Лабораторная работа №6

Дисциплина: Информационная безопасность

Манаева Варвара Евгеньевна

Содержание

Список иллюстраций

Список таблиц

1 Техническое оснащение:

- Персональный компьютер с операционной системой Windows 10;
- Планшет для записи видеосопровождения и голосовых комментариев;
- Microsoft Teams, использующийся для записи скринкаста лабораторной работы;
- Приложение Pycharm для редактирования файлов формата *md*;
- *pandoc* для конвертации файлов отчётов и презентаций.

2 Цели и задачи работы

2.1 Цель

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

2.2 Задачи

1. Подготовить лабораторный стенд;
2. Запустить Apache в системе;
3. Создать небольшой веб-сервер;
4. Посмотреть различные варианты настроек сервера и изучить реакции на изменение этих настроек.

3 Теоретическое введение и подготовка лабораторного стенда

HTTP-сервер Apache — самый широко используемый веб-сервер в мире. Он имеет множество мощных функций, включая динамически загружаемые модули, надежную поддержку различных медиаформатов и интеграцию с другим популярным программным обеспечением.

Apache доступен в используемых по умолчанию репозиториях программного обеспечения CentOS, т. е. вы можете установить его с помощью менеджера пакетов `yum`.

Порядок настройки машины для использования `Apache`:

```
sudo yum update httpd
sudo yum install httpd
sudo yum install firewalld
sudo systemctl start firewalld
sudo firewall-cmd --permanent --add-service=ssh
sudo firewall-cmd --permanent --add-service=http
sudo firewall-cmd --permanent --add-service=https
sudo firewall-cmd --reload
sudo systemctl enable firewalld
```

В файле `/etc/httpd/httpd.conf` необходимо задать:

```
ServerName test.ru
```

Также необходимо проследить, чтобы пакетный фильтр был отключён. Отключить фильтр можно командами:

```
sudo iptables -F
sudo iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT
```

4 Выполнение лабораторной работы [1]

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команды sestatus. (4.1)

```
sestatus
```

```
[vemanaeva@vemanaeva ~]$ 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
Max kernel policy version:      31
```

Рис. 4.1: Проверим SELinux

2. Обратимся с помощью браузера к веб-серверу, запущенному на компьютере, и убедимся, что последний работает (4.2)

```
sudo systemctl start httpd
service httpd status
```

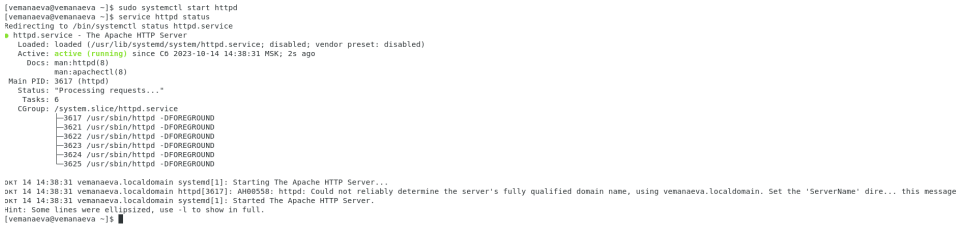


Рис. 4.2: Проверяем

3. Найдём веб-сервер Apache в списке процессов и определим его контекст безопасности (4.3)

```
ps auxZ | grep httpd
ps -eZ | grep httpd
```

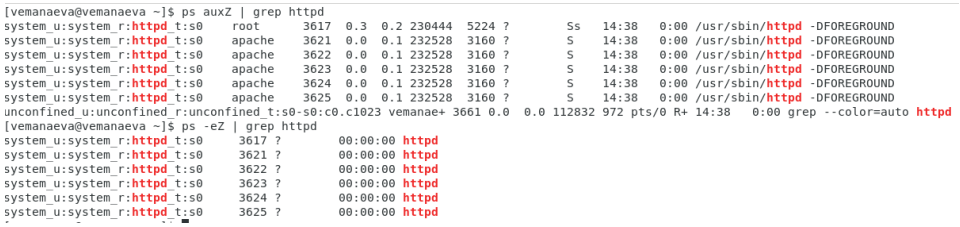


Рис. 4.3: Контекст безопасности

4. Посмотрим текущее состояние переключателей SELinux для Apache (4.4)

```
sestatus -bigrep httpd
```

```
[vemanaeva@vemanaeva ~]$ 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
Max kernel policy version:      31

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
condor_tcp_network_connect                  off
conman_can_network                          off
conman_use_nfs                              off
container_connect_any                       off
cron_can_relabel                            off
cron_system_cronjob_use_shares              off
cron_userdomain_transition                  on
cups_execmem                                off
cvs_read_shadow                             off
daemons_dump_core                           off
daemons_enable_cluster_mode                 off
daemons_use_tcp_wrapper                     off
daemons_use_tty                             off
dbadm_exec_content                          on
```

Рис. 4.4: Текущее состояние переключателей SELinux для `Apache`

5. Посмотрим статистику по политике, а также определим множество пользователей, ролей и типов (4.5)

`seinfo`

```
[vemanaeva@vemanaeva ~]$ seinfo

Statistics for policy file: /sys/fs/selinux/policy
Policy Version & Type: v.31 (binary, mls)

   Classes:            130    Permissions:       272
   Sensitivities:        1    Categories:       1024
   Types:             4793    Attributes:        253
   Users:                8    Roles:              14
   Booleans:           316    Cond. Expr.:       362
   Allow:           107834    Neverallow:          0
   Auditallow:         158    Dontaudit:       10022
   Type_trans:       18153    Type_change:        74
   Type_member:         35    Role allow:         37
   Role_trans:         414    Range_trans:      5899
   Constraints:        143    Validatetrans:       0
   Initial SIDs:        27    Fs_use:             32
   Genfscon:           103    Portcon:           614
   Netifcon:             0    Nodecon:             0
   Permissives:          0    Polcap:              5
```

Рис. 4.5: Статистика по политике

6. Определим тип файлов и поддиректорий, находящихся в директории `/var/www` (4.6)

```
ls -lZ /var/www
```

```
[vemanaeva@vemanaeva ~]$ ls -lZ /var/www
drwxr-xr-x. root root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin
drwxr-xr-x. root root system_u:object_r:httpd_sys_content_t:s0 html
[vemanaeva@vemanaeva ~]$
```

Рис. 4.6: Типы поддиректорий и файлов в директории `/var/www`

7. Определим тип файлов, находящихся в директории `/var/www/html` (4.7)

```
ls -lZ /var/www/html
```

```
[vemanaeva@vemanaeva ~]$ ls -lZ /var/www/html
[vemanaeva@vemanaeva ~]$
```

Рис. 4.7: Тип файлов в директории `/var/www/html`

8. Определим круг пользователей, которым разрешено создание файлов в директории `/var/www/html` (только суперпользователь) (4.8)

```
ls -l /var/www
```

```
   Genfscon:          103    Portcon:           614
   Netifcon:            0    Nodecon:             0
   Permissives:         0    Polcap:              5
Суперпользователь может
Другие пользователи нет
[vemanaeva@vemanaeva ~]$ ls -lZ /var/www
drwxr-xr-x. root root system_u:object_r:httpd_sys_script_exec_t:s0 cgi-bin
drwxr-xr-x. root root system_u:object_r:httpd_sys_content_t:s0 html
```

Рис. 4.8: Определим права на файл

9. Создадим от имени суперпользователя html-файл `/var/www/html/test.html` (4.9)

```
sudo nano /var/www/html/test.html
```

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

Рис. 4.9: Создадим файл

10. Проверим контекст созданного файла. (4.10)

```
ls -Z /var/www/html/test.html
```

Рис. 4.10: Определим контекст

11. Обратимся к файлу через веб-сервер, введя в браузере адрес “http://127.0.0.1/test.html”. Убедимся, что файл был успешно отображён. (4.11)

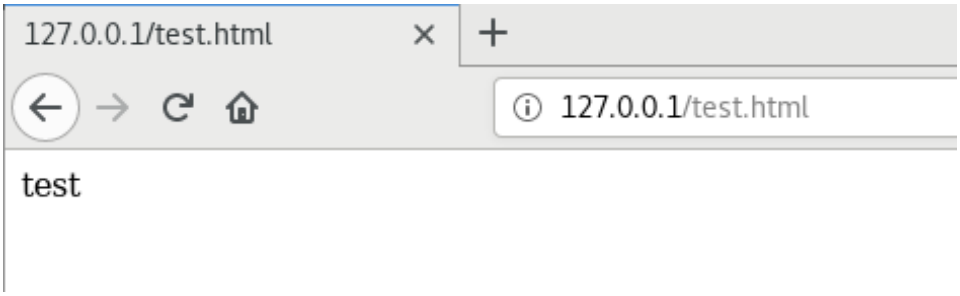


Рис. 4.11: http://127.0.0.1/test.html

12. Изучим справку и выясните, какие контексты файлов определены для httpd. Сопоставим их с типом файла test.html, проверив контекст файла (4.12)

```
man httpd_selinux
ls -Z /var/www/html/test.html
```

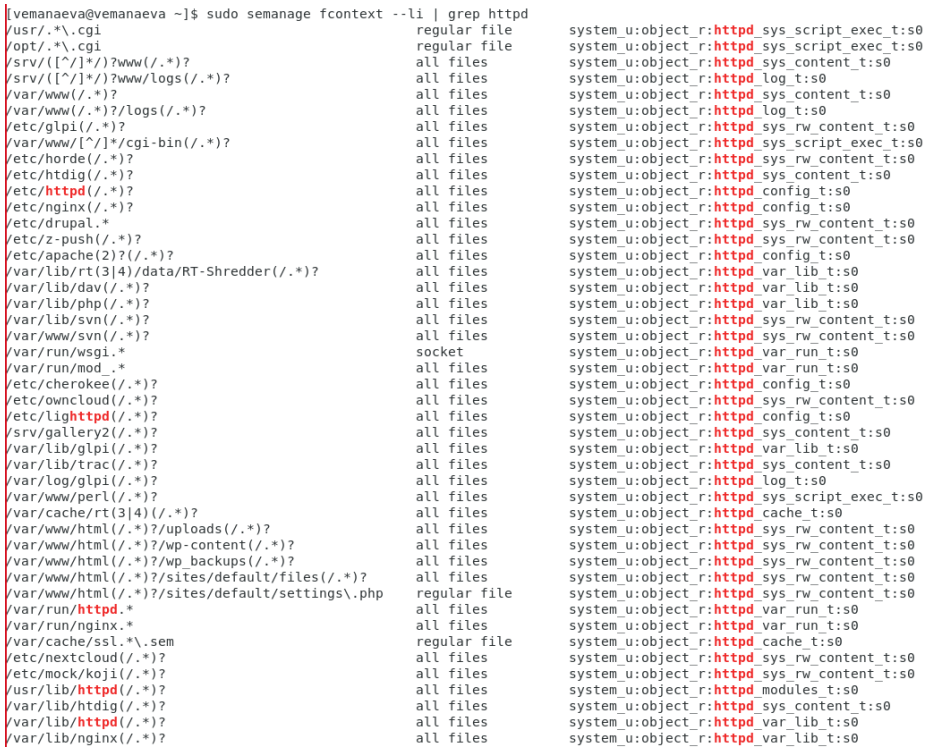


Рис. 4.12: Контексты файлов, определённые для SELinux

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 (4.13)

```
sudo chcon -t samba_share_t /var/www/html/test.html
ls -Z /var/www/html/test.html
```

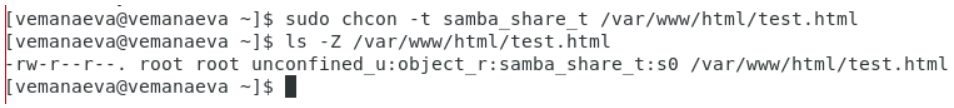


Рис. 4.13: Изменяем контекст файла

14. Попробуем ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес “http://127.0.0.1/test.html”. И получаем сообщение об ошибке Forbidden. (4.14)

Рис. 4.14: http://127.0.0.1/test.html

15. Проанализируем ситуацию. Просмотрим log-файлы веб-сервера Apache. (4.15)

```
ls -l /var/www/html/test.html
tail /var/log/messages
```

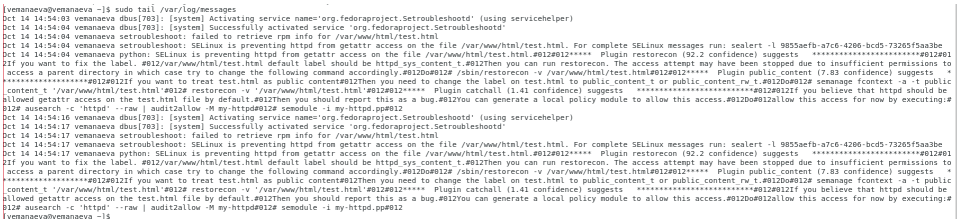


Рис. 4.15: log-файлы

16. Запустим 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 (4.16)

```
nano /etc/httpd/httpd.conf
```

Рис. 4.16: Замена строки

17. Выполним перезапуск веб-сервера Apache (4.17)

```
sudo systemctl start httpd
```

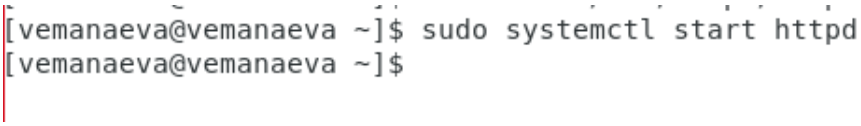
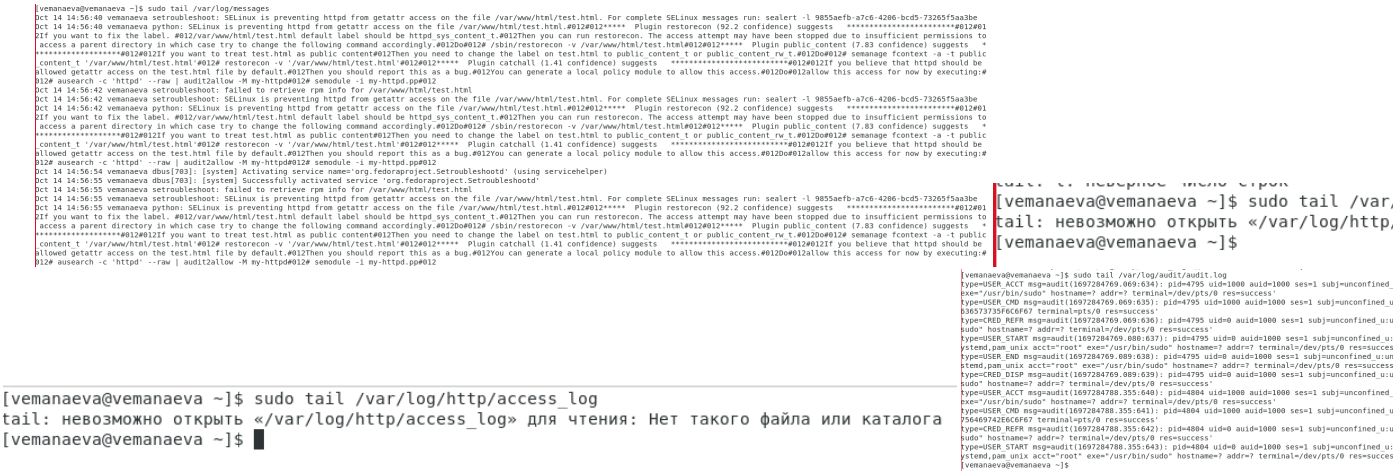


Рис. 4.17: Перезапуск веб-сервера

18. Проанализируем лог-файлы `/var/log/messages`. Также просмотрим файлы `/var/log/http/error_log`, `/var/log/http/access_log` и `/var/log/audit/audit.log` и выясним, в каких файлах появились записи (**??**)

```
tail /var/log/messages
tail /var/log/http/error_log
tail /var/log/http/access_log
tail /var/log/audit/audit.log
```



19. Выполним команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверим список портов. Убедимся, что порт 81 появился в списке. (4.18)

```
sudo semanage port -a -t http_port_t --proto tcp 81
semanage port -l | grep http_port_t
```

Рис. 4.18: Список портов

20. Пробуем запустить веб-сервер Apache ещё раз. И он работает. (4.19)

```
sudo systemctl start httpd
```

```
[vemanaeva@vemanaeva ~]$ sudo systemctl start httpd
[sudo] пароль для vemanaeva:
Попробуйте ещё раз.
[sudo] пароль для vemanaeva:
[vemanaeva@vemanaeva ~]$
```

Рис. 4.19: Веб-сервер работает

21. Вернём контекст `httpd_sys_content__t` к файлу `/var/www/html/test.html`. После этого попробуем получить доступ к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1:81/test.html. (4.20)

```
chcon -t httpd_sys_content_t /var/www/html/test.html
```

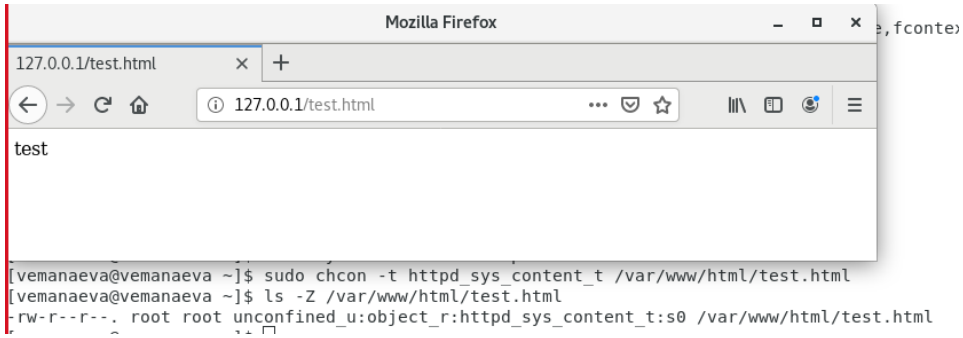


Рис. 4.20: http://127.0.0.1:81/test.html

22. Исправим обратно конфигурационный файл apache, вернув Listen 80. (4.21)

```
nano /etc/httpd/httpd.conf
```

Рис. 4.21: Исправляем конфигурационный файл

23. Удалим привязку http_port_t к 81 порту и проверим, что порт 81 удалён.(4.22)

```
semanage port -d -t http_port_t -p tcp 81
```

```
[vemanaeva@vemanaeva ~]$ sudo semanage port -d -t http_port_t --proto tcp 81
ValueError: Порт tcp/81 определен на уровне политики и не может быть удален
[vemanaeva@vemanaeva ~]$ sudo systemctl start httpd
[vemanaeva@vemanaeva ~]$ sudo semanage port -d -t http_port_t --proto tcp 81
ValueError: Порт tcp/81 определен на уровне политики и не может быть удален
[vemanaeva@vemanaeva ~]$
```

Рис. 4.22: Удаляем привязку

24. Удалим файл `/var/www/html/test.html` командой (4.23)

```
rm /var/www/html/test.html
```

```
[vemanaeva@vemanaeva ~]$ sudo rm /var/www/html/test.html
[vemanaeva@vemanaeva ~]$ ls -l /var/www/html
итого 0
[vemanaeva@vemanaeva ~]$
```

Рис. 4.23: Удаляем файл

5 Выводы по проделанной работе

5.1 Вывод

В результате выполнения работы мы развили навыки администрирования ОС Linux, получили первое практическое знакомство с технологией SELinux и проверили работу SELinx на практике совместно с веб-сервером `Apache`.

Были записаны скринкасты выполнения и защиты лабораторной работы.

Ссылки на скринкасты:

- Выполнение, Youtube
- Выполнение, Rutube
- Защита презентации, Youtube
- Защита презентации, Rutube

6 Список литературы

1. Лабораторная работа № 6 [Электронный ресурс]. Российский Университет Дружбы Народов имени Патрису Лумумбы, 2023. URL: https://esystem.rudn.ru/mod/resource/view.php?id=1031379.