

Мандатное разграничение прав в Linux

Основы информационной безопасности

Симонова Полина Игоревна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	8
4	Выводы	19
5	Список литературы	20

Список иллюстраций

3.1	проверка режима работы SELinux	8
3.2	Проверка работы Apache	9
3.3	Контекст безопасности Apache	9
3.4	Состояние переключателей SELinux	10
3.5	Статистика по политике	11
3.6	Типы поддиректорий	11
3.7	Типы файлов	11
3.8	Создание файла	12
3.9	Контекст файла	12
3.10	Отображение файла	13
3.11	Изучение справки по команде	14
3.12	Изменение контекста	14
3.13	Отображение файла	14
3.14	Попытка прочесть лог-файл	15
3.15	Изменение файла	15
3.16	Изменение порта	16
3.17	Попытка прослушивания другого порта	16
3.18	Проверка лог-файлов	16
3.19	Проверка лог-файлов	17
3.20	Проверка портов	17
3.21	Перезапуск сервера	17
3.22	Проверка сервера	18
3.23	Проверка порта 81	18
3.24	Удаление файла	18

Список таблиц

1 Цель работы

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

2 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. Более подробно см. в [f].

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.

3 Выполнение лабораторной работы

Вошла в систему под своей учетной записью. Убедилась, что SELinux работает в режиме enforcing политики targeted с помощью команд `getenforce` и `sestatus` (рис. [fig:001]).

```
pisimonova@pisimonova:~$ getenforce
Enforcing
pisimonova@pisimonova:~$ 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
pisimonova@pisimonova:~$
```

Рисунок 3.1: проверка режима работы SELinux

Запускаю сервер `apache`, далее обращаюсь с помощью браузера к веб-серверу, запущенному на компьютере, он работает, что видно из вывода команды `service httpd status` (рис. [fig:002]).


```

Active: active (running) since Sat 2026-02-14 15:57:54 MSK
; 1min 17s ago
Invocation: d18e6a1fc1f943a3aa8967eda6c1661f
Docs: man:httpd.service(8)
Main PID: 14998 (httpd)
Status: "Total requests: 0; Idle/Busy workers 100/0;Reques
ts/sec: 0; Bytes served/"
Tasks: 177 (limit: 10544)
Memory: 17.9M (peak: 18.1M)
CPU: 98ms
CGroup: /system.slice/httpd.service
├─14998 /usr/sbin/httpd -DFOREGROUND
├─15041 /usr/sbin/httpd -DFOREGROUND
├─15042 /usr/sbin/httpd -DFOREGROUND
├─15045 /usr/sbin/httpd -DFOREGROUND
└─15046 /usr/sbin/httpd -DFOREGROUND

фев 14 15:57:33 pisimonova systemd[1]: Starting httpd.service -
The Apache HTTP Server.>
фев 14 15:57:33 pisimonova (httpd)[14998]: httpd.service: Refer
enced but unset environm
фев 14 15:57:43 pisimonova httpd[14998]: AH00558: httpd: Could
not reliably determine t
фев 14 15:57:54 pisimonova systemd[1]: Started httpd service -

```

Рисунок 3.2: Проверка работы Apache

С помощью команды `ps auxZ | grep httpd` нашла веб-сервер Apache в списке процессов. Его контекст безопасности - `httpd_t` (рис. [fig:003]).

```

pisimonova@pisimonova:~$ ps auxZ | grep httpd
system_u:system_r:httpd_t:s0 root 14998 0.0 0.3 19136 6716 ?
Ss 15:57 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 15041 0.0 0.2 18792 3728 ?
S 15:57 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 15042 0.0 0.3 1109220 5368 ?
Sl 15:57 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 15045 0.0 0.3 978084 5356 ?
Sl 15:57 0:00 /usr/sbin/httpd -DFOREGROUND
system_u:system_r:httpd_t:s0 apache 15046 0.0 0.3 978084 5352 ?
Sl 15:57 0:00 /usr/sbin/httpd -DFOREGROUND
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 pisimon+ 15951 0.0 0.1 227
712 2168 pts/0 S+ 16:06 0:00 grep --color=auto httpd

```

Рисунок 3.3: Контекст безопасности Apache

Просмотрела текущее состояние переключателей SELinux для Apache с помощью команды `sestatus -bigrep httpd` (рис. [fig:004]).

```

pisimonova@pisimonova:~$ 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               on
abrt_upload_watch_anon_write    on
auditadm_exec_content           on
authlogin_nsswitch_use_ldap     off
authlogin_radius                off
authlogin_yubikey               off
cdrecord_read_content           off
cluster_can_network_connect     off
cluster_manage_all_files        off
cluster_use_execmem             off
colord_use_nfs                  off
condor_tcp_network_connect      off

```

Рисунок 3.4: Состояние переключателей SELinux

Просмотрела статистику по политике с помощью команды `seinfo`. Множество пользователей - 8, ролей - 39, типов - 5135. (рис. [fig:005]).

```

Statistics for policy file: /sys/fs/selinux/policy
Policy Version:          33 (MLS enabled)
Target Policy:           selinux
Handle unknown classes:  allow
Classes:                 133   Permissions:          458
Sensitivities:           1     Categories:           1024
Types:                   4699  Attributes:           251
Users:                   8     Roles:                15
Booleans:                325   Cond. Expr.:          353
Allow:                   59647  Neverallow:            0
Auditallow:              158   Dontaudit:            7974
Type_trans:              246125 Type_change:            66
Type_member:              34   Range_trans:          3893
Role allow:              39    Role_trans:           318
Constraints:             70   Validatetrans:         0
MLS Constrain:           72   MLS Val. Tran:         0
Permissives:            24    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:                112   Portcon:               667
Netifcon:                0     Nodecon:               0

```

Рисунок 3.5: Статистика по политике

Типы поддиректорий, находящихся в директории `/var/www`, с помощью команды `ls -lZ /var/www` следующие: владелец - root, права на изменения только у владельца. Файлов в директории нет (рис. [fig:006]).

```

по команде «ls -lZ» можно получить дополнительную информацию.
pisimonova@pisimonova:~$ ls -lZ /var/www
итого 0
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_script_exec_t:s0 6 дек 10 03:
00 cgi-bin
drwxr-xr-x. 2 root root system_u:object_r:httpd_sys_content_t:s0 6 дек 10 03:
00 html
pisimonova@pisimonova:~$ ls -lZ /var/www/html

```

Рисунок 3.6: Типы поддиректорий

В директории `/var/www/html` нет файлов. (рис. [fig:007]).

```

00 html
pisimonova@pisimonova:~$ ls -lZ /var/www/html
итого 0

```

Рисунок 3.7: Типы файлов

Создать файл может только суперпользователь, поэтому от его имени создаем файл touch.html со следующим содержанием:

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

Листинг 1 (рис. [fig:008]).

```
pisimonova@pisimonova:~$ nano test.html
pisimonova@pisimonova:~$ sudo nano test.html
pisimonova@pisimonova:~$ sudo cat /var/www/html/test.html
pisimonova@pisimonova:~$ ls -lZ /var/www/html
итого 0
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 0 фев 14 16:
27 test.html
pisimonova@pisimonova:~$ sudo touch /var/www/html/test.html
pisimonova@pisimonova:~$ ls
sshgit      work        Загрузки   Общедоступные
sshgit.pub  Видео       Изображения 'Рабочий стол'
test.html   Документы  Музыка     Шаблоны
pisimonova@pisimonova:~$
pisimonova@pisimonova:~$ nano test.html
```

Рисунок 3.8: Создание файла

Проверяю контекст созданного файла. По умолчанию это httpd_sys_content_t (рис. [fig:009]).

```
pisimonova@pisimonova:~$ ls -lZ /var/www/html
итого 0
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 0 фев 14 16:
27 test.html
pisimonova@pisimonova:~$ sudo touch /var/www/html/test.html
```

Рисунок 3.9: Контекст файла

Обращаюсь к файлу через веб-сервер, введя в браузере адрес http://127.0.0.1/test.html. Файл был успешно отображён (рис. [fig:010]).

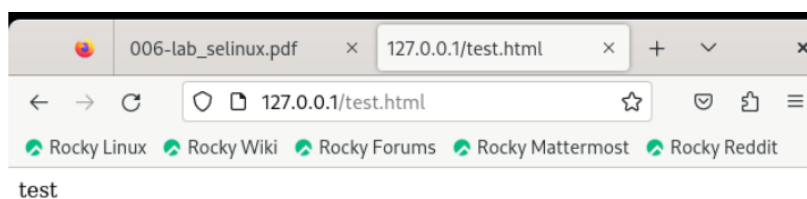


Рисунок 3.10: Отображение файла

Изучила справку `man httpd_selinux`. Рассмотрим полученный контекст детально. 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. (рис. [fig:011]).

```
HTTPD(8)                                httpd

NAME
    httpd - Apache Hypertext Transfer Protocol Server

SYNOPSIS
    httpd [ -d serverroot ] [ -f config ] [ -C directive ] [ -c directive ]
    [ -e level ] [ -E file ] [ -k start|restart|graceful|stop|graceful-stop ]
    [ -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 can be run as a standalone daemon process. When used like this it will
    create child processes or threads to handle requests.

    In general, httpd should not be invoked directly, but rather should
    be installed on Unix-based systems or as a service on Windows NT, 2000, or
```

Рисунок 3.11: Изучение справки по команде

Изменяю контекст файла `/var/www/html/test.html` `httpd_sys_content_t` на любой другой, к которому процесс `httpd` не должен иметь доступа, например, `samba_share_t`: `chcon -t samba_share_t /var/www/html/test.html`
`ls -Z /var/www/html/test.html` Контекст действительно поменялся (рис. [fig:012]).

```
pisimonova@pisimonova:~$ sudo chcon -t samba_share_t /var/www/html/test.html
pisimonova@pisimonova:~$ ls -lZ /var/www/html
итого 4
-rw-r--r--. 1 root root unconfined_u:object_r:samba_share_t:s0 33 фев 14 16:36 test.html
```

Рисунок 3.12: Изменение контекста

При попытке отображения файла в браузере получаем сообщение об ошибке (рис. [fig:013]).

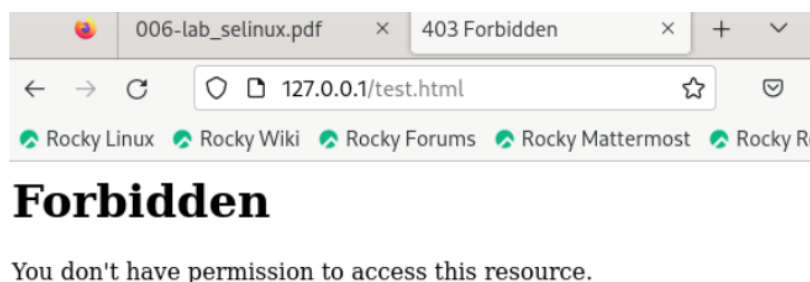


Рисунок 3.13: Отображение файла

файл не был отображён, хотя права доступа позволяют читать этот файл любому пользователю, потому что установлен контекст, к которому процесс httpd не должен иметь доступа.

Просматриваю log-файлы веб-сервера Apache и 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`. (рис. [fig:014]).

```
pisimonova@pisimonova:~$ ls -l /var/www/html/test.html
-rw-r--r--. 1 root root 33 фев 14 16:36 /var/www/html/test.html
pisimonova@pisimonova:~$ tail /var/log/messages
tail: невозможно открыть '/var/log/messages' для чтения: Отказано в доступе
pisimonova@pisimonova:~$ tail /var/log/audit/audit.log
tail: невозможно открыть '/var/log/audit/audit.log' для чтения: Отказано в доступе
pisimonova@pisimonova:~$ sudo tail /var/log/messages
Feb 14 16:39:55 pisimonova systemd[1]: dbus-:1.1-org.fedoraproject.SetroubleshootPrivileged@0.service: Deactivated successfully.
Feb 14 16:39:56 pisimonova systemd[1]: setroubleshootd.service: Deactivated successfully.
Feb 14 16:39:56 pisimonova systemd[1]: setroubleshootd.service: Consumed 401ms CPU time, 71.3M memory peak.
Feb 14 16:42:55 pisimonova systemd-logind[1038]: Existing logind session ID 3 used by new audit session, ignoring.
Feb 14 16:42:55 pisimonova systemd[1]: Created slice user-0.slice - User Slice of UID 0.
Feb 14 16:42:55 pisimonova systemd-logind[1038]: New session c22 of user root.
Feb 14 16:42:55 pisimonova systemd[1]: Starting user-runtime-dir@0.service - User Runtime Directory /run/user/0...
Feb 14 16:42:55 pisimonova systemd[1]: Finished user-runtime-dir@0.service - User Runtime Directory /run/user/0.
Feb 14 16:42:55 pisimonova systemd[1]: Starting user@0.service - User Manager for UID 0...
Feb 14 16:42:55 pisimonova systemd-logind[1038]: New session 22 of user root.
pisimonova@pisimonova:~$ sudo tail /var/log/audit/audit.log
type=SYSCALL msg=audit(1771076608.935:1724): arch=c000003e syscall=321 success=yes
```

Рисунок 3.14: Попытка прочесть лог-файл

Чтобы запустить веб-сервер Apache на прослушивание TCP-порта 81 (а не 80, как рекомендует IANA и прописано в `/etc/services`) открываю файл `/etc/httpd/httpd.conf` для изменения. (рис. [fig:015]).

```
pisimonova@pisimonova:~$ sudo nano /etc/httpd/httpd.conf
```

Рисунок 3.15: Изменение файла

Нахожу строчку `Listen 80` и заменяю её на `Listen 81`. (рис. [fig:016]).

```
#
# 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
#
```

Рисунок 3.16: Изменение порта

Выполняю перезапуск веб-сервера Apache. Произошёл сбой, потому что порт 80 для локальной сети, а 81 нет (рис. [fig:017]).

Попытка соединения не удалась

Firefox не может установить соединение с сервером 127.0.0.1.

- Возможно, сайт временно недоступен или перегружен запросами. Подождите некоторое время и попробуйте снова.
- Если вы не можете загрузить ни одну страницу – проверьте настройки соединения с Интернетом.
- Если ваш компьютер или сеть защищены межсетевым экраном или прокси-сервером – убедитесь, что Firefox разрешён выход в Интернет.

Попробовать снова

Рисунок 3.17: Попытка прослушивания другого порта

Проанализируйте лог-файлы: `tail -n1 /var/log/messages` (рис. [fig:018]).

```
pisimonova@pisimonova:~$ sudo nano /etc/httpd/httpd.conf
```

Рисунок 3.18: Проверка лог-файлов

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. Запись появилась в файлу `error_log` (рис. [fig:019]).

```

pisimonova@pisimonova:~$ semanage port -l | grep http_port_t
ValueError: Политика SELinux не задана, или нет доступа к хранилищу.
pisimonova@pisimonova:~$ sudo nano /etc/httpd/httpd.conf
pisimonova@pisimonova:~$ sudo 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,permissive,dontaudit}
               ...
semanage: error: unrecognized arguments: -p 81
pisimonova@pisimonova:~$ sudo semanage port -l | grep http_port_t
http_port_t      tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
http_port_t      udp      80, 443
pegasus_http_port_t tcp      5988
pisimonova@pisimonova:~$ sudo systemctl resart httpd
Unknown command verb 'resart', did you mean 'restart'?
pisimonova@pisimonova:~$ sudo systemctl restart httpd
pisimonova@pisimonova:~$ sudo chcon -t httpd_sys_content_t /var/www/html/test.htm
l

```

Рисунок 3.19: Проверка лог-файлов

Выполняю команду `semanage port -a -t http_port_t -p tcp 81`. После этого проверяю список портов командой `semanage port -l | grep http_port_t`. Порт 81 появился в списке (рис. [fig:020]).

```

pisimonova@pisimonova:~$ sudo 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,permissive,dontaudit}
               ...
semanage: error: unrecognized arguments: -p 81
pisimonova@pisimonova:~$ sudo semanage port -l | grep http_port_t
http_port_t      tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
http_port_t      udp      80, 443
pegasus_http_port_t tcp      5988
pisimonova@pisimonova:~$ sudo systemctl resart httpd
Unknown command verb 'resart', did you mean 'restart'?
pisimonova@pisimonova:~$ sudo systemctl restart httpd

```

Рисунок 3.20: Проверка портов

Перезапускаю сервер Apache (рис. [fig:021]).

```

pisimonova@pisimonova:~$ sudo systemctl restart httpd
pisimonova@pisimonova:~$ sudo chcon -t httpd_sys_content_t /var/www/html/test.htm
l
pisimonova@pisimonova:~$ sudo systemctl restart httpd

```

Рисунок 3.21: Перезапуск сервера

Теперь он работает, ведь мы внесли порт 81 в список портов `httpd_port_t` (рис. [fig:022]).

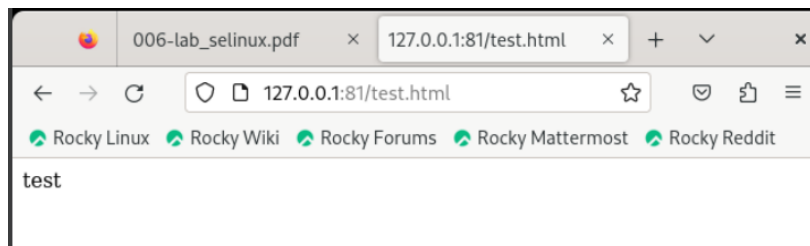


Рисунок 3.22: Проверка сервера

Возвращаю в файле `/etc/httpd/httpd.conf` порт 80, вместо 81. Проверяю, что порт 81 удален, это правда. (рис. [fig:023]).

```
pisimonova@pisimonova:~$ sudo nano /etc/httpd/httpd.conf
pisimonova@pisimonova:~$ semanage port -d -t http_port_t -p tcp 81
ValueError: Политика SELinux не задана, или нет доступа к хранилищу.
```

Рисунок 3.23: Проверка порта 81

Далее удаляю файл `test.html`, проверяю, что он удален(рис. [fig:024]).

```
pisimonova@pisimonova:~$ ls -lZ /var/www/html
итого 4
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 33 фев 14 16
:36 test.html
pisimonova@pisimonova:~$ rm /var/www/html/test.html
```

Рисунок 3.24: Удаление файла

4 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.

5 Список литературы

[0] Методические рекомендации курса