

Презентация лабораторной работы

6

ТЕМА «Мандатное разграничение прав в Linux»

Выполнил:

Студент группы НПИбд-02-21

Студенческий билет № 1032205421

Стелина Петрити

Цель работы

Развить навыки управления операционной системой Linux, получив практический опыт работы с технологией SELinux. Исследовать работу SELinux на практике в сочетании с веб-сервером Apache.

Последовательность выполнения работы

6.3. Подготовка лабораторного стенда и методические рекомендации

1. При подготовке стенда обратите внимание, что необходимая для работы и указанная выше политика targeted и режим enforcing используются в данном дистрибутиве по умолчанию, т.е. каких-то специальных настроек не требуется. При этом следует убедиться, что политика и режим включены, особенно когда работа будет проводиться повторно и велика вероятность изменений при предыдущем использовании системы

```
[stelinapetrity@localhost ~]$ su -
Password:
[root@localhost ~]# getenforce
Enforcing
[root@localhost ~]# 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
[root@localhost ~]#
```

рис. 1 SELinux и Enforcing

2. При необходимости администратор должен разбираться в работе SELinux и уметь как исправить конфигурационный файл /etc/selinux/config, так и проверить используемый режим и политику.
3. Необходимо, чтобы был установлен веб-сервер Apache. При установке системы в конфигурации «рабочая станция» указанный пакет не ставится.

```
[root@localhost ~]# yum install httpd
Rocky Linux 9 - BaseOS                               3.9 kB/s |
Rocky Linux 9 - BaseOS                               946 kB/s |
Rocky Linux 9 - AppStream                             5.2 kB/s |
Rocky Linux 9 - AppStream                             4.5 MB/s |
Rocky Linux 9 - Extras                                3.6 kB/s |
Rocky Linux 9 - Extras                                13 kB/s |
Dependencies resolved.
=====
Package                               Architecture      Version           Repository
=====
Installing:
httpd                                x86_64            2.4.57-11.el9_4.1 appstream
```

рис. 2 Установка httpd

4. В конфигурационном файле /etc/httpd/httpd.conf необходимо задать параметр ServerName: ServerName test.ru чтобы при запуске веб-сервера не выдавались лишние сообщения об ошибках, не относящихся к лабораторной работе.

```
#
ServerAdmin root@localhost

#
# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
ServerName test.ru
```

рис. 3 Добавьте строку ServerName test.ru

5. Также необходимо проследить, чтобы пакетный фильтр был отключён или в своей рабочей конфигурации позволял подключаться к 80-у и 81-у портам протокола tcp. Отключить фильтр можно командами `iptables -F`, `iptables -P INPUT ACCEPT` `iptables -P OUTPUT ACCEPT` либо добавить разрешающие правила: `iptables -I INPUT -p tcp --dport 80 -j ACCEPT`, `iptables -I INPUT -p tcp --dport 81 -j ACCEPT`, `iptables -I OUTPUT -p tcp --sport 80 -j ACCEPT`, `iptables -I OUTPUT -p tcp --sport 81 -j ACCEPT`

```
[root@localhost ~]# iptables -F
[root@localhost ~]# iptables -P INPUT ACCEPT
iptables v1.8.10 (nf_tables): unknown protocol "input" specified
Try `iptables -h' or 'iptables --help' for more information.
[root@localhost ~]# iptables -F
[root@localhost ~]# iptables -P INPUT ACCEPT
[root@localhost ~]# iptables -P OUTPUT ACCEPT
[root@localhost ~]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
iptables v1.8.10 (nf_tables): unknown option "--dport"
Try `iptables -h' or 'iptables --help' for more information.
[root@localhost ~]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
[root@localhost ~]# iptables -I INPUT -p tcp --dport 81 -j ACCEPT
[root@localhost ~]# iptables -I INPUT -p tcp --sport 80 -j ACCEPT
[root@localhost ~]# iptables -I INPUT -p tcp --sport 81 -j ACCEPT
[root@localhost ~]#
```

рис. 4 Настройка пакета iptables

6. Обратите внимание, что данные правила не являются «точными» и рекомендуемыми на все случаи жизни, они лишь позволяют правильно организовать работу стенда.
7. В работе специально не делается акцент, каким браузером (или какой консольной программой) будет производиться подключение к веб-серверу. По желанию могут использоваться разные программы, такие как консольные `links`, `lynx`, `wget` и графические `konqueror`, `opera`, `firefox` или др.

```
[root@localhost ~]# yum install lynx
Last metadata expiration check: 0:06:47 ago on Wed 09 Oct 2024 02:07:32 PM CEST.
Dependencies resolved.
=====
Package                        Architecture      Version           Repositor
=====
Installing:
lynx                           x86_64            2.8.9-20.el9     appstream
Transaction Summary
=====
Install 1 Package

Total download size: 1.5 M
Installed size: 6.1 M
Is this ok [y/N]: y
Downloading Packages:
lynx-2.8.9-20.el9.x86_64.rpm
Total
```

рис. 5 Выбор браузера lynx

6.4. Порядок выполнения работы

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

```
[root@localhost ~]# getenforce
Enforcing
[root@localhost ~]# 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
[root@localhost ~]#
```

рис. 6 Вход в систему и проверка

2. Обратитесь с помощью браузера к веб-серверу, запущенному на вашем компьютере, и убедитесь, что последний работает: `service httpd status` или `/etc/rc.d/init.d/httpd status`. Если не работает, запустите его так же, но с параметром `start`.

```
[root@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; disabled; preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
[root@localhost ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@localhost ~]#
```

рис. 7 статус веб-сервера, если сервер не запущен, запустите с `start`

3. Найдите веб-сервер Apache в списке процессов, определите его контекст безопасности и занесите эту информацию в отчёт. Например, можно использовать команду `ps auxZ | grep httpd` или `ps -eZ | grep httpd`

```
[root@localhost ~]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root      3625  0.1  0.6 20152 11396 ?        Ss   14:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  3626  0.0  0.3 22032  7228 ?        S    14:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  3627  0.0  0.9 1112588 17452 ?        Sl   14:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  3628  0.0  0.8 981452 15136 ?        Sl   14:17   0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  3629  0.0  0.8 981452 15136 ?        Sl   14:17   0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 3806  0.0  0.1 221664 2176 pts/0 S+   14:18   0:00 grep --color=auto httpd
[root@localhost ~]# ps -eZ | grep httpd
system_u:system_r:httpd_t:s0 3625 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 3626 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 3627 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 3628 ? 00:00:00 httpd
system_u:system_r:httpd_t:s0 3629 ? 00:00:00 httpd
[root@localhost ~]#
```

рис. 8 Определение контекста безопасности Apache

4. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd`

```

[root@localhost ~]# 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
cluster_manage_all_files        off
cluster_use_execmem             off
cobbler_anon_write              off
cobbler_can_network_connect     off
cobbler_use_cifs                off
cobbler_use_nfs                 off
collectd_tcp_network_connect    off
colord_use_nfs                  off

```

рис. 9 Просмотр состояние переключателей SELinux для Apache

Обратите внимание, что многие из них находятся в положении «off».

5. Посмотрите статистику по политике с помощью команды `seinfo`, также определите множество пользователей, ролей, типов.

```
[root@localhost ~]# 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:            65500    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 Constrain:    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

[root@localhost ~]#
```

рис. 10 статистики по политике SELinux

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

```
[root@localhost ~]# 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 18:30 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 Aug  8 18:30 html
[root@localhost ~]#
```

рис. 11 Определение типов файлов и директорий

7. Определите тип файлов, находящихся в директории /var/www/html: `ls -lZ /var/www/html`

```
[root@localhost ~]# ls -lZ /var/www/html
total 0
[root@localhost ~]#
```

рис. 12 Определение типов файлов в /var/www/html

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

```
[root@localhost ~]# ls -lZ /var/www/html
total 0
[root@localhost ~]#
```

рис. 13 Проверка прав создания файлов

9. Создайте от имени суперпользователя (так как в дистрибутиве после установки только ему разрешена запись в директорию) html-файл `/var/www/html/test.html` следующего содержания: `test`

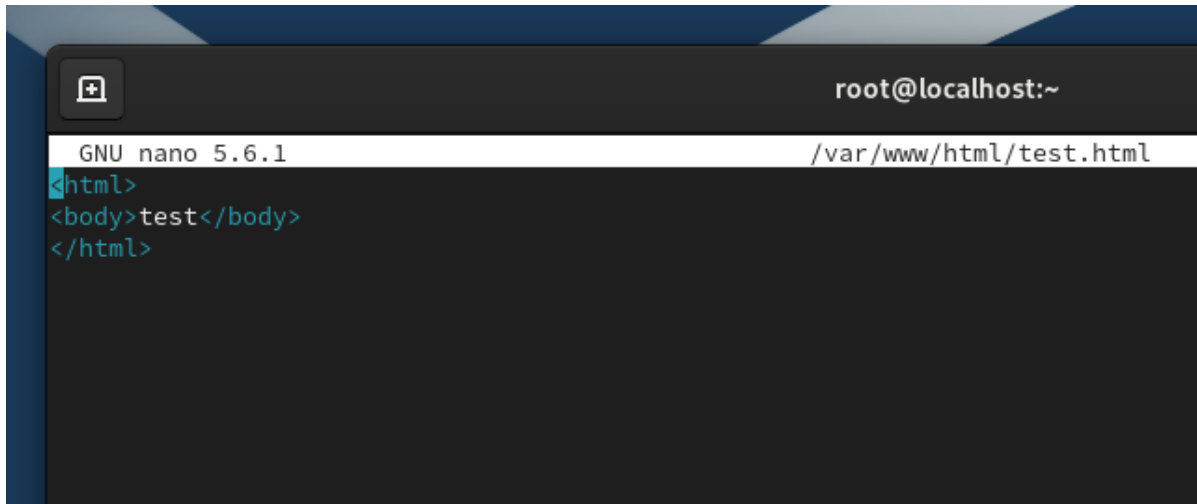


рис. 14 Создание HTML-файла

10. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории `/var/www/html`.

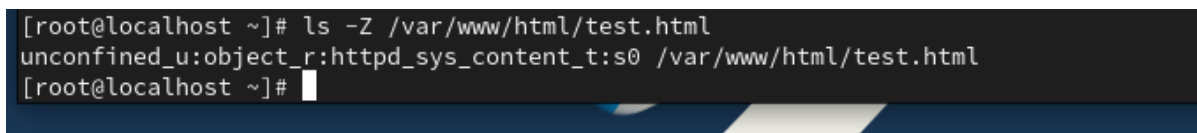


рис. 15 контекст созданного файла

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

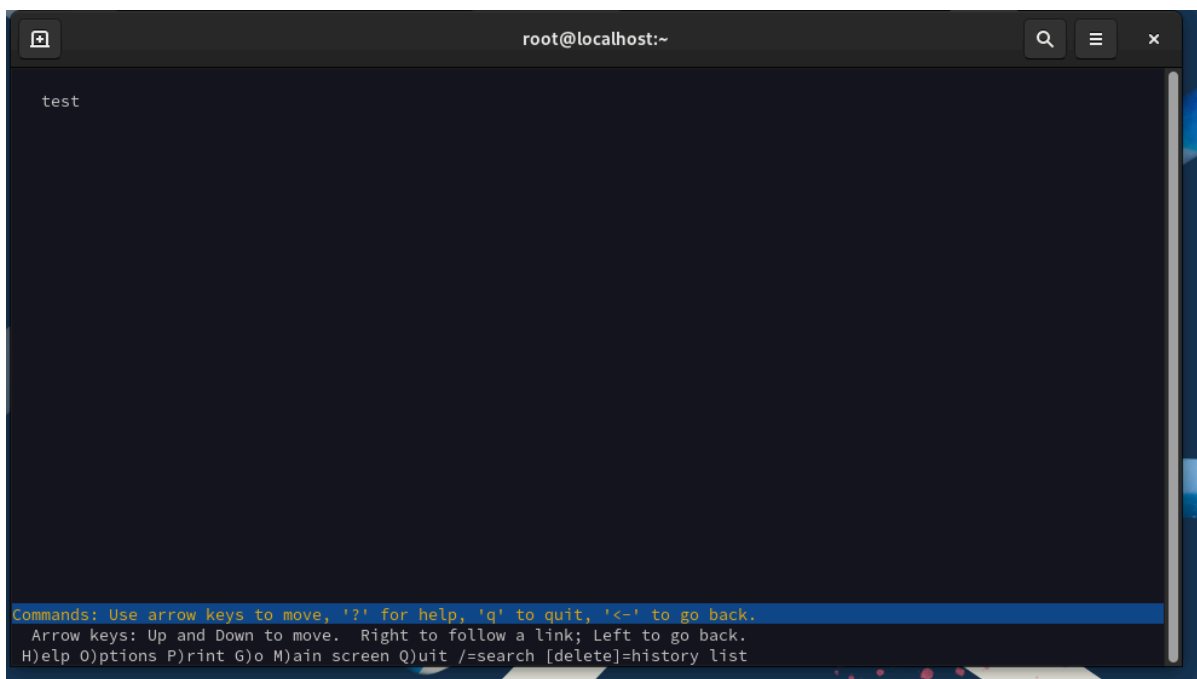


рис. 16 файл отображается корректно

12. Изучите справку `man httpd_selinux` и выясните, какие контексты файлов определены для `httpd`. Сопоставьте их с типом файла `test.html`. Проверить контекст файла можно командой `ls -Z. ls -Z /var/www/html/test.html`

Рассмотрим полученный контекст детально. Обратите внимание, что 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 /proc 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.

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: `chcon -t samba_share_t /var/www/html/test.html` `ls -Z /var/www/html/test.html`. После этого проверьте, что контекст поменялся.

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

рис. 17 Изменение контекста файла

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: Forbidden You don't have permission to access /test.html on this server.

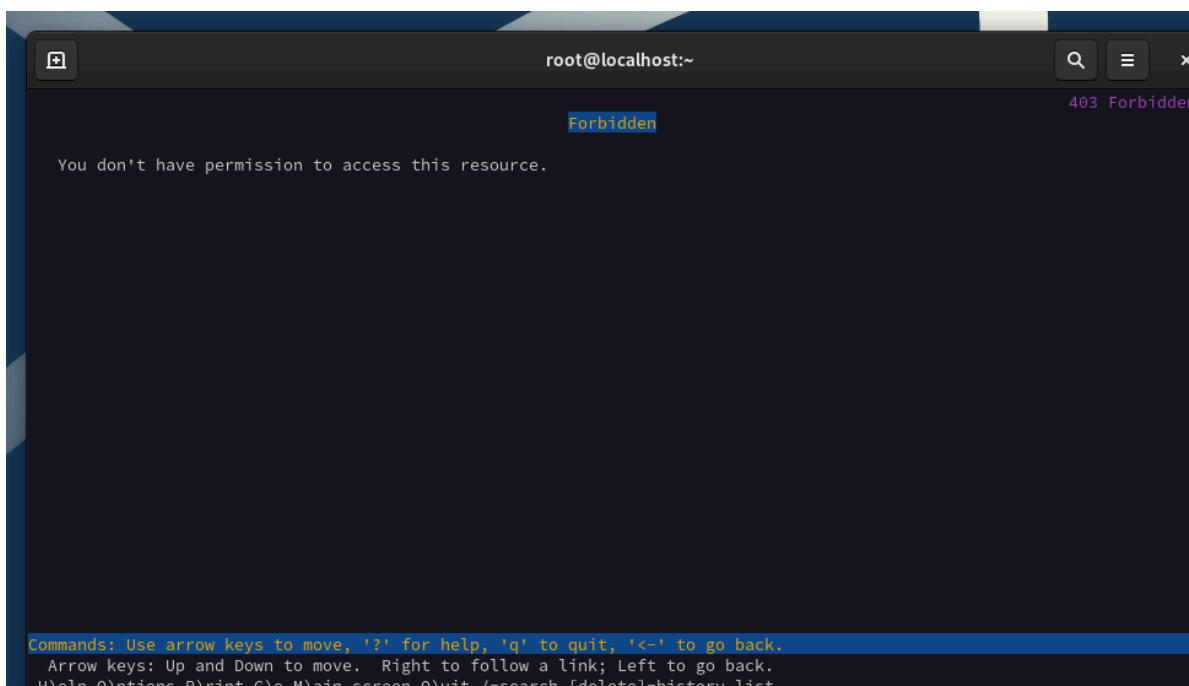


рис. 18 Проверка доступа к файлу

15. Проанализируйте ситуацию. Почему файл не был отображён, если права доступа позволяют читать этот файл любому пользователю? `ls -l /var/www/html/test.html` Просмотрите log-файлы веб-сервера Apache. Также просмотрите системный лог-файл: `tail /var/log/messages` Если в системе окажутся запущенными процессы `setroubleshootd` и `auditd`, то вы также

сможете увидеть ошибки, аналогичные указанным выше, в файле /var/log/audit/audit.log.
Проверьте это утверждение самостоятельно.

```
[root@localhost ~]# lynx http://127.0.0.1/test.html
[root@localhost ~]# ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 Oct  9 14:28 /var/www/html/test.html
[root@localhost ~]# tail /var/log/messages
Oct  9 14:32:50 localhost systemd[1]: Created slice Slice /system/dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged.
Oct  9 14:32:50 localhost systemd[1]: Started dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@0.service.
Oct  9 14:32:53 localhost setroubleshoot[4049]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /
var/www/html/test.html. For complete SELinux messages run: sealert -l 58942c13-61d1-4fa5-9ea3-8331b642c124
Oct  9 14:32:53 localhost setroubleshoot[4049]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /
var/www/html/test.html.#012#012***** Plugin restorecon (92.2 confidence) suggests *****#012#012If
you want to fix the label. #012/var/www/html/test.html default label should be httpd_sys_content_t.#012Then you can ru
n restorecon. The access attempt may have been stopped due to insufficient permissions to access a parent directory in
which case try to change the following command accordingly.#012Do#012# /sbin/restorecon -v /var/www/html/test.html#012#
012***** Plugin public_content (7.83 confidence) suggests *****#012#012If you want to treat test.html
as public content#012Then you need to change the label on test.html to public_content_t or public_content_rw_t.#012Do#
012# semanage fcontext -a -t public_content_t '/var/www/html/test.html'#012# restorecon -v '/var/www/html/test.html'#01
2#012***** Plugin catchall (1.41 confidence) suggests *****#012#012If you believe that httpd sh
ould be allowed getattr access on the test.html file by default.#012Then you should report this as a bug.#012You can ge
nerate a local policy module to allow this access.#012Do#012allow this access for now by executing:#012# ausearch -c 'h
ttpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
Oct  9 14:32:53 localhost setroubleshoot[4049]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /
var/www/html/test.html. For complete SELinux messages run: sealert -l 58942c13-61d1-4fa5-9ea3-8331b642c124
Oct  9 14:32:53 localhost setroubleshoot[4049]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /
```

рис. 19 Анализ логов

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.

```
# 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
```

рис. 20 Запуск Apache на порту 81

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

```
[root@localhost ~]# systemctl restart httpd
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
   Active: active (running) since Wed 2024-10-09 14:37:49 CEST; 11s ago
     Docs: man:httpd.service(8)
  Main PID: 4073 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 10966)
   Memory: 25.5M
      CPU: 120ms
   CGroup: /system.slice/httpd.service
           └─4073 /usr/sbin/httpd -DFOREGROUND
             └─4075 /usr/sbin/httpd -DFOREGROUND
               └─4076 /usr/sbin/httpd -DFOREGROUND
                 └─4077 /usr/sbin/httpd -DFOREGROUND
                   └─4078 /usr/sbin/httpd -DFOREGROUND

Oct 09 14:37:49 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Oct 09 14:37:49 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Oct 09 14:37:49 localhost.localdomain httpd[4073]: Server configured, listening on: port 81
[root@localhost ~]#
```

рис. 21 Перезапуск веб-сервера

18. Проанализируйте лог-файлы: tail -nl /var/log/messages, /var/log/http/access_log

```
[root@localhost ~]# tail -n1 /var/log/messages
Oct 9 14:37:49 localhost httpd[4073]: Server configured, listening on: port 81
[root@localhost ~]# tail -n1 /var/log/httpd/error_log
tail: invalid number of lines: '1'
[root@localhost ~]# tail -n1 /var/log/httpd/error_log
[Wed Oct 09 14:37:49.495842 2024] [core:notice] [pid 4073:tid 4073] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
[root@localhost ~]# sudo tail -n1 /var/log/httpd/access_log
127.0.0.1 - - [09/Oct/2024:14:32:48 +0200] "GET /test.html HTTP/1.0" 403 199 "-" "Lynx/2.8.9rel.1 libwww-FM/2.14 SSL/1.4.1 OpenSSL/3.0.7"
[root@localhost ~]# sudo tail -n1 /var/log/httpd/audit.log
tail: cannot open '/var/log/httpd/audit.log' for reading: No such file or directory
[root@localhost ~]# sudo tail -n1 /var/log/audit/audit.log
type=USER_START msg=audit(1728477641.027:228): pid=4267 uid=0 auid=1000 ses=3 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 msg='op=PAM:session_open grantors=pam_keyinit,pam_limits,pam_systemd,pam_unix acct="root" exe="/usr/sbin/sudo" hostname=? addr=? terminal=/dev/pts/0 res=success'UID="root" AUID="stelinapetrityi"
[root@localhost ~]#
```

рис. 22 Анализ логов после изменения порта

19. Выполните команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверьте список портов командой `semanage port -l | grep http_port_t`. Убедитесь, что порт 81 появился в списке.

```
[root@localhost ~]# semanage port -a -t http_port_t -p tcp 81
Port tcp/81 already defined, modifying instead
[root@localhost ~]# 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
[root@localhost ~]#
```

рис. 23 Добавление нового порта в SELinux

20. Попробуйте запустить веб-сервер Apache ещё раз. <http://127.0.0.1:81/test.html>
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».

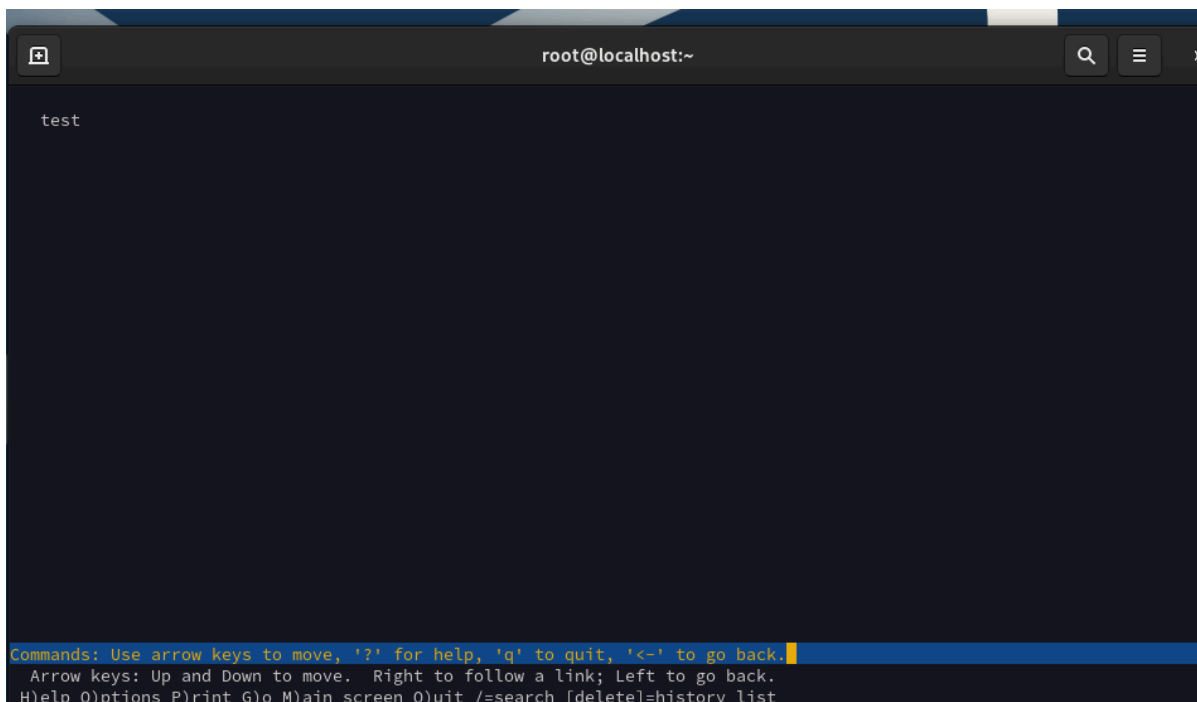


рис. 24 Восстановление контекста файла

```
[root@localhost ~]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@localhost ~]# lynx http://127.0.0.1:81/test.html
[root@localhost ~]#
```

рис. 25 Перезапуск веб-сервера Apache

22. Исправьте обратно конфигурационный файл apache, вернув Listen 80.

```
# 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 Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify   ^G Go To Line M-E Redo
```

рис. 26 Возвращение Apache на порт 80

23. Удалите привязку http_port_t к 81 порту: semanage port -d -t http_port_t -p tcp 81 и проверьте, что порт 81 удалён.

```
[root@localhost ~]# sudo semanage port -d -t http_port_t -p tcp 81
[root@localhost ~]#
```

рис. 27 Удаление порта 81 из списка SELinux

24. Удалите файл /var/www/html/test.html: rm /var/www/html/test.html

```
[root@localhost ~]# rm /var/www/html/test.html
rm: remove regular file '/var/www/html/test.html'? y
[root@localhost ~]#
```

рис. 28 Удаление файла

Вывод

В данной лабораторной работе мы изучили настройку и механизмы безопасности веб-сервера Apache в среде SELinux. Мы проверили активность службы Apache и убедились, что она корректно настроена для прослушивания на определённых портах, таких как 80 и 81.

Анализируя контексты SELinux и права доступа пользователей, мы поняли, как SELinux управляет доступом к файлам и каталогам в корневом каталоге веб-сервера. Возникшие проблемы, связанные с политиками SELinux, показали важность глубокого понимания взаимодействия этой системы безопасности с сетевыми службами, что ещё раз подчеркнуло необходимость детального изучения SELinux в реальных условиях.