

Отчет по лабораторной работе №6

по предмету Информационная безопасность

Алхимова Дарья Сергеевна

Содержание

Цель работы	4
Задание	5
Теоретическое введение	8
Выполнение лабораторной работы	9
Выводы	20
Список литературы	21

Список иллюстраций

1	Проверка работы SELinux	9
2	Проверка работы веб-сервера	10
3	Контекст Apache	10
4	Состояние переключателей SELinux для Apache	11
5	Статистика по политике	11
6	Типы файлов, поддиректорий и права для /www и /html	12
7	Создание файла test.html	12
8	Контекст файла test.html	12
9	Открытие test.html через браузер	13
10	Вызов справки httpd	13
11	Смена контекста файла test.html	13
12	Повторное открытие test.html через браузер	14
13	Просмотр системного лога	14
14	Открытие httpd.conf	15
15	httpd.conf до исправления	15
16	httpd.conf после исправления	16
17	Перезапуск Apache	16
18	Просмотр логов	16
19	Просмотр access_log	17
20	Просмотр error_log	17
21	Просмотр audit.log	17
22	Добавление и просмотр наличия 81 порта	18
23	Запуск Apache	18
24	Возвращение контекста файлу test.html	18
25	Открытие файла test.html через браузер	18
26	Возвращение настроек на 80 порт httpd.conf	19
27	Попытка удаления 81 порта	19
28	Удаление файла test.html	19

Цель работы

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

Задание

1. Войдите в систему с полученными учётными данными и убедитесь, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus`.
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`.
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`.
4. Посмотрите текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd`. 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.
6. Определите тип файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды `ls -lZ /var/www`.
7. Определите тип файлов, находящихся в директории `/var/www/html`: `ls -lZ /var/www/html`.
8. Определите круг пользователей, которым разрешено создание файлов в директории `/var/www/html`.
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:

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

10. Проверьте контекст созданного вами файла. Занесите в отчёт контекст, присваиваемый по умолчанию вновь созданным файлам в директории /var/www/html.
11. Обратитесь к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Убедитесь, что файл был успешно отображён.
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)`
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.
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.
```
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 `audtd`,

- то вы также сможете увидеть ошибки, аналогичные указанным выше, в файле `/var/log/audit/audit.log`. Проверьте это утверждение самостоятельно.
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`.
 17. Выполните перезапуск веб-сервера Apache. Произошёл сбой? Поясните почему?
 18. Проанализируйте лог-файлы: `tail -nl /var/log/messages`.
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.
 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.
 20. Попробуйте запустить веб-сервер Apache ещё раз. Поняли ли вы, почему он сейчас запустился, а в предыдущем случае не смог?
 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».
 22. Исправьте обратно конфигурационный файл `apache`, вернув `Listen 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.
 24. Удалите файл `/var/www/html/test.html`: `rm /var/www/html/test.html`

Теоретическое введение

Так как по умолчанию пользователи CentOS являются свободными от типа (unconfined в переводе с англ. означает свободный), созданным файлам по умолчанию сопоставляется 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 /rgos файлы, относящиеся к процессам, могут иметь роль system_r. Если активна политика MLS, то могут использоваться и другие роли, например, secadm_r. Тип httpd_sys_content_t позволяет процессу httpd получить доступ к файлу. Благодаря наличию последнего типа можно получить доступ к файлу при обращении к нему через браузер.

Выполнение лабораторной работы

1. Вошла в систему с полученными учётными данными и убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (рис. 1).

```
[root@dsalkhimova dsalkhimova]# getenforce
Enforcing
[root@dsalkhimova dsalkhimova]# 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:         disabled
Policy MLS status:             enabled
Policy deny_unknown status:    allowed
Max kernel policy version:     31
[root@dsalkhimova dsalkhimova]# █
```

Рис. 1: Проверка работы SELinux

2. Обратилась с помощью браузера к веб-серверу, запущенному на моем компьютере, и убедилась, что последний работает: `service httpd status` или `/etc/rc.d/init.d/httpd status` (рис. 2).

```

[root@dsalkhimova conf]# 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; vendor preset: disabled)
   Active: active (running) since Tue 2022-10-11 20:25:41 MSK; 17min ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Process: 14772 ExecStop=/bin/kill -WINCH ${MAINPID} (code=exited, status=0/SUCCESS)
 Main PID: 14780 (httpd)
    Status: "Total requests: 0; Current requests/sec: 0; Current traffic:  0 B/sec"
      Tasks: 6
   CGroup: /system.slice/httpd.service
           └─14780 /usr/sbin/httpd -DFOREGROUND
             └─14781 /usr/sbin/httpd -DFOREGROUND
               └─14782 /usr/sbin/httpd -DFOREGROUND
                 └─14785 /usr/sbin/httpd -DFOREGROUND
                   └─14786 /usr/sbin/httpd -DFOREGROUND
                     └─14787 /usr/sbin/httpd -DFOREGROUND

Oct 11 20:25:40 dsalkhimova.localdomain systemd[1]: Stopped The...

```

Рис. 2: Проверка работы веб-сервера

3. Нашла веб-сервер Apache в списке процессов, определила его контекст безопасности (`unconfined_t`). Использовала команду `ps auxZ | grep httpd` (рис. 3).

```

[root@dsalkhimova dsalkhimova]# ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root      16166  0.1  0.6 314768 6388 ?        Ss   20:59  0
:01 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  16171  0.0  0.5 316988 5792 ?        S    20:59  0
:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  16172  0.0  0.5 316988 5468 ?        S    20:59  0
:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  16173  0.0  0.5 316988 5468 ?        S    20:59  0
:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  16174  0.0  0.5 316988 5468 ?        S    20:59  0
:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  16175  0.0  0.5 316988 5468 ?        S    20:59  0
:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache  16274  0.0  0.5 316988 5468 ?        S    21:01  0
:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-c0:c1023 root 16840 0.0  0.0 112808 968 pts/0 R+
21:26  0:00 grep --color=auto httpd
[root@dsalkhimova dsalkhimova]#

```

Рис. 3: Контекст Apache

4. Посмотрела текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd` (рис. 4).

```
[root@dsalkhimova dsalkhimova]# sestatus -b igrep 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: disabled
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
```

Рис. 4: Состояние переключателей SELinux для Apache

- Посмотрела статистику по политике с помощью команды `seinfo`, также определила множество пользователей, ролей, типов (рис. 5).

```
zoneminder_run_sudo off
[root@dsalkhimova dsalkhimova]# 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
```

Рис. 5: Статистика по политике

- Определила тип файлов и поддиректорий, находящихся в директории `/var/www`, с помощью команды `ls -lZ /var/www` (рис. 6).
- Определила тип файлов, находящихся в директории `/var/www/html`: `ls -lZ /var/www/html` (рис. 6) - нет файлов и поддиректорий в данной дирек-

тории.

8. Определила круг пользователей, которым разрешено создание файлов в директории `/var/www/html` (рис. 6) - только владельцу.

```
[root@dsalkhimova dsalkhimova]# 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
[root@dsalkhimova dsalkhimova]# ls -lZ /var/www/html
[root@dsalkhimova dsalkhimova]#
```

Рис. 6: Типы файлов, поддиректорий и права для `/www` и `/html`

9. Создала от имени суперпользователя html-файл `/var/www/html/test.html` следующего содержания:

```
<html>
<body>test</body>
</html> (рис. 7)
```

```
[root@dsalkhimova dsalkhimova]# touch /var/www/html/test.html
[root@dsalkhimova dsalkhimova]# nano /var/www/html/test.html
```

Рис. 7: Создание файла `test.html`

10. Проверила контекст созданного файла командой `ls -Z /var/www/html/test.html`. Контекст, присваиваемый по умолчанию вновь созданным файлам в директории `/var/www/html` - `httpd_sys_content_t` (рис. 8).

```
[root@dsalkhimova dsalkhimova]# ls -lZ /var/www/html/test.html
-rw-r--r--. root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
```

Рис. 8: Контекст файла `test.html`

11. Обратилась к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Убедилась, что файл был успешно отображён (рис. 9).

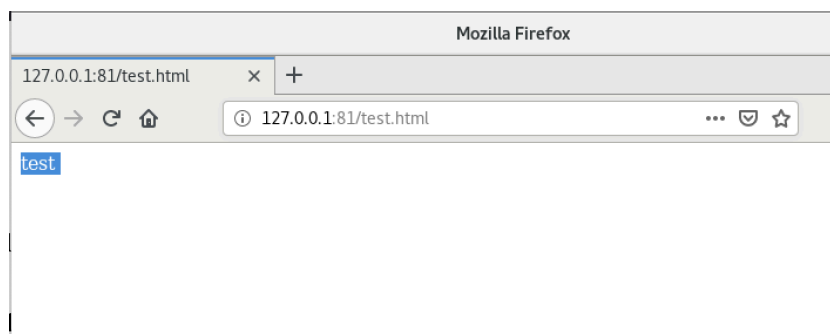


Рис. 9: Открытие test.html через браузер

12. Справку по команде `man httpd_selinux` не удалось получить, поэтому я изучила, какие контексты файлов определены для httpd с помощью интернета (рис. 10).

```
[root@dsalkhimova dsalkhimova]# man httpd_selinux
No manual entry for httpd_selinux
```

Рис. 10: Вызов справки httpd

13. Изменила контекст файла `/var/www/html/test.html` с `httpd_sys_content_t` на `samba_share_t`:

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

`ls -Z /var/www/html/test.html`. После этого проверила, что контекст поменялся (рис. 11).

```
[root@dsalkhimova dsalkhimova]# chcon -t samba_share_t /var/www/html/test.html
[root@dsalkhimova dsalkhimova]# ls -lZ /var/www/html/test.html
-rw-r--r--. root root unconfined_u:object_r:samba_share_t:s0 /var/www/html/test.html
```

Рис. 11: Смена контекста файла test.html

14. Попробуйте ещё раз получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1/test.html`. Получила сообщение об ошибке (рис. 12).

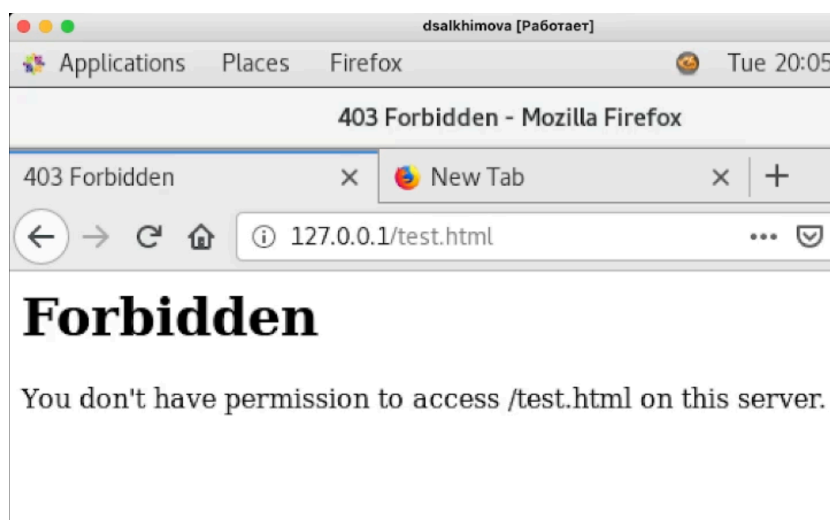


Рис. 12: Повторное открытие test.html через браузер

15. Файл не был отображён, т.к. политика ролевого разделения доступа RBAC используется процессами, но не файлами, поэтому роли не имеют никакого значения для файлов. Тип `httpd_sys_content_t` позволял процессу `httpd` получить доступ к файлу, а тип `samba_share_t` - нет. Просмотрела системный лог-файл: `tail /var/log/messages` (рис. 13).

```
[root@dsalkhimova dsalkhimova]# tail /var/log/messages
Oct 11 20:05:50 dsalkhimova dbus[694]: [system] Activating service name='org.fedoraproject.Setroubleshootd' (using servicehelper)
Oct 11 20:05:57 dsalkhimova dbus[694]: [system] Successfully activated service 'org.fedoraproject.Setroubleshootd'
Oct 11 20:06:00 dsalkhimova setroubleshoot: failed to retrieve rpm info for /var/www/html/test.html
Oct 11 20:06:00 dsalkhimova setroubleshoot: SELinux is preventing httpd from getattr access on the file /var/www/html/test.html. For complete SELinux messages run: sealert -l 6895de24-0633-418d-982c-8889ec3d3a3b
Oct 11 20:06:00 dsalkhimova python: SELinux is preventing 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 run 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' #012
```

Рис. 13: Просмотр системного лога

В системе были запущенными процессы `setroubleshootd` и `auditd`. Посмотрела ошибки, аналогичные указанным выше, в файле `/var/log/audit/audit.log`.

16. Попробовала 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 нашла строчку Listen 80 и заменила её на Listen 81 (рис. 14, рис. 15, рис. 16).

```
[root@dsalkhimova etc]# cd httpd
[root@dsalkhimova httpd]# ls
conf  conf.d  conf.modules.d  logs  modules  run
[root@dsalkhimova httpd]# nano conf
[root@dsalkhimova httpd]# ls | grep httpd
[root@dsalkhimova httpd]# nano httpd.conf
```

Рис. 14: Открытие httpd.conf

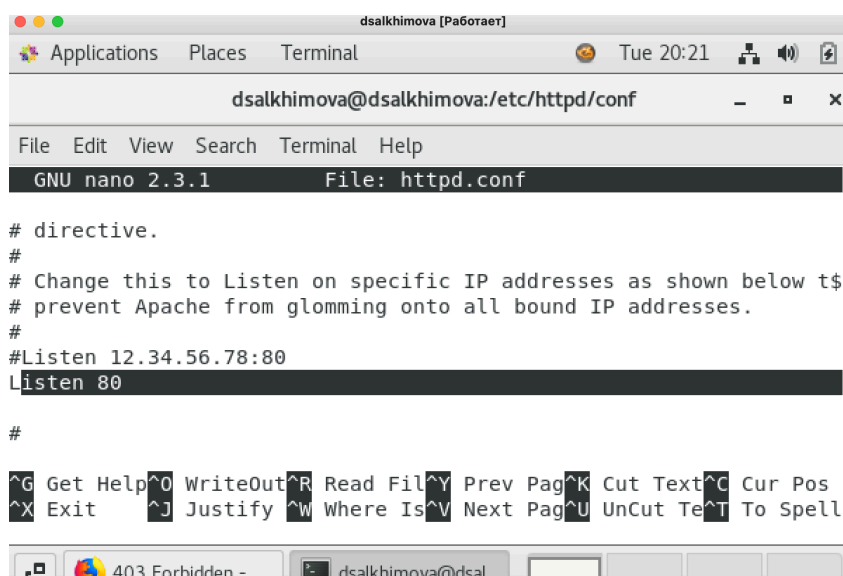


Рис. 15: httpd.conf до исправления

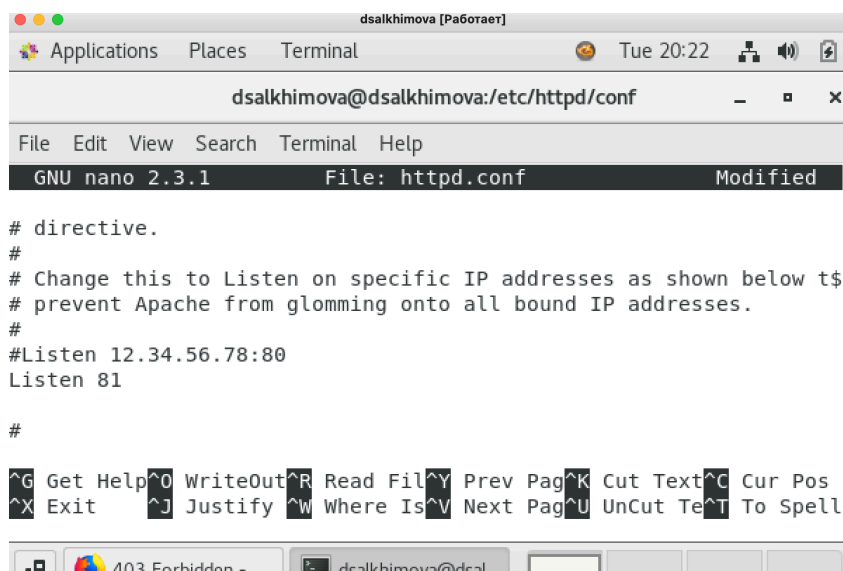


Рис. 16: httpd.conf после исправления

17. Выполнила перезапуск веб-сервера Apache (рис. 17).

```

[dsalkhimova@dsalkhimova root]$ service httpd restart
Redirecting to /bin/systemctl restart httpd.service

```

Рис. 17: Перезапуск Apache

18. Проанализировала лог-файлы: `tail -nl /var/log/messages` (рис. 18).

```

[root@dsalkhimova ~]# tail -nl /var/log/messages
Oct 11 21:58:04 dsalkhimova systemd: Started Hostname Service.
[root@dsalkhimova ~]#

```

Рис. 18: Просмотр логов

Просмотрела файлы `/var/log/http/error_log` (рис. 20), `/var/log/http/access_log` (рис. 19) и `/var/log/audit/audit.log`. Записи появились в `error_log` и `access_log` (рис. 21).


```

dsalkhimova@dsalkhimova:~/var/log/httpd
File Edit View Search Terminal Help
GNU nano 2.3.1 File: error_log

[Tue Oct 11 20:53:29.019971 2022] [core:notice] [pid 15984] SELinux policy enabled; httpd ru$
[Tue Oct 11 20:53:29.022620 2022] [suexec:notice] [pid 15984] AH01232: suEXEC mechanism enab$
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using$
[Tue Oct 11 20:53:29.158238 2022] [lbmethod_heartbeat:notice] [pid 15984] AH02282: No slotme$
[Tue Oct 11 20:53:29.233055 2022] [mpm_prefork:notice] [pid 15984] AH00163: Apache/2.4.6 (Ce$
[Tue Oct 11 20:53:29.233133 2022] [core:notice] [pid 15984] AH00094: Command line: '/usr/sbi$
[Tue Oct 11 20:59:44.839103 2022] [mpm_prefork:notice] [pid 15984] AH00170: caught SIGWINCH,$
[Tue Oct 11 20:59:46.458141 2022] [core:notice] [pid 16166] SELinux policy enabled; httpd ru$
[Tue Oct 11 20:59:46.468464 2022] [suexec:notice] [pid 16166] AH01232: suEXEC mechanism enab$
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using$
[Tue Oct 11 20:59:46.602129 2022] [lbmethod_heartbeat:notice] [pid 16166] AH02282: No slotme$
[Tue Oct 11 20:59:46.711392 2022] [mpm_prefork:notice] [pid 16166] AH00163: Apache/2.4.6 (Ce$
[Tue Oct 11 20:59:46.711473 2022] [core:notice] [pid 16166] AH00094: Command line: '/usr/sbi$

G Get Help      ^O WriteOut     ^R Read File    ^V Prev Page    ^K Cut Text     ^C Cur Pos
X Exit          ^J Justify      ^W Where Is     ^N Next Page    ^U UnCut Text   ^T To Spell

```

Рис. 19: Просмотр access_log

```

dsalkhimova@dsalkhimova:~/var/log/httpd
File Edit View Search Terminal Help
GNU nano 2.3.1 File: access_log

127.0.0.1 - - [11/Oct/2022:19:21:50 +0300] "GET / HTTP/1.1" 403 4897 "-" "Mozilla/5.0 (X11; $
127.0.0.1 - - [11/Oct/2022:19:21:51 +0300] "GET /noindex/css/bootstrap.min.css HTTP/1.1" 200$
127.0.0.1 - - [11/Oct/2022:19:21:51 +0300] "GET /noindex/css/open-sans.css HTTP/1.1" 200 508$
127.0.0.1 - - [11/Oct/2022:19:21:51 +0300] "GET /images/poweredby.png HTTP/1.1" 200 3956 "ht$
127.0.0.1 - - [11/Oct/2022:19:21:51 +0300] "GET /images/apache_pb.gif HTTP/1.1" 200 2326 "ht$
127.0.0.1 - - [11/Oct/2022:19:21:51 +0300] "GET /noindex/css/fonts/Light/OpenSans-Light.woff$
127.0.0.1 - - [11/Oct/2022:19:21:51 +0300] "GET /noindex/css/fonts/Bold/OpenSans-Bold.woff H$
127.0.0.1 - - [11/Oct/2022:19:21:52 +0300] "GET /noindex/css/fonts/Light/OpenSans-Light.ttf $
127.0.0.1 - - [11/Oct/2022:19:21:52 +0300] "GET /noindex/css/fonts/Bold/OpenSans-Bold.ttf HT$
::1 - - [11/Oct/2022:19:21:57 +0300] "OPTIONS * HTTP/1.0" 200 - "-" "Apache/2.4.6 (CentOS) P$
::1 - - [11/Oct/2022:19:21:58 +0300] "OPTIONS * HTTP/1.0" 200 - "-" "Apache/2.4.6 (CentOS) P$
127.0.0.1 - - [11/Oct/2022:20:02:58 +0300] "GET /test.html HTTP/1.1" 200 33 "-" "Mozilla/5.0$
127.0.0.1 - - [11/Oct/2022:20:05:47 +0300] "GET /test.html HTTP/1.1" 403 211 "-" "Mozilla/5.$
127.0.0.1 - - [11/Oct/2022:21:01:16 +0300] "GET /test.html HTTP/1.1" 200 33 "-" "Mozilla/5.0$

```

Рис. 20: Просмотр error_log

```

[root@dsalkhimova ~]# cd /var
[root@dsalkhimova var]# cd log
[root@dsalkhimova log]# cd audit
[root@dsalkhimova audit]# ls
audit.log
[root@dsalkhimova audit]# tail -n1 audit.log
type=USER_END msg=audit(1665509401.588:1047): pid=14974 uid=0 auid=0 ses=51 subj=system_u:sys
tem_r:crond t:s0-s0:c0.c1023 msg='op=PAM:session_close grantors=pam_loginuid,pam_keyinit,pam
limits,pam_systemd acct="root" exe="/usr/sbin/crond" hostname=? addr=? terminal=cron res=succ
ess'

```

Рис. 21: Просмотр audit.log

19. Выполнила команду `semanage port -a -t http_port_t -p tcp 81` и проверила список портов командой `semanage port -l | grep http_port_t` (рис. 22). Порт 81 присутствует в списке.

```
[root@dsalkhimova dsalkhimova]# semanage port -a -t http_port_t -p tcp 81
ValueError: Port tcp/81 already defined
[root@dsalkhimova dsalkhimova]# 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
```

Рис. 22: Добавление и просмотр наличия 81 порта

20. Попробовала запустить веб-сервер Apache ещё раз (рис. 23). Запуск прошёл успешно.

```
[root@dsalkhimova dsalkhimova]# service httpd restart
Redirecting to /bin/systemctl restart httpd.service
```

Рис. 23: Запуск Apache

21. Вернула контекст `httpd_sys_content_t` к файлу `/var/www/html/test.html`: `chcon -t httpd_sys_content_t /var/www/html/test.html` (рис. 24).

```
[root@dsalkhimova dsalkhimova]# chcon -t httpd_sys_content_t /var/www/html/test.html
[root@dsalkhimova dsalkhimova]# ls -lZ /var/www/html/test.html
-rw-r--r-- root root unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[root@dsalkhimova dsalkhimova]#
```

Рис. 24: Возвращение контекста файлу test.html

После этого попробовала получить доступ к файлу через веб-сервер, введя в браузере адрес `http://127.0.0.1:81/test.html`. Открылось содержимое файла — слово «test» (рис. 25).

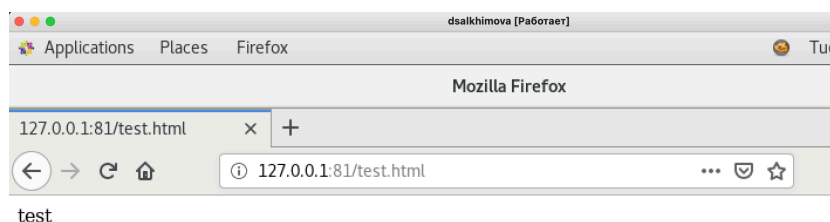


Рис. 25: Открытие файла test.html через браузер

22. Исправила обратно конфигурационный файл apache, вернув Listen80 (рис. 26).

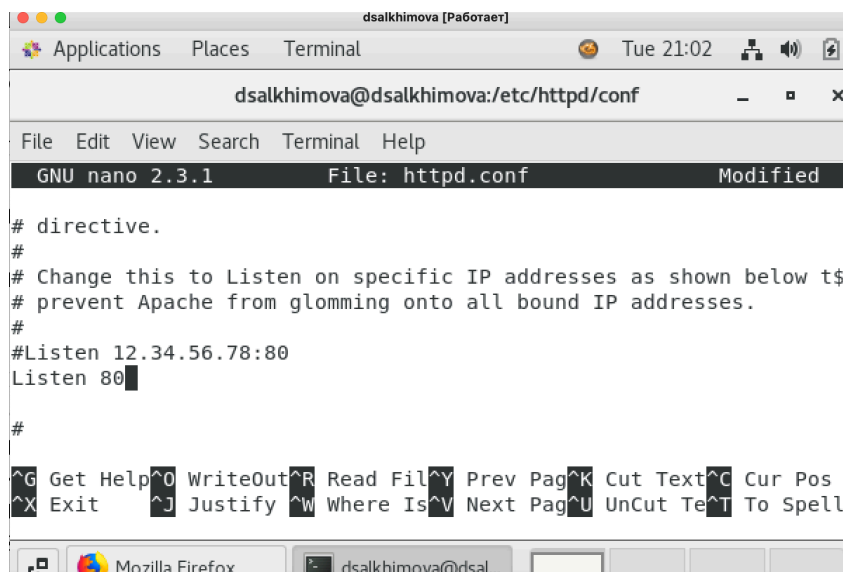


Рис. 26: Возвращение настроек на 80 порт httpd.conf

23. Попробовала удалить привязку http_port_t к 81 порту: `semanage port -d -t http_port_t -p tcp 81` - операция запрещена (рис. 27). Порт 81 остался в списке.

```
[root@dsalkhimova conf]# semanage port -d -t http_port_t -p tcp 81
ValueError: Port tcp/81 is defined in policy, cannot be deleted
```

Рис. 27: Попытка удаления 81 порта

24. Удалила файл `/var/www/html/test.html`: `rm /var/www/html/test.html` (рис. 28).

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

Рис. 28: Удаление файла test.html

Выводы

В процессе выполнения данной лабораторной работы я приобрела навыки администрирования ОС Linux. Получила первое практическое знакомство с технологией SELinux и проверила работу SELinux на практике совместно с веб-сервером Apache.

Список литературы

1. Описание лабораторной работы 6 - URL: https://esystem.rudn.ru/pluginfile.php/1652173/mod_resource/content/2/006-lab_selinux.pdf
2. Активация Apache - URL: <https://stackoverflow.com/questions/51108495/fresh-install-httpd-service-unit-not-found>