fedora:~#
```

Рис. 3.11: Проверка статуса SELinux

10. Проверка контекста созданного файла

- Затем я проверил контекст этого файла командой: (рис. 3.12)



```
root@fedora:~  
root@fedora:~# ls -lZ /var/www/html  
total 4  
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 11 oct. 13:02 test.html  
root@fedora:~#
```

Рис. 3.12: Проверка контекста созданного файла

- Контекст был установлен как `httpd_sys_content_t`, что позволило серверу Apache получить доступ к файлу через браузер.

11. Тестирование работы веб-сервера

- Я открыл браузер и перешел по адресу: (рис. 3.13)

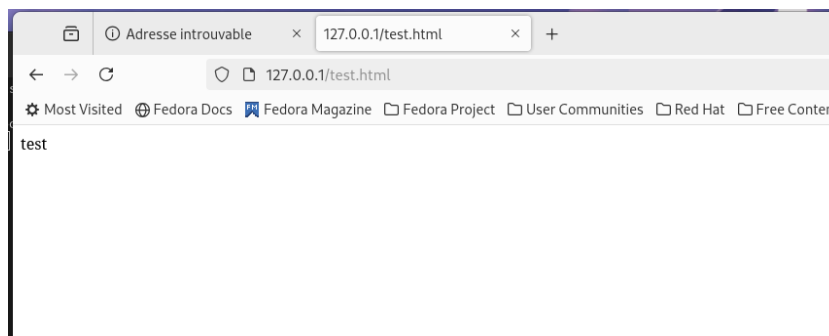
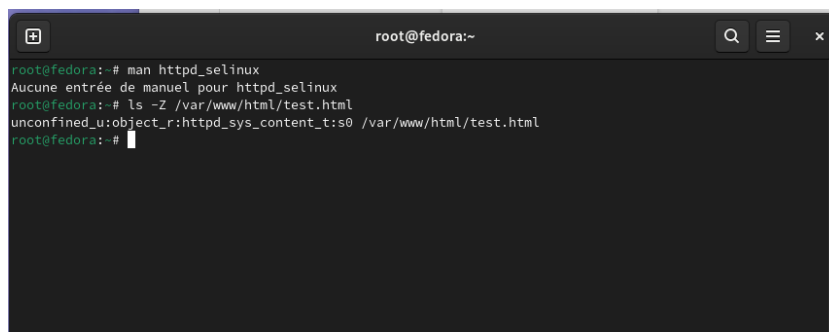


Рис. 3.13: Тестирование работы веб-сервера

- Файл успешно отобразился, что подтвердило правильность настроек SELinux для данного файла.

12. Анализ контекста файлов

- Я изучил контексты, определенные для файлов Apache. Контекст `httpd_sys_content_t` позволяет серверу Apache получать доступ к файлам, но не выполнять их или модифицировать без специальных разрешений. (рис. 3.14)

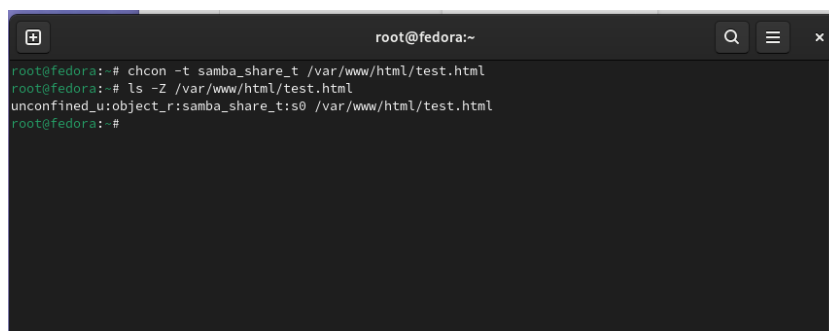


```
root@fedora:~  
root@fedora: # man httpd_selinux  
Aucune entrée de manuel pour httpd_selinux  
root@fedora: # ls -Z /var/www/html/test.html  
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html  
root@fedora: #
```

Рис. 3.14: Анализ контекста файлов

13. Изменение контекста безопасности

- Я изменил контекст безопасности файла `test.html` на `samba_share_t`, чтобы проверить, как изменятся права доступа: (рис. 3.15)



```
root@fedora:~  
root@fedora: # chcon -t samba_share_t /var/www/html/test.html  
root@fedora: # ls -Z /var/www/html/test.html  
unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html  
root@fedora: #
```

Рис. 3.15: Изменение контекста безопасности

14. Проверка блокировки доступа

- После изменения контекста на `samba_share_t`, я снова попытался открыть файл через браузер, но получил ошибку **403 Forbidden**. Это показало, что SELinux эффективно блокирует доступ к файлу с неподходящим контекстом. (рис. 3.16)

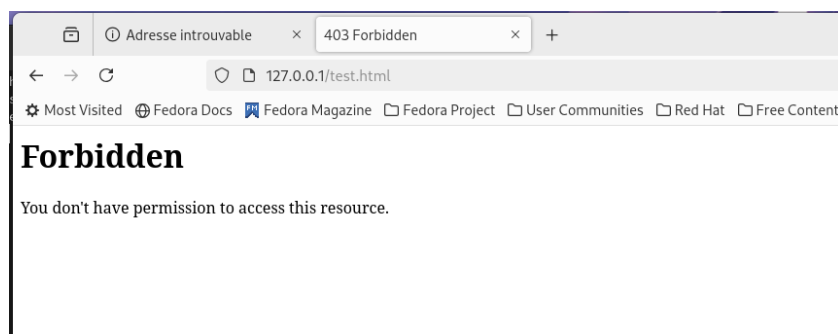


Рис. 3.16: Проверка блокировки доступа

- После этого проверил, что контекст поменялся.

15. Анализ ситуаций

- Я проанализировал ситуацию с помощью следующих команд: (рис. 3.17)

```

root@fedora:~
root@fedora:~# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 11 oct. 13:02 /var/www/html/test.html
root@fedora:~# tail /var/log/messages
tail: impossible d'ouvrir '/var/log/messages' en lecture: Aucun fichier ou dossier de ce nom
root@fedora:~# tail /var/log/httpd/error_log
[Thu Oct 10 23:15:36.216514 2024] [suexec:notice] [pid 8579:tid 8579] AH01232: suEXEC mechanism enabled (
wrapper: /usr/sbin/suexec)
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using fe80::e025:8
0df:3d38:1a9%enp0s3. Set the 'ServerName' directive globally to suppress this message
[Thu Oct 10 23:15:42.182180 2024] [lbmethod_heartbeat:notice] [pid 8579:tid 8579] AH02282: No slotmem fro
m mod_heartbeat
[Thu Oct 10 23:15:42.190237 2024] [systemd:notice] [pid 8579:tid 8579] SELinux policy enabled; httpd runn
ing as context system_u:system_r:httpd_t:s0
[Thu Oct 10 23:15:42.217776 2024] [warn] [pid 8587:tid 8587] ./mod_dnssd.c: No services found to regist
e
r
[Thu Oct 10 23:15:42.230686 2024] [mpm_event:notice] [pid 8579:tid 8579] AH00489: Apache/2.4.62 (Fedora L
inux) configured -- resuming normal operations
[Thu Oct 10 23:15:42.230741 2024] [core:notice] [pid 8579:tid 8579] AH00094: Command line: '/usr/sbin/htt
pd -D FOREGROUND'
[Fri Oct 11 00:50:02.812031 2024] [core:error] [pid 11458:tid 11487] (13)Permission denied: [client 127.0
.0.1:50252] AH00035: access to /test.html denied (filesystem path '/var/www/html/test.html') because sear
ch permissions are missing on a component of the path
[Fri Oct 11 13:29:05.272422 2024] [core:error] [pid 8589:tid 8677] (13)Permission denied: [client 127.0.0
.1:41352] AH00035: access to /test.html denied (filesystem path '/var/www/html/test.html') because sear
ch permissions are missing on a component of the path
root@fedora:~# tail /var/log/audit/audit.log
type=BPF msg=audit(1728642363.038:765): prog-id=163 op=UNLOAD
type=SERVICE_STOP msg=audit(1728642368.849:766): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u
:system_r:init_t:s0 msg='unit=systemd-hostnamed comm="systemd" exe="/usr/lib/systemd/systemd" hostname=?
addr=? terminal=? res=success'UID="root" AUID="unset"
type=BPF msg=audit(1728642368.872:767): prog-id=166 op=UNLOAD
type=BPF msg=audit(1728642368.872:768): prog-id=165 op=UNLOAD
type=BPF msg=audit(1728642368.872:769): prog-id=164 op=UNLOAD
type=USER_END msg=audit(1728642374.364:770): pid=15498 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconfi
ned_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_close grantors=pam_keyinit,pam_limits,pam_keyinit,p
am_limits,pam_systemd,pam_unix acct="root" exe="/usr/bin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=
success'UID="jordaniakondzo" AUID="jordaniakondzo"
type=CRED_DISP msg=audit(1728642374.377:771): pid=15498 uid=1000 auid=1000 ses=3 subj=unconfined_u:unconf
ined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:setcred grantors=pam_unix acct="root" exe="/usr/bin/sudo"
hostname=? addr=? terminal=/dev/pts/0 res=success'UID="jordaniakondzo" AUID="jordaniakondzo"
type=SERVICE_STOP msg=audit(1728642528.033:772): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u
:system_r:init_t:s0 msg='unit=packagekit comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=?
terminal=? res=success'UID="root" AUID="unset"
type=AVC msg=audit(1728642545.195:773): avc: denied { getattr } for pid=8589 comm="httpd" path="/var/w
ww/html/test.html" dev="sda3" ino=662265 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:obje

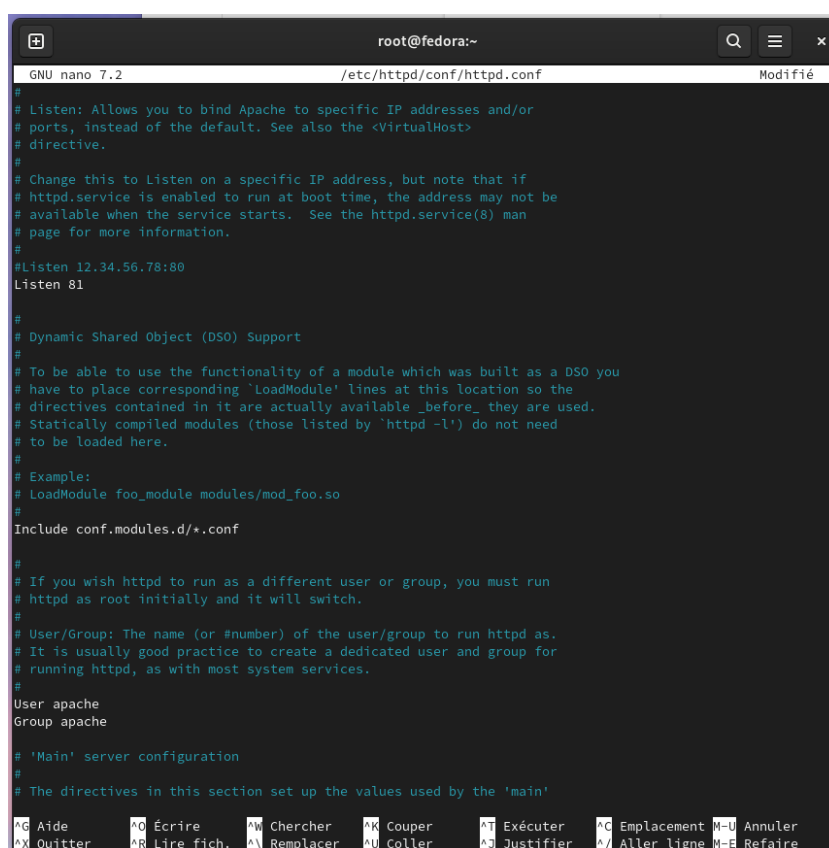
```

Рис. 3.17: Анализ ситуаций

- Анализ логов не выявил ошибок или предупреждений, связанных с перезапуском сервера.

16. Замена порта 80 на 81 для Apache в SELinux

- Попробовал 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 нашёл строчку Listen 80 и заменил её на Listen 81. (рис. 3.18)



```

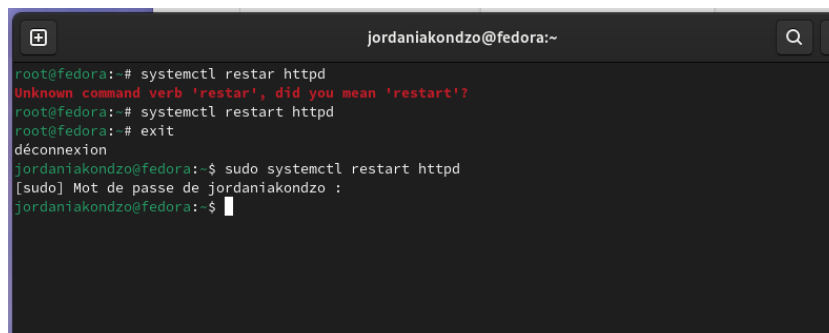
root@fedora:~
GNU nano 7.2 /etc/httpd/conf/httpd.conf
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at 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
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf
#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
# It is usually good practice to create a dedicated user and group for
# running httpd, as with most system services.
#
User apache
Group apache
#
# 'Main' server configuration
#
# The directives in this section set up the values used by the 'main'
^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler
^X Quitter   ^R Lire fich. ^\ Remplacer ^U Coller    ^D Justifier ^/_ Aller ligne M-E Refaire

```

Рис. 3.18: Замена порта 80 на 81 для Apache в SELinux

17. Перезапуск веб-сервера Apache

- Я перезапустил сервер Apache с помощью команды: (рис. 3.19)



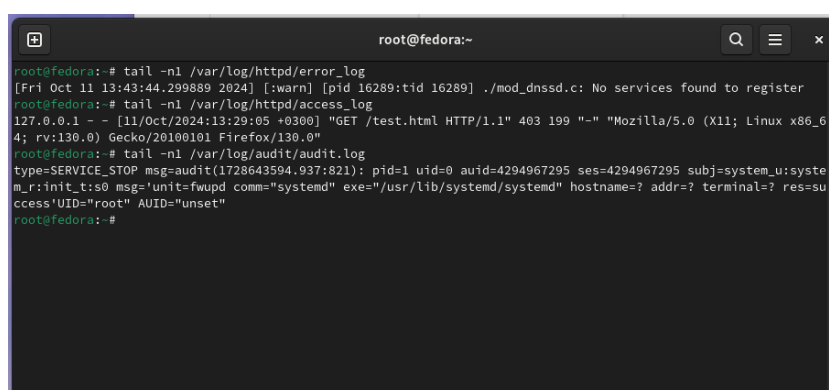
```
jordaniakondzo@fedora:~  
root@fedora:~# systemctl restar httpd  
Unknown command verb 'restar', did you mean 'restart'?  
root@fedora:~# systemctl restart httpd  
root@fedora:~# exit  
déconnexion  
jordaniakondzo@fedora:~$ sudo systemctl restart httpd  
[sudo] Mot de passe de jordaniakondzo :  
jordaniakondzo@fedora:~$
```

Рис. 3.19: Перезапуск веб-сервера Apache

- При перезапуске не возникло никаких сбоев, что подтвердило корректность конфигурации.

18. Анализ лог-файлов

- Я проанализировал лог-файлы с помощью следующих команд: (рис. 3.20)

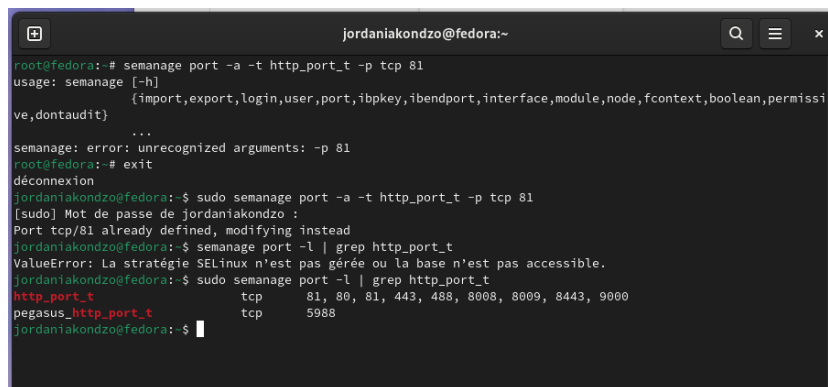


```
root@fedora:~# tail -n1 /var/log/httpd/error_log  
[Fri Oct 11 13:43:44.299889 2024] [:warn] [pid 16289:tid 16289] ./mod_dnssd.c: No services found to register  
root@fedora:~# tail -n1 /var/log/httpd/access_log  
127.0.0.1 ~ [11/Oct/2024:13:29:05 +0300] "GET /test.html HTTP/1.1" 403 199 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:130.0) Gecko/20100101 Firefox/130.0"  
root@fedora:~# tail -n1 /var/log/audit/audit.log  
type=SERVICE_STOP msg=audit(1728643594.937:821): pid=1 uid=0 auid=4294967295 ses=4294967295 subj=system_u:system_r:init_t:s0 msg='unit=fwupd comm="systemd" exe="/usr/lib/systemd/systemd" hostname=? addr=? terminal=? res=success'UID="root" AUID="unset"  
root@fedora:~#
```

Рис. 3.20: Анализ лог-файлов

19. Добавление порта 81 для Apache в SELinux

- Для добавления порта 81 я использовал команду: (рис. 3.21)

A terminal window titled 'jordaniakondzo@fedora:~' showing the process of adding port 81 to the http_port_t SELinux port set. The user runs 'semanage port -a -t http_port_t -p tcp 81'. The command fails with an error: 'semanage: error: unrecognized arguments: -p 81'. The user then runs 'sudo semanage port -a -t http_port_t -p tcp 81', which succeeds, showing 'Port tcp/81 already defined, modifying instead'. Finally, the user runs 'semanage port -l | grep http_port_t', which outputs a list of ports: 'http_port_t tcp 81, 80, 81, 443, 488, 8008, 8009, 8443, 9000' and 'pegasus_http_port_t tcp 5988'.

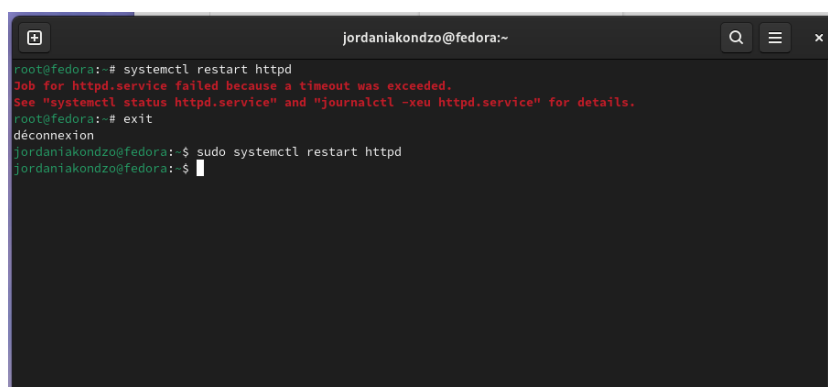
```
jordaniakondzo@fedora:~  
root@fedora:~# semanage port -a -t http_port_t -p tcp 81  
usage: semanage [-h]  
           {import,export,login,user,port,ibpkey,ibendport,interface,module,node,fcontext,boolean,permission,dontaudit}  
           ...  
semanage: error: unrecognized arguments: -p 81  
root@fedora:~# exit  
déconnexion  
jordaniakondzo@fedora:~$ sudo semanage port -a -t http_port_t -p tcp 81  
[sudo] Mot de passe de jordaniakondzo :  
Port tcp/81 already defined, modifying instead  
jordaniakondzo@fedora:~$ semanage port -l | grep http_port_t  
ValueError: La stratégie SELinux n'est pas gérée ou la base n'est pas accessible.  
jordaniakondzo@fedora:~$ sudo semanage port -l | grep http_port_t  
http_port_t      tcp      81, 80, 81, 443, 488, 8008, 8009, 8443, 9000  
pegasus_http_port_t  tcp      5988  
jordaniakondzo@fedora:~$
```

Рис. 3.21: Добавление порта 81 для Apache в SELinux

- Это позволило серверу Apache работать на порту 81.

20. Проверка списка портов

- Я проверил, что порт 81 был успешно добавлен с помощью команды: (рис. 3.22)

A terminal window titled 'jordaniakondzo@fedora:~' showing the user attempting to restart the httpd service. The first command 'systemctl restart httpd' fails with the message 'Job for httpd.service failed because a timeout was exceeded. See "systemctl status httpd.service" and "journalctl -xeu httpd.service" for details.'. The user then runs 'sudo systemctl restart httpd', which succeeds without any output.

```
jordaniakondzo@fedora:~  
root@fedora:~# systemctl restart httpd  
Job for httpd.service failed because a timeout was exceeded.  
See "systemctl status httpd.service" and "journalctl -xeu httpd.service" for details.  
root@fedora:~# exit  
déconnexion  
jordaniakondzo@fedora:~$ sudo systemctl restart httpd  
jordaniakondzo@fedora:~$
```

Рис. 3.22: Проверка списка портов

- В выводе я увидел, что порты 80 и 81 доступны для Apache.

21. Перезапуск Apache

- Я перезапустил веб-сервер Apache, чтобы изменения вступили в силу: (рис. 3.23)

- Файл **test.html** успешно отобразился, что подтвердило работу Apache на новом порту 81.

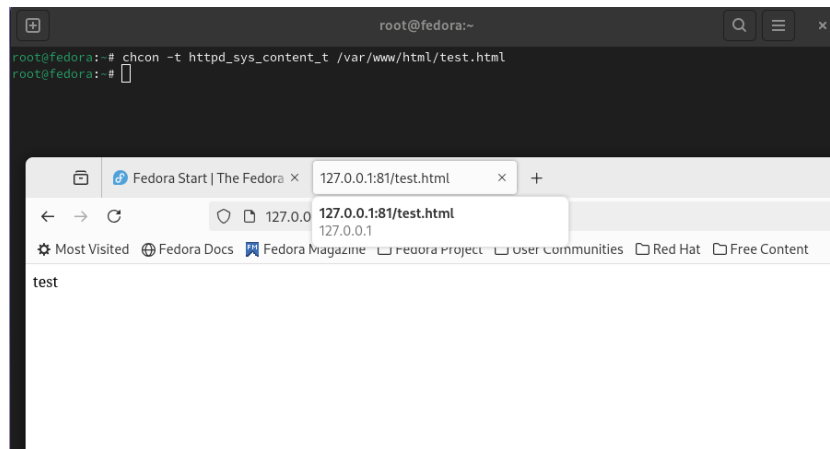
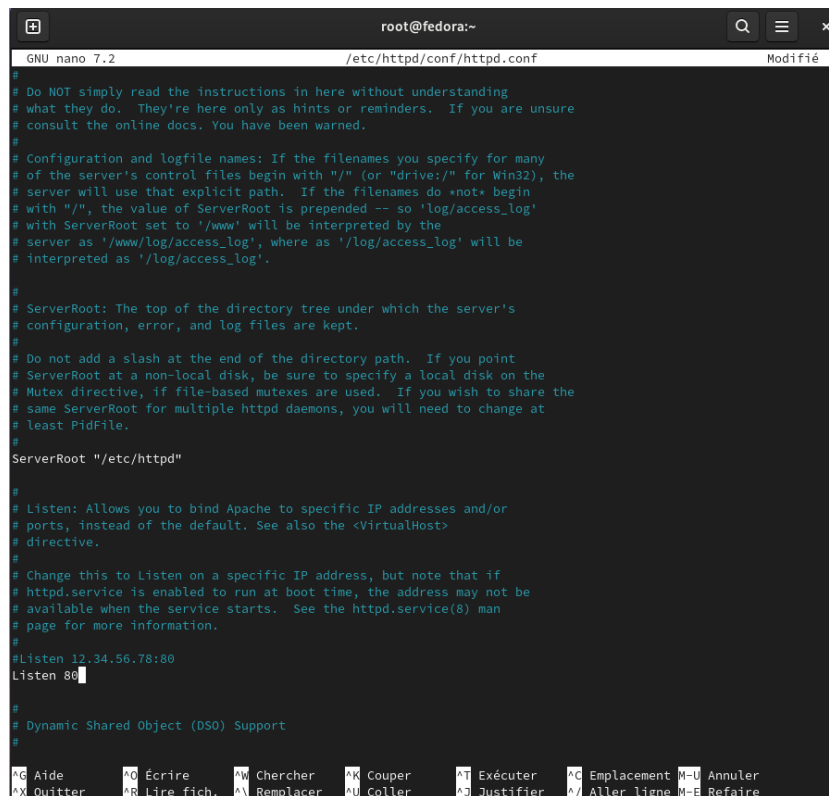


Рис. 3.23: Перезапуск Apache

22. Вернул контекст `httpd_sys_content_t` к файлу `/var/www/html/ test.html`:
(рис. 3.24)

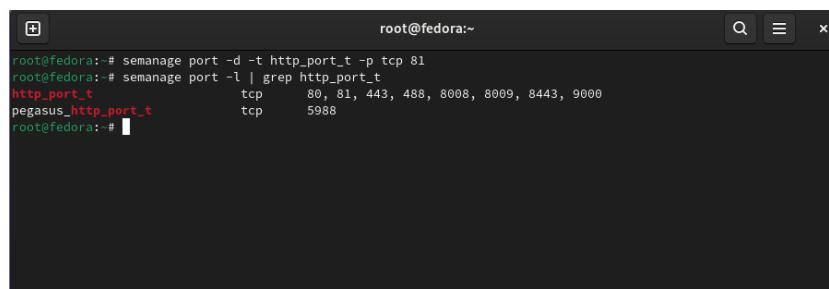


```
GNU nano 7.2 /etc/httpd/conf/httpd.conf
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders.  If you are unsure
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify 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 80
#
# Dynamic Shared Object (DSO) Support
#
#G Aide      ^O Écrire   ^W Chercher  ^K Couper    ^T Exécuter  ^C Emplacement M-U Annuler
^X Quitter   ^R Lire fich.^V Remplacer ^U Coller    ^J Justifier ^/_ Aller ligne M-E Refaire
```

Рис. 3.24: Вернул контекст httpd_sys_content_t

23. Удаление привязки порта 81 к Apache

- После успешного тестирования я удалил порт 81 из списка разрешенных для SELinux с помощью команды: (рис. 3.25)



```
root@fedora:~# semanage port -d -t http_port_t -p tcp 81
root@fedora:~# 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@fedora:~#
```

Рис. 3.25: Удаление привязки порта 81 к Apache

24. Удаление тестового файла

- Я удалил тестовый файл test.html с помощью команды: (рис. 3.26)

A terminal window titled 'root@fedora:~' with search, menu, and close icons. It shows a sequence of commands: 'rm /var/www/html/test.html', a confirmation prompt 'rm : supprimer '/var/www/html/test.html' du type fichier ? y', 'ls /var/www/html', and 'ls -l /var/www/html' which outputs 'total 0'.

```
root@fedora:~# rm /var/www/html/test.html
rm : supprimer '/var/www/html/test.html' du type fichier ? y
root@fedora:~# ls /var/www/html
root@fedora:~# ls -l /var/www/html
total 0
root@fedora:~#
```

Рис. 3.26: Удаление тестового файла

4 Выводы

В ходе лабораторной работы я приобрел практические навыки работы с SELinux в связке с веб-сервером Apache. Я научился настраивать контексты безопасности для файлов и управлять портами, используя SELinux для обеспечения мандатного контроля доступа. Работая с различными контекстами, такими как `httpd_sys_content_t` и `samba_share_t`, я увидел, как SELinux блокирует несанкционированный доступ, что помогает значительно повысить безопасность системы.