

Отчёт по лабораторной работе №6

Информационная безопасность

Мандатное разграничение прав в Linux

Хусаинова Динара Айратовна НПИбд-02-21, 1032212283

Содержание

Цель работы	4
Теоретическое введение	5
Выполнение лабораторной работы	7
Выводы	17
Список литературы. Библиография	18

Список иллюстраций

1	getenforce и sestatus	7
2	status	8
3	ps auxZ grep httpd and sestatus	9
4	ls -lZ /var/www	10
5	создание файла	10
6	проверка контекста	11
7	обращение к файлу	11
8	man httpd_selinux	12
9	проверка контекста файла	12
10	Изменение контекста файла	13
11	You don't have permission to access	13
12	просмотр логов	14
13	замена порта	15
14	сбой	15
15	сбой	16
16	удаление файла	16

Цель работы

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

Теоретическое введение

1. **SELinux (Security-Enhanced Linux)** обеспечивает усиление защиты путем внесения изменений как на уровне ядра, так и на уровне пространства пользователя, что превращает ее в действительно «непробиваемую» операционную систему. Впервые эта система появилась в четвертой версии CentOS, а в 5 и 6 версии реализация была существенно дополнена и улучшена.

SELinux имеет три основных режим работы:

- **Enforcing:** режим по умолчанию. При выборе этого режима все действия, которые каким-то образом нарушают текущую политику безопасности, будут блокироваться, а попытка нарушения будет зафиксирована в журнале.
- **Permissive:** в случае использования этого режима, информация о всех действиях, которые нарушают текущую политику безопасности, будут зафиксированы в журнале, но сами действия не будут заблокированы.
- **Disabled:** полное отключение системы принудительного контроля доступа.

Политика SELinux определяет доступ пользователей к ролям, доступ ролей к доменам и доступ доменов к типам. Контекст безопасности — все атрибуты SELinux — роли, типы и домены.

2. **Apache** — это свободное программное обеспечение, с помощью которого можно создать веб-сервер. Данный продукт возник как доработанная версия другого HTTP-клиента от национального центра суперкомпьютерных приложений (NCSA).

Для чего нужен Apache сервер:

- чтобы открывать динамические PHP-страницы,
- для распределения поступающей на сервер нагрузки,
- для обеспечения отказоустойчивости сервера,
- чтобы потренироваться в настройке сервера и запуске PHP-скриптов.

Apache является кроссплатформенным ПО и поддерживает такие операционные системы, как Linux, BSD, MacOS, Microsoft, BeOS и другие.

Выполнение лабораторной 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`.

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

Рис. 1: `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`.

```
[dakhusainova@dakhusainova ~]$ sudo systemctl start httpd
[dakhusainova@dakhusainova ~]$ 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: enabled)
  Active: active (running) since Thu 2024-10-10 18:59:27 MSK; 1min 15s ago
  Docs: man:httpd.service(8)
  Main PID: 3387 (httpd)
  Status: "Total requests: 0; Idle/Busy workers 100/0; Requests served 0/0"
  Tasks: 177 (limit: 12195)
  Memory: 22.3M
  CPU: 72ms
  CGroup: /system.slice/httpd.service
          └─3387 /usr/sbin/httpd -DFOREGROUND
             3388 /usr/sbin/httpd -DFOREGROUND
             3392 /usr/sbin/httpd -DFOREGROUND
             3393 /usr/sbin/httpd -DFOREGROUND
             3394 /usr/sbin/httpd -DFOREGROUND

Oct 10 18:59:25 dakhusainova.localdomain systemd[1]: Starting The Apache HTTP Server: httpd.
Oct 10 18:59:27 dakhusainova.localdomain systemd[1]: Started The Apache HTTP Server: httpd.
Oct 10 18:59:27 dakhusainova.localdomain httpd[3387]: Server configured for IPv4 and IPv6.
lines 1-19/19 (END)...skipping...
• httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: enabled)
  Active: active (running) since Thu 2024-10-10 18:59:27 MSK; 1min 15s ago
  Docs: man:httpd.service(8)
  Main PID: 3387 (httpd)
  Status: "Total requests: 0; Idle/Busy workers 100/0; Requests served 0/0"
  Tasks: 177 (limit: 12195)
  Memory: 22.3M
  CPU: 72ms
```

Рис. 2: status

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».


```
[dakhusainova@dakhusainova ~]$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 3387 0.0 0.5 203
64 11528 ? Ss 18:59 0:00 /usr/sbin/httpd -DFOREGROUN
D
system_u:system_r:httpd_t:s0 apache 3388 0.0 0.3 220
96 7144 ? S 18:59 0:00 /usr/sbin/httpd -DFOREGROUN
D
system_u:system_r:httpd_t:s0 apache 3392 0.0 0.5 9815
20 11092 ? Sl 18:59 0:00 /usr/sbin/httpd -DFOREGROUN
D
system_u:system_r:httpd_t:s0 apache 3393 0.0 0.5 9815
20 11096 ? Sl 18:59 0:00 /usr/sbin/httpd -DFOREGROUN
D
system_u:system_r:httpd_t:s0 apache 3394 0.0 0.6 1112
656 13288 ? Sl 18:59 0:00 /usr/sbin/httpd -DFOREGROUN
D
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 dakhusa+
3760 0.0 0.1 221664 2176 pts/0 S+ 19:01 0:00 grep --color=au
to httpd
[dakhusainova@dakhusainova ~]$ sestatus -bigrep httpd
sestatus: invalid option -- 'i'

Usage: sestatus [OPTION]

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

Without options, show SELinux status.
[dakhusainova@dakhusainova ~]$
```

Рис. 3: ps auxZ | grep httpd and sestatus

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

```
[dakhusainova@dakhusainova ~]$ 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

[dakhusainova@dakhusainova ~]$ 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 19:30 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s
0 6 Aug  8 19:30 html
[dakhusainova@dakhusainova ~]$ ls -lZ /var/www/html
total 0
[dakhusainova@dakhusainova ~]$
```

Рис. 4: ls -lZ /var/www

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

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

Рис. 5: создание файла

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

```
[dakhusainova@dakhusainova ~]$ ls -lZ /var/www/html
total 4
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 35 0
ct 10 19:07 test.html
[dakhusainova@dakhusainova ~]$
```

Рис. 6: проверка контекста

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

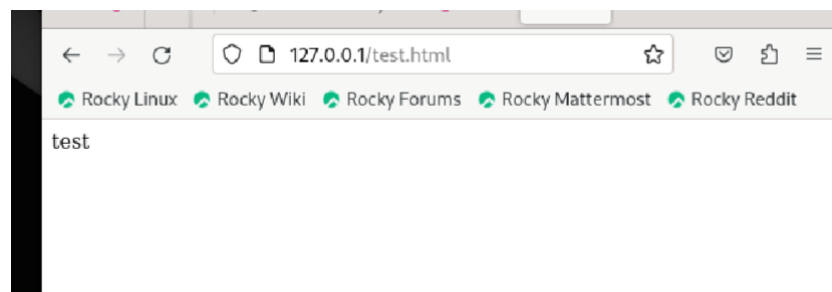
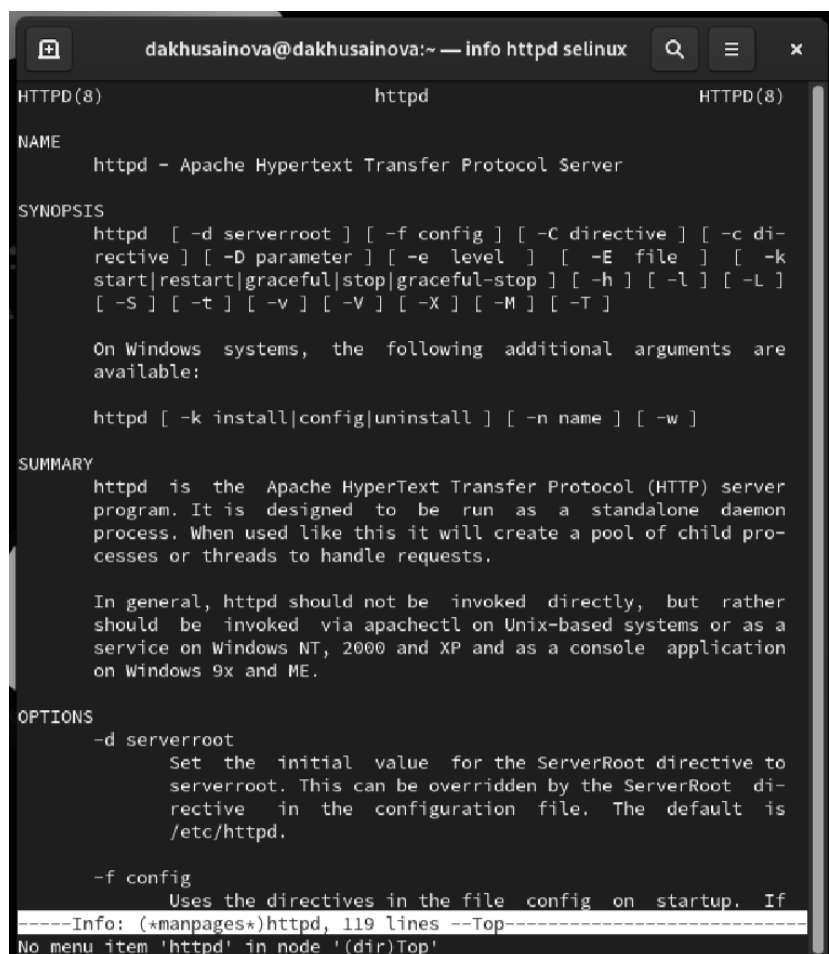


Рис. 7: обращение к файлу

12. Изучите справку `man httpd_selinux` и выясните, какие контексты файлов определены для `httpd`.



```
dakhusainova@dakhusainova:~ — info httpd selinux
HTTPD(8)                                httpd                                HTTPD(8)

NAME
    httpd - Apache Hypertext Transfer Protocol Server

SYNOPSIS
    httpd [ -d serverroot ] [ -f config ] [ -C directive ] [ -c di-
    rective ] [ -D parameter ] [ -e level ] [ -E file ] [ -k
    start|restart|graceful|stop|graceful-stop ] [ -h ] [ -l ] [ -L ]
    [ -S ] [ -t ] [ -v ] [ -V ] [ -X ] [ -M ] [ -T ]

    On Windows systems, the following additional arguments are
    available:

    httpd [ -k install|config|uninstall ] [ -n name ] [ -w ]

SUMMARY
    httpd is the Apache HyperText Transfer Protocol (HTTP) server
    program. It is designed to be run as a standalone daemon
    process. When used like this it will create a pool of child pro-
    cesses or threads to handle requests.

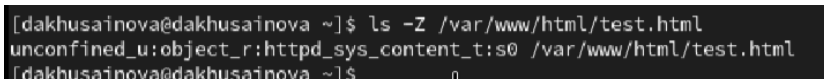
    In general, httpd should not be invoked directly, but rather
    should be invoked via apachectl on Unix-based systems or as a
    service on Windows NT, 2000 and XP and as a console application
    on Windows 9x and ME.

OPTIONS
    -d serverroot
        Set the initial value for the ServerRoot directive to
        serverroot. This can be overridden by the ServerRoot di-
        rective in the configuration file. The default is
        /etc/httpd.

    -f config
        Uses the directives in the file config on startup. If
    -----Info: (*manpages*)httpd, 119 lines --Top-----
    No menu item 'httpd' in node '(dir)Top'
```

Рис. 8: man httpd_selinux

Сопоставьте их с типом файла test.html. Проверить контекст файла можно командой ls -Z. ls -Z /var/www/html/test.html



```
[dakhusainova@dakhusainova ~]$ ls -Z /var/www/html/test.html
unconfined_u:object_r:httpd_sys_content_t:s0 /var/www/html/test.html
[dakhusainova@dakhusainova ~]$
```

Рис. 9: проверка контекста файла

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 разде-

ления 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`

```
[dakhusainova@dakhusainova ~]$ sudo chcon -t samba_share_t /var/www/html/test.html
[dakhusainova@dakhusainova ~]$ ls -lZ /var/www/html/test.html
-rw-r--r--. 1 root root unconfined_u:object_r:samba_share_t:s0 35 Oct 10
19:07 /var/www/html/test.html
[dakhusainova@dakhusainova ~]$ ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 35 Oct 10 19:07 /var/www/html/test.html
[dakhusainova@dakhusainova ~]$
```

Рис. 10: 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.`

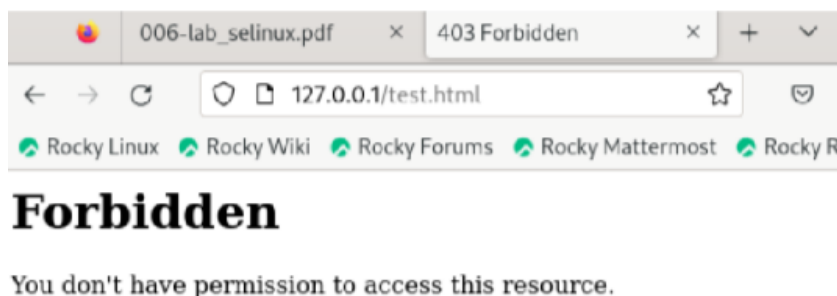


Рис. 11: You don't have permission to access

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.

```
[dakhusainova@dakhusainova ~]$ sudo tail /var/log/messages
Oct 10 19:21:46 dakhusainova systemd[1]: Started SETroubleshoot daemon for processing new SELinux denial logs.
Oct 10 19:21:46 dakhusainova setroubleshoot[4506]: failed to retrieve rpm info for path '/var/www/html/test.html':
Oct 10 19:21:46 dakhusainova systemd[1]: Created slice Slice /system/dbus-1.1-org.fedoraproject.SetroubleshootPrivileged.
Oct 10 19:21:46 dakhusainova systemd[1]: Started dbus-1.1-org.fedoraproject.SetroubleshootPrivileged@0.service.
Oct 10 19:21:47 dakhusainova setroubleshoot[4506]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/www/html/test.html. For complete SELinux messages run: sealert -l 68690a52-e372-4bef-8939-553277261788
Oct 10 19:21:47 dakhusainova setroubleshoot[4506]: SELinux is preventing /usr/sbin/httpd from getattr access on the file /var/www/html/test.html.
***** 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#012***** Plugin catchall (1.41 confidence) suggests *****
*****#012#012If you believe that httpd should be allowed getattr access on the test.html file by default.#012Then you should report this as a bug.#012You can generate a local policy module to allow this access.#012Do#012allow this access for now by executing:#012# ausearch -c 'httpd' --raw | audit2allow -M my-httpd#012# semodule -X 300 -i my-httpd.pp#012
Oct 10 19:21:47 dakhusainova setroubleshoot[4506]: SELinux is preventing
```

Рис. 12: просмотр 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`.

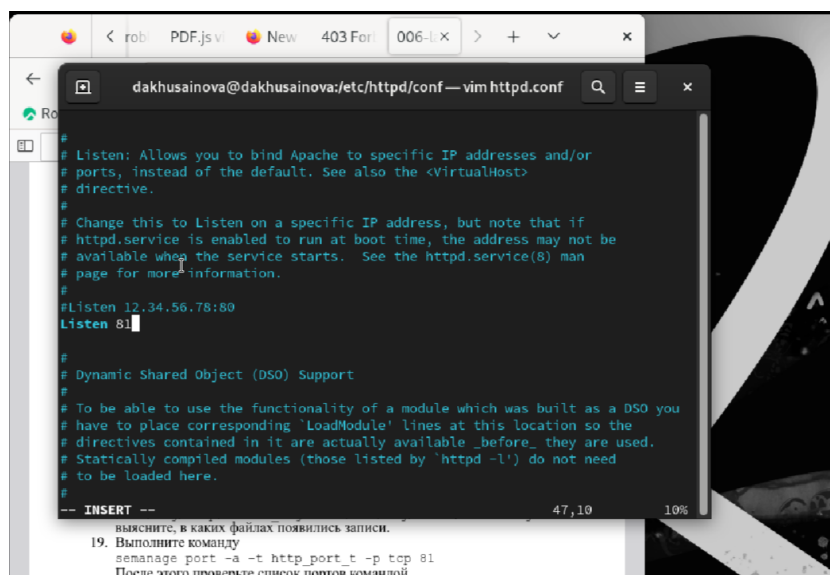


Рис. 13: замена порта

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

Unable to connect

Firefox can't establish a connection to the server at 127.0.0.1:81.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the web.

Try Again

Рис. 14: сбой

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.

```
[dakhusainova@dakhusainova conf]$ sudo semanage port -l | grep http_port_t
http_port_t          tcp      80, 81, 443, 488, 8008, 8009, 8443, 9
pegasus_http_port_t  tcp      5988
[dakhusainova@dakhusainova conf]$
```

Рис. 15: сбой

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`

```
[dakhusainova@dakhusainova conf]$ sudo rm /var/www/html/test.html
[dakhusainova@dakhusainova conf]$
```

Рис. 16: удаление файла

Выводы

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

Список литературы. Библиография

[0] Методические материалы курса

[1] Права доступа: <https://codechick.io/tutorials/unix-linux/unix-linux-permissions>